Management Guide

AppManager® for Microsoft Windows

October 2019

Legal Notice

Contents

7 PowerShell Knowledge Scripts 257

8 General Knowledge Scripts 261

10 ASYNC Knowledge Scripts 345

11 Action Knowledge Scripts 365

About NetIQ Corporation

We are a global, enterprise software company, with a focus on the three persistent challenges in your environment: Change, complexity and risk—and how we can help you control them.

Our Viewpoint

Adapting to change and managing complexity and risk are nothing new

In fact, of all the challenges you face, these are perhaps the most prominent variables that deny you the control you need to securely measure, monitor, and manage your physical, virtual, and cloud computing environments.

Enabling critical business services, better and faster

We believe that providing as much control as possible to IT organizations is the only way to enable timelier and cost effective delivery of services. Persistent pressures like change and complexity will only continue to increase as organizations continue to change and the technologies needed to manage them become inherently more complex.

Our Philosophy

Selling intelligent solutions, not just software

In order to provide reliable control, we first make sure we understand the real-world scenarios in which IT organizations like yours operate — day in and day out. That's the only way we can develop practical, intelligent IT solutions that successfully yield proven, measurable results. And that's so much more rewarding than simply selling software.

Driving your success is our passion

We place your success at the heart of how we do business. From product inception to deployment, we understand that you need IT solutions that work well and integrate seamlessly with your existing investments; you need ongoing support and training post-deployment; and you need someone that is truly easy to work with — for a change. Ultimately, when you succeed, we all succeed.

Our Solutions

- Identity & Access Governance
- Access Management
- Security Management
- Systems & Application Management
- Workload Management
- Service Management

Contacting Sales Support

For questions about products, pricing, and capabilities, contact your local partner. If you cannot contact your partner, contact our Sales Support team.

Worldwide:	www.netiq.com/about_netiq/officelocations.asp
United States and Canada:	1-888-323-6768
Email:	info@netiq.com
Web Site:	www.netiq.com

Contacting Technical Support

For specific product issues, contact our Technical Support team.

Worldwide:	www.netiq.com/support/contactinfo.asp
North and South America:	1-713-418-5555
Europe, Middle East, and Africa:	+353 (0) 91-782 677
Email:	support@netiq.com
Web Site:	www.netiq.com/support

Contacting Documentation Support

Our goal is to provide documentation that meets your needs. The documentation for this product is available on the NetIQ Web site in HTML and PDF formats on a page that does not require you to log in. If you have suggestions for documentation improvements, click **comment on this topic** at the bottom of any page in the HTML version of the documentation posted at www.netiq.com/documentation. You can also email Documentation-Feedback@netiq.com. We value your input and look forward to hearing from you.

Contacting the Online User Community

NetIQ Communities, the NetIQ online community, is a collaborative network connecting you to your peers and NetIQ experts. By providing more immediate information, useful links to helpful resources, and access to NetIQ experts, NetIQ Communities helps ensure you are mastering the knowledge you need to realize the full potential of IT investments upon which you rely. For more information, visit community.netiq.com.

About this Book and the Library

The NetIQ AppManager product (AppManager) is a comprehensive solution for managing, diagnosing, and analyzing performance, availability, and health for a broad spectrum of operating environments, applications, services, and server hardware.

AppManager provides system administrators with a central, easy-to-use console to view critical server and application resources across the enterprise. With AppManager, administrative staff can monitor computer and application resources, check for potential problems, initiate responsive actions, automate routine tasks, and gather performance data for real-time and historical reporting and analysis.

Intended Audience

This guide provides information for individuals responsible for installing an AppManager module and monitoring specific applications with AppManager.

Other Information in the Library

The library provides the following information resources:

Installation Guide for AppManager

Provides complete information about AppManager pre-installation requirements and step-by-step installation procedures for all AppManager components.

User Guide for AppManager Control Center

Provides complete information about managing groups of computers, including running jobs, responding to events, creating reports, and working with Control Center. A separate guide is available for the AppManager Operator Console.

Administrator Guide for AppManager

Provides information about maintaining an AppManager management site, managing security, using scripts to handle AppManager tasks, and leveraging advanced configuration options.

Upgrade and Migration Guide for AppManager

Provides complete information about how to upgrade from a previous version of AppManager.

Management guides

Provide information about installing and monitoring specific applications with AppManager.

Help

Provides context-sensitive information and step-by-step guidance for common tasks, as well as definitions for each field on each window.

The AppManager library is available in Adobe Acrobat (PDF) format from the AppManager Documentation page of the NetIQ Web site.

1 Introducing AppManager for Microsoft Windows

This chapter provides an overview of monitoring Microsoft Windows operating systems with AppManager.

1.1 Features and Benefits

AppManager for Microsoft Windows provides several categories of Knowledge Scripts that enable you to identify and monitor the health, availability, and performance of key resources. These scripts allow you to monitor and manage crucial resource properties at a depth unparalleled by any other solution. You can configure each Knowledge Script to raise an event, collect data for reporting, and perform automated problem management when an event occurs.

With AppManager for Microsoft Windows, you gain access to a set of tools you can leverage to gather a wide range of diagnostic and management data, which can help prevent outages and keep things running smoothly.

In AppManager 9.1 and later releases, AppManager for Microsoft Windows is shown as an application of the Windows agent on which it is running, in the Control Center Navigation pane and the Operator Console Treeview.

AppManager provides a comprehensive solution for monitoring Microsoft Windows. With AppManager for Microsoft Windows, you can:

- View all discovered Windows servers and operating system configuration details
- Discover and manage the virtual server cluster alias
- Monitor the status of important Windows services, including Distributed File System (DFS), File Replication Service (FRS), Internet Authentication Service (IAS), Remote Storage service, Quality of Service (QoS), and Resource Reservation Protocol (RSVP)
- Monitor Windows event logs and raise events when expected entries are not present
- Monitor recursive queries, secure updates, query activity, Windows Internet Name Service (WINS) activity, the number of dynamic updates queued, and update errors for a DNS server
- Monitor and refresh group policies
- Monitor logical and physical disk statistics
- Monitor printer errors, events, queue length, and bytes printed per second
- Monitor the status and length of SMTP queues
- Monitor Microsoft automatic update (AU) activity, Background Intelligent Transfer Service (BITS) activity, the Distributed COM (DCOM) list, fax activity, plug-and-play (PNP) activity, and System Restore (SR) service activity
- Monitor SNMP traps forwarded from NetIQ SNMP Trap Receiver
- Run PowerShell commands

1.2 Understanding Windows Knowledge Script Categories

AppManager for Windows provides Knowledge Script categories for different versions of the Microsoft Windows operating system. These Knowledge Script categories are grouped according to functional areas.

Typically the Knowledge Script categories are upward-compatible. For example, the NT Knowledge Scripts and NTAdmin Knowledge Scripts run on Windows Server 2003, Windows XP, Windows Vista, Windows Server 2008, Windows Server 2008 R2, Windows 7, Windows 8, Windows 10, and Windows Server 2012. The WIN2000 Knowledge Scripts run on Windows Server 2003, Windows Server 2008, and Windows Server 2012.

Use the existing categories of Knowledge Scripts, such as NT, NTAdmin, and WIN2000, to monitor Windows Server 2008 and Windows Server 2012 computers.

There is no separate category of Knowledge Scripts for Windows Server 2008 or later.

2 Installing and Configuring AppManager for Microsoft Windows

This chapter provides installation instructions and describes system requirements for AppManager for Microsoft Windows.

This chapter assumes you have AppManager installed. For more information about installing AppManager or about AppManager system requirements, see the *Installation Guide for AppManager*, which is available on the AppManager Documentation page.

NOTE: Version 7.x or later of the AppManager for Microsoft Windows module is not supported on Windows NT computers. However, if you have an older version of the AppManager for Microsoft Windows module installed on a Windows NT computer, you can continue to monitor that computer.

2.1 System Requirements

For the latest information about supported software versions and the availability of module updates, visit the AppManager Supported Products page. Unless noted otherwise, this module supports all updates, hotfixes, and service packs for the releases listed below.

AppManager for Microsoft Windows has the following requirements:

Software/Hardware	Version
NetIQ AppManager installed on the AppManager repository (QDB) computers, on the Windows computers you want to monitor (agents), and on all console computers	8.0.3, 8.2, 9.1, 9.2, 9.5, or later AppManager agent 8.0.3, 8.2, 9.1, 9.2, 9.5, or later is required: **NOTE:** For more information about Hotfixes, see the AppManager Suite Hotfixes page.
Microsoft operating system installed on the agent computers	One of the following operating systems: ◆ Windows Server 2019 ◆ Windows Server 2016 ◆ Windows Server 2012 R2 ◆ Windows Server 2012 ◆ Windows 10 (32-bit or 64-bit) ◆ Windows 8.1 (32-bit or 64-bit) ◆ Windows 8 (32-bit or 64-bit) ◆ Windows 7 (32-bit or 64-bit) ◆ Windows Server 2008 R2, including the Intel Itanium (IA64) architecture ◆ Windows Server 2008 (32-bit or 64-bit), including the Intel Itanium (IA64) architecture

Software/Hardware	Version
Microsoft Windows PowerShell installed on the agent computers where you want to execute PowerShell commands	1.0 or later **NOTE:** PowerShell is not supported on the Intel Itanium (IA64) platform.
Microsoft .NET Framework installed on the agent computers where you want to run the `PowerShell_RunCommand` and `Action_RunPowerShell` Knowledge Scripts	3.0 or later
Microsoft .NET Framework installed on the agent computers where you want updated monitoring of Windows event logs	Version 3.5 or later to obtain the latest monitoring available with General_EventLog and General_EventLogRx
AppManager for Microsoft Windows installed on management servers	7.6 for support of Action Knowledge Script jobs running on management servers
Microsoft SQL Server Native Client 11.0 **(for TLS 1.2 support)**	11.3.6538.0 or later **NOTE:** The SQL Server Native client can be installed from this Microsoft download link.

NOTE: If you want TLS 1.2 support and are running AppManager 9.1 or 9.2, then you are required to perform some additional steps. To know about the steps, see the article.

2.2 Installing the Module

Run the module installer on the Microsoft Windows computers you want to monitor (agents) to install the agent components, and run the module installer on all console computers to install the Help and console extensions.

Access the `AM70-WinOS-8`*`x.x`*`.0.msi` module installer from the `AM70_WinOS_7.`*`x.x`*`.0` self-extracting installation package on the AppManager Module Upgrades & Trials page.

For Windows environments where User Account Control (UAC) is enabled, install the module using an account with administrative privileges. Use one of the following methods:

- Log in to the server using the account named Administrator. Then, run the module installer `NT.msi` file from a command prompt or by double-clicking it.
- Log in to the server as a user with administrative privileges and run the module installer `NT.msi` file as an administrator from a command prompt. To open a command-prompt window at the administrative level, right-click a command-prompt icon or a Windows menu item and select **Run as administrator**.

You can install the Knowledge Scripts into local or remote AppManager repositories (QDBs). The module installer installs Knowledge Scripts for each module directly into the QDB instead of installing the scripts in the `\AppManager\qdb\kp` folder as in previous releases of AppManager.

You can install the module manually, or you can use Control Center to deploy the module on a remote computer where an agent is installed. For more information, see Section 2.3, “Deploying the Module with Control Center,” on page 36. However, if you do use Control Center to deploy the module, Control Center only installs the *agent* components of the module. The module installer installs the QDB and console components as well as the agent components on the agent computer.

NetIQ Corporation recommends that you install the agent using the Local System account.

To install the module manually:

1. Double-click the module installer `.msi` file.
2. Accept the license agreement.
3. Review the results of the pre-installation check. You can expect one of the following three scenarios:
    - **No AppManager agent is present**. In this scenario, the pre-installation check fails, and the installer does not install agent components.
    - **An AppManager agent is present, but some other prerequisite fails**. In this scenario, the default is to not install agent components because of one or more missing prerequisites. However, you can override the default by selecting **Install agent component locally**. A missing application server for this particular module often causes this scenario. For example, installing the AppManager for Microsoft SharePoint module requires the presence of a Microsoft SharePoint server on the selected computer.
    - **All prerequisites are met**. In this scenario, the installer will install the agent components.
4. To install the Knowledge Scripts into the QDB:

    4a Select **Install Knowledge Scripts** to install the repository components, including the Knowledge Scripts, object types, and SQL stored procedures.

    4b Specify the SQL Server name of the server hosting the QDB, as well as the case-sensitive QDB name.
5. (Conditional) If you use Control Center 7.x, run the module installer for each QDB attached to Control Center.
6. (Conditional) If you use Control Center 8.x or later, run the module installer only for the primary QDB, and Control Center will automatically replicate this module to secondary QDBs.
7. Run the module installer on all console computers to install the Help and console extensions.
8. Run the module installer on the Windows computers you want to monitor (agents) to install the agent components.
9. Run the module installer on all management server computers to support Action Knowledge Scripts.
10. (Conditional) If you want to use NetIQ Trap Receiver to check for SNMP traps, install Trap Receiver by running `\AppManager\bin\NetIQTrapReceiver_Setup.exe`. For more information, see SNMPTrap.
11. (Conditional) If you have not discovered Windows resources, run the Discovery_NT Knowledge Script on all agent computers where you installed the module.
12. (Conditional) If your environment contains virtual server applications on clusters, such as Microsoft SQL Server and Microsoft Exchange, run the Discovery_Cluster Knowledge Script to discover applications on clustered servers and the virtual cluster server.
13. To get the updates provided in this release, upgrade any running Knowledge Script jobs. For more information, see Section 2.5, “Upgrading Knowledge Script Jobs,” on page 37.

After the installation has completed, the `WinOS_Install.log` file, located in the `\NetIQ\Temp\NetIQ_Debug\`<*ServerName*> folder, lists any problems that occurred.

NOTE: To view Reports in ServerGroup, install AppManager Microsoft Windows KS on all AM repositories (QDB) using the Install Knowledge Scripts option. This updates the report pack, which is needed to have the construct to query server groups when configuring ReportAM_KS reports.

2.3 Deploying the Module with Control Center

You can use Control Center to deploy the module on a remote computer where an agent is installed. This topic briefly describes the steps involved in deploying a module and provides instructions for checking in the module installation package. For more information, see the *Control Center User Guide for AppManager*, which is available on the AppManager Documentation page.

Deployment Overview

This section describes the tasks required to deploy the module on an agent computer.

To deploy the module on an agent computer:

1. Verify the default deployment credentials.
2. Check in an installation package. For more information, see "Checking In the Installation Package" on page 36.
3. Configure an email address to receive notification of a deployment.
4. Create a deployment rule or modify an out-of-the-box deployment rule.
5. Approve the deployment task.
6. View the results.

Checking In the Installation Package

You must check in the installation package, `AM70-WinOS-8.`*`x.x`*`.0.xml`, before you can deploy the module on an agent computer.

To check in a module installation package:

1. Log on to Control Center using an account that is a member of a user group with deployment permissions.
2. Navigate to the **Deployment** tab (for AppManager 8.x or later) or **Administration** tab (for AppManager 7.x).
3. In the Deployment folder, select **Packages**.
4. On the Tasks pane, click **Check in Deployment Packages** (for AppManager 8.x or later) or **Check in Packages** (for AppManager 7.x).
5. Navigate to the folder where you saved `AM70-WinOS-8.`*`x.x`*`.0.xml` and select the file.
6. Click **Open**. The Deployment Package Check in Status dialog box displays the status of the package check in.
7. To get the updates provided in this release, upgrade any running Knowledge Script jobs. For more information, see Section 2.5, "Upgrading Knowledge Script Jobs," on page 37.

2.4 Silently Installing the Module

To silently (without user intervention) install a module using the default settings, run the following command in Administrator mode from the folder in which you saved the module installer:

```
msiexec.exe /i "AM70-WinOS-8.x.x.0.msi" /qn
```

where *x.x* is the actual version number of the module installer.

To get the updates provided in this release, upgrade any running Knowledge Script jobs. For more information, see Section 2.5, “Upgrading Knowledge Script Jobs,” on page 37.

To create a log file that describes the operations of the module installer, add the following flag to the command noted above:

```
/L* "AM70-WinOS-8.x.x.0.msi.log"
```

The log file is created in the directory in which you saved the module installer.

NOTE: To perform a silent install on an AppManager agent running Windows Server 2008 R2 or Windows Server 2012, open a command prompt at the administrative level and select **Run as administrator** before you run the silent install command listed above.

To silently install the module on a remote AppManager repository, you can use Windows authentication or SQL authentication.

Windows authentication:

```
AM70-WinOS-8.x.x.0.msi /qn MO_B_QDBINSTALL=1 MO_B_MOINSTALL=0
MO_B_SQLSVR_WINAUTH=1 MO_SQLSVR_NAME=SQLServerName MO_QDBNAME=AM-RepositoryName
```

SQL authentication:

```
AM70-WinOS-8.x.x.0.msi /qn MO_B_QDBINSTALL=1 MO_B_MOINSTALL=0
MO_B_SQLSVR_WINAUTH=0 MO_SQLSVR_USER=SQLLogin MO_SQLSVR_PWD=SQLLoginPassword
MO_SQLSVR_NAME=SQLServerName MO_QDBNAME=AM-RepositoryName
```

2.5 Upgrading Knowledge Script Jobs

The module upgrade process *retains* any changes you may have made to the parameter settings for the Knowledge Scripts in the previous version of this module.

NOTE: Unless you review the management guide or the online Help for that Knowledge Script, you will not know about any changes to default parameter values that came with this release.

You can push the changes for updated scripts to running Knowledge Script jobs in one of the following ways:

- Use the AMAdmin_UpgradeJobs Knowledge Script.
- Use the Properties Propagation feature.

Running AMAdmin_UpgradeJobs

The AMAdmin_UpgradeJobs Knowledge Script can push changes to running Knowledge Script jobs. Upgrading jobs to use the most recent script version allows the jobs to take advantage of the latest script logic while maintaining existing parameter values for the job.

For more information, see the **Help** for the AMAdmin_UpgradeJobs Knowledge Script.

Propagating Knowledge Script Changes

You can propagate script changes to jobs that are running and to Knowledge Script Groups, including recommended Knowledge Script Groups and renamed Knowledge Scripts.

Before propagating script changes, verify that the script parameters are set to your specifications. New parameters may need to be set appropriately for your environment or application.

You can choose to propagate only properties (specified in the Schedule and Values tabs), only the script (which is the logic of the Knowledge Script), or both. Unless you know specifically that changes affect only the script logic, you should propagate both properties and the script.

For more information about propagating Knowledge Script changes, see the “Running Monitoring Jobs” chapter of the *Operator Console User Guide for AppManager*.

Propagating Changes to Ad Hoc Jobs or Knowledge Script Groups

You can propagate the properties and the logic (script) of a Knowledge Script to ad hoc jobs started by that Knowledge Script. Corresponding jobs are stopped and restarted with the Knowledge Script changes.

You can also propagate the properties and logic of a Knowledge Script to corresponding Knowledge Script Group members. After you propagate script changes to Knowledge Script Group members, you can propagate the updated Knowledge Script Group members to associated running jobs. Any monitoring jobs started by a Knowledge Script Group member are restarted with the job properties of the Knowledge Script Group member.

To propagate changes to ad hoc Knowledge Script jobs or Knowledge Script Groups:

1. In the Knowledge Script view, select the Knowledge Script or Knowledge Script Group for which you want to propagate changes.
2. Right-click the script or group and select **Properties propagation** > **Ad Hoc Jobs**.
3. Select the components of the Knowledge Script that you want to propagate to associated ad hoc jobs or groups and click **OK**:

Select	To propagate
Script	The logic of the Knowledge Script.
Properties	Values from the Knowledge Script Schedule and Values tabs, such as schedule, monitoring values, actions, and advanced options. The module upgrade process *retains* any changes you might have made to the parameter settings for the Knowledge Scripts in the previous version of this module

2.6 Configuring the PowerShell Execution Policy

The PowerShell Execution Policy determines whether PowerShell scripts are allowed to run. By default, the Execution Policy is set to `Restricted`. If you try to run scripts under the `Restricted` policy, AppManager generates error messages.

NOTE: If you are using PowerShell version 2.0 or later, you can ignore the steps in this topic, as well as topics 2.7, 2.8, and 2.9. You can move on to Section 2.10, “Changing PowerShell Configuration Settings,” on page 42. However, you may need to refer to these four topics if the certificate-related steps fail during the installation process.

If you are using PowerShell version 1.0, follow the steps in this topic as well as topics 2.7, 2.8, and 2.9.

The Execution Policy directly affects the PowerShell Knowledge Scripts. Although these Knowledge Scripts are written in VBScript and installed as <*scriptname.qml*>, the logic for the scripts is contained in complementary PowerShell scripts that are installed on the agent computer along with the module. The PowerShell scripts use the same name as the PowerShell Knowledge Scripts, but with a `.ps1` extension.

The digital signature encoded in a PowerShell Knowledge Script is tied to the contents of the script. If you change the script, the signature is no longer valid and you cannot execute the script. If you change a PowerShell Knowledge Script, you must do one of the following:

- Re-sign the scripts using your own digital certificate.
- Change the Execution Policy to either **RemoteSigned** or **Unrestricted**. A group policy that governs script execution overrides any policy changes you might make with the `Set-ExecutionPolicy` cmdlet. For example, if the group policy forbids script execution, you cannot change the policy by running `Set-ExecutionPolicy`. You must first change the group policy to allow script execution, and then run `Set-ExecutionPolicy` to select a specific Execution Policy.

Before AppManager can execute the PowerShell Knowledge Scripts, change the Execution Policy from `Restricted` to one of the following policy options:

AllSigned

Allows execution of scripts that have been digitally signed by a trusted publisher. If you select the **AllSigned** policy, perform the steps outlined in Section 2.7, “Trusting PowerShell Knowledge Scripts,” on page 40.

RemoteSigned

Allows local scripts to run regardless of signature, and requires trusted digital signatures only for remote scripts. Chapter 7, “PowerShell Knowledge Scripts,” on page 257 are local scripts.

Unrestricted

Allows both local and remote scripts to run, regardless of signature.

To change the PowerShell Execution Policy:

1 Open the Command Shell on an agent computer.

NOTE: On 64-bit Windows computers, use the **Windows PowerShell** Command Shell to change the Execution Policy. Do not use the Windows PowerShell (x86) Command Shell.

2 Run the following cmdlet:

```
Set-ExecutionPolicy <policy>
```

where <*policy*> is the name of the Execution Policy you want to change.

3 Repeat steps **1** and **2** on all agent computers.

2.7 Trusting PowerShell Knowledge Scripts

When a PowerShell script is executed under an AllSigned policy, PowerShell verifies that the script contains a digital signature and that the signature is associated with a trusted publisher. NetIQ Corporation signs the AppManager for Windows PowerShell scripts. If you use the **AllSigned** policy, you must choose to trust NetIQ Corporation by importing the NetIQ Corporation digital certificate into the local certificate store on *each* Windows server in your environment.

You can import the digital certificate by running one of the AppManager for Windows PowerShell scripts from the command line.

To import the digital certificate:

1. Open the Command Shell on the agent computer.
2. Change to the `AppManager\bin\PowerShell\Scripts` directory.
3. Type `.\PowerShell_RunCommand.ps1`.
4. Press `Enter`.
5. Type `A` at the prompt asking whether the script should be allowed to run.
6. Press `Enter`.

These steps allow the NetIQ Corporation digital certificate to be imported into the certificate store for the user running the script. You need to run only one script to establish trust. It does not matter which script you run.

At this point, trust is established *only* between NetIQ Corporation and the user running the script. *Trust is not established for any other user*. If the AppManager agent runs under a different user account such as Local System, a domain account, or a local computer account, the agent will not have a trust relationship and will not be allowed to execute the AppManager for Windows PowerShell scripts.

To extend trust to all other user accounts, see Section 2.8, “Extending Trust to All User Accounts,” on page 40.

To establish trust between all users accounts and the Microsoft digital certificate, see Section 2.9, “Establishing Trust for the Microsoft Certificate,” on page 42.

2.8 Extending Trust to All User Accounts

To execute PowerShell scripts under the **AllSigned** Execution Policy, extend trust to all user accounts. Extending trust is a two-phase process that involves exporting the digital certificate from the current user and importing the digital certificate to all users on the local computer.

Exporting the NetIQ Corporation Digital Signature Certificate

To extend trust to all user accounts, first export the NetIQ Corporation digital signature certificate from the current user using the Microsoft Management Console.

To export the NetIQ Corporation digital signature certificate from the current user:

1. On the **Start** menu, click **Run**.
2. In the **Open** field, type `mmc.exe`, and then click **OK**.

3. On the **File** menu, click **Add/Remove Snap-in**.
4. Click **Add** and then select the **Certificates** snap-in.
5. Click **Add**, select **My user account**, and then click **Finish**.
6. Click **Close**, and then click **OK**.
7. Expand the **Certificates - Current User** node.
8. Expand the **Trusted Publishers** sub-node and select the **Certificates** sub-node.
9. In the right pane, right-click the **NetIQ** certificate, select **All Tasks**, and then select **Export**.
10. Click **Next** in the Certificate Export Wizard.
11. Select **DER encoded binary**, and then click **Next**.
12. Click **Browse**, select the **Desktop** icon, type `NetIQ` in the **File name** field, and then click **Save**.
13. Click **Next**, and then click **Finish**.

Importing the NetIQ Corporation Digital Signature

The next phase of extending trust to all user accounts involves importing the NetIQ Corporation digital signature to all users on the local computer. Use the Microsoft Management Console to execute the import procedure.

To import the NetIQ Corporation digital certificate to all users on the local computer:

1. On the File menu in the Microsoft Management Console window, click **Add/Remove Snap-in**.
2. Click **Add**, and then select the **Certificates** snap-in.
3. Click **Add**, select **Computer account**, and then click **Next**.
4. Select **Local computer**, and then click **Finish**.
5. Click **Close**, and then click **OK**.
6. Expand **Certificates (Local Computer)** and select **Trusted Publishers**.
7. Right-click in the right pane, select **All Tasks**, and then select **Import**.
8. Click **Next** in the Certificate Import Wizard.
9. Click **Browse**, click the **Desktop** icon, select **NetIQ.cer**, and then click **Open**.
10. Click **Next** in the Wizard.
11. Select **Place all certificates in the following store**.
12. Click **Browse**, and then select **Show physical stores**.
13. Expand **Trusted Publishers** and select **Local Computer**.
14. Click **OK**.
15. Click **Next** in the Certificate Import Wizard, and then click **Finish**.

After you complete both the phases of the trust process, the NetIQ Corporation certificate is contained in the certificate store for the local computer, allowing a user to execute the PowerShell scripts.

2.9 Establishing Trust for the Microsoft Certificate

For the PowerShell Knowledge Scripts to run properly, a trust relationship must exist between the Microsoft digital signature certificate and the user account under which the AppManager agent is running.

To create this trust relationship, perform the steps outlined in Section 2.8, “Extending Trust to All User Accounts,” on page 40, with the following exceptions:

- In Step 9 on page 41, select **Microsoft.cer** instead of **NetIQ.cer**.
- In Step 12 on page 41, type `Microsoft` instead of `NetIQ`.

2.10 Changing PowerShell Configuration Settings

AppManager for Windows includes the following components:

- A client object, `MCPSHostClient.dll`, which runs within the AppManager agent. This client object starts the server program and asks it to run jobs.
- A server program, `MCPSHostServer.exe`, which provides the PowerShell environment in which the PowerShell Knowledge Scripts are executed.

Both components have associated configuration files that define certain operational parameters. You can modify some of these settings to fine-tune performance or to specify resource usage limits.

The configuration files are in `.XML` format. After making changes, ensure that the files retain their well-formed `.XML` format. Also do not remove or change settings other than those documented here. NetIQ Corporation strongly recommends that you create backup copies of these files before modifying them.

NOTE: This topic does not discuss all configuration settings. As a rule, if a configuration setting is not discussed in this topic, do not change the value of that setting.

Client Configuration Settings

The client configuration file, `MCPSHostClient.dll.config`, resides in the `AppManager\bin\PowerShell` directory. You can change the following settings.

In the `<appSettings>` section:

maxActiveServers

Use this setting to specify the maximum number of servers that can be active at any time. Use this setting in conjunction with `maxMemoryUsage` to specify a lower memory threshold with an increased number of servers that can be used. This combination is beneficial for situations in which a server exceeds the memory limitation and has to shut down. If only one server can be active at a time, job requests are blocked until the server restarts. If you allow more than one server to be active, job requests can be executed in other server processes or on new servers if the current number of active servers is less than `maxActiveServers`.

serverStartupTimeout

> If `MCPSHostServer.exe` is not already running when a job is scheduled for execution, the client starts the server automatically. After starting the server, the client attempts to contact it. Use this configuration setting to specify the number of seconds that the client should attempt to contact the server. An error event is raised if the client cannot contact the server within the specified period.

In the `<log4net>` section:

file

> Use this setting to specify the pathname of the log file. If the pathname is a relative path, it is considered to be relative to the `\AppManager\bin\PowerShell` directory.

appendToFile

> Use this setting to indicate whether the client overwrites the existing log file or appends to it, at the time the client is loaded into the AppManager agent.

maxSizeRollBackups

> Use this setting to specify the number of old log files you want to retain.

maximumFileSize

> Use this setting to specify the maximum size of a log file. After a log file reaches this size, it is deleted, or renamed if the `maxSizeRollBackups` value is greater than `0`.

Server Configuration Settings

The server configuration file, `MCPSHostServer.exe.config`, resides in the `AppManager\bin\PowerShell` directory. You can change the following settings.

In the `<appSettings>` section:

serverShutdownTimeout

> Use this setting to specify the number of seconds that the server will remain running when no jobs are executing. If no jobs are submitted to the server during this period, the server shuts down and will restart the next time a client needs to run a job.

upperMaxRunspaceHosts

> The PowerShell runspace pool allocates runspaces as needed. Each execution of a job requires one runspace. Runspaces return to the pool after use and are then available for other jobs. Use this setting to set the absolute limit on the number of runspaces allocated for a pool. If a client requests a runspace when none is available and the pool has reached this limit, the client is blocked from running until a runspace becomes available.

If you do not specify the runspace setting, the pool always allocates a new runspace, even if all others are in use, thereby ensuring that clients never have to wait for a runspace to be available.

maxActiveServers

> Use this setting to specify the maximum number of servers that can be active at any time. Use this setting in conjunction with `maxMemoryUsage` to specify a lower memory threshold with an increased number of servers that can be used. This combination is beneficial for situations in which a server exceeds the memory limitation and has to shut down. If only one server can be active at a time, job requests are blocked until the server restarts. If you allow more than one server to be active, job requests can be executed in other server processes or on new servers if the current number of active servers is less than `maxActiveServers`.

maxMemoryUsage

Use this setting to specify the maximum amount of memory, in megabytes, that the server process should consume. If memory usage exceeds the maximum size, the server blocks additional requests from clients and restarts automatically after the last client has finished job execution.

In the `<log4net>` section:

file

Use this setting to specify the pathname of the log file. If the pathname is a relative path, it is considered to be relative to the `\AppManager\bin\PowerShell` directory.

appendToFile

Use this setting to indicate whether the client overwrites the existing log file or appends to it, at the time the client is loaded into the AppManager agent.

maxSizeRollBackups

Use this setting to specify the number of old log files you want to retain.

maximumFileSize

Use this setting to specify the maximum size of a log file. After a log file reaches this size, it is deleted, or renamed if the `maxSizeRollBackups` value is greater than 0.

2.11 Troubleshooting Knowledge Script Errors

Knowledge Scripts that collect performance counters, might return an error that a particular performance counter cannot be found.

While using Knowledge Scripts that collect performance counters, if any job returns an error indicating a particular performance counter cannot be found, look in the registry on the agent computer where the job was running, and open the registry key, `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\PerfProc\Performance`

1. In the right pane, double-click `Disable Performance Counters`.
2. Review the value data to ensure that it is set to `0`. If the value is not 0, change it.
3. Click OK, and the close the Registry Editor window.

NOTE: A reboot of the agent computer may be required to allow the counters to be visible again.

2.12 Troubleshooting PowerShell Errors

PowerShell Knowledge Scripts may raise such events as "`PowerShell script failed to run to completion`" or "`Error executing PowerShell script.`" These errors can occur when Knowledge Scripts take a long time to run, or when there is contention for access to the server that executes the PowerShell scripts, `MCPSHostServer.exe`. The following are recommendations for resolving these issues.

Increase the amount of memory that can be used by MCPSHostServer.exe

Increasing the memory limit reduces the frequency with which the server restarts due to excessive memory usage. Increasing the memory limit also reduces the number of PowerShell errors; each time the server recognizes that it is exceeding its memory usage threshold, the server prevents new jobs from executing until all existing jobs have completed and the server restarts. If existing jobs take a significant amount of time to complete, the waiting jobs may time out and return errors. To increase the amount of memory `MCPSHostServer.exe` is allowed to use, modify the value of the `maxMemoryUsage` setting. For more information, see Section 2.10, “Changing PowerShell Configuration Settings,” on page 42.

Increase the number of PowerShell execution environments or runspaces that MCPSHostServer.exe can host

The default number of runspaces is five, which means no more than five Knowledge Script jobs can be running simultaneously in the server. If you attempt to run additional jobs, the jobs are held back until runspaces become available as existing jobs complete their iterations. Being held back increases the chance that jobs will time out before running, or before completing their iteration. To increase the number of available runspaces, modify the `upperMaxRunspaceHosts` setting. For more information, see Section 2.10, “Changing PowerShell Configuration Settings,” on page 42.

Note that increasing this value will be beneficial if you are running more than five PowerShell Knowledge Scripts jobs, but even then the benefit may not be significant.

3 Discovery Knowledge Scripts

The following Discovery Knowledge Scripts are available to help you discover Microsoft Windows resources and cluster resources.

From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. In the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
Discovery_NT	Discovers Microsoft Windows configuration and resource information.
Discovery_Cluster	Discovers clustered applications on physical computers and virtual machines.
Discovery_ReportAgent	Discovers the NetIQ AppManager report agent and its data sources.

3.1 Discovery_NT

Use this Knowledge Script to discover Microsoft Windows configuration and resource information, including related resources such as network services, printers, and .NET Common Language Runtime (CLR) objects.

Discover Windows resources before discovering clustered applications, if applicable in your environment. For more information, see Discovery_Cluster.

This script discovers shared disks under the Cluster Alias object in the Navigation pane or the TreeView and local disks under regular Windows server objects.

To discover all shared disks in a Windows cluster and the shared mount points rooted from those shared disks under the Cluster Alias object in the Navigation pane or the TreeView, make sure that the NetIQ Client Resource Monitor service (`NetIQmc`) is running under an account whose user is a member of the domain of the computer. If the `NetIQmc` service is running under an account that is not a domain account but is a member of the local Administrators group on the agent, review the following procedures on the Securing a Remote WMI Connection page from Microsoft to ensure the Discovery_NT script can gather the required disk information:

- *To grant DCOM remote launch and activation permissions for a user or group*
- *To grant DCOM remote access permissions*

If you delete or add a resource object, or if you make any other kind of change that might affect the monitoring of your resources, run the Discovery_NT Knowledge Script again to update your list of resource objects. In addition, if you are running this module on AppManager 8 or higher, you can use the delta discovery feature in Control Center to run discovery on a schedule to more quickly detect changes to your environment.

Resource Objects

Windows 2003 Server or later

Default Schedule

By default, this script is only run once for each computer.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity when discovery fails	Set the event severity level, from 1 to 40, to reflect the importance of an event in which discovery fails. This script always raises an event when the job fails for any reason. The default is 10.
Raise event when discovery succeeds?	Select **Yes** to raise an event when discovery succeeds. The default is unselected.
Event severity when discovery succeeds	Set the event severity level, from 1 to 40, to reflect the importance of an event in which discovery succeeds. The default is 35.
Discovery Options	
Discover accessible printers?	Select **Yes** to discover accessible printers on the specified computer. Deselect the check box to disable the discovery of all printers if you have a large number of printers for a single AppManager agent to prevent the discovery process from timing out while trying to discover all the printers. The default is unselected.
Discover .NET Common Language Runtime (CLR) objects?	Select **Yes** to discover .NET Common Language Runtime (CLR) objects. If you select Yes and enable delta discovery, the delta discovery process might generate an excessive number of events that are not significant, because the list of applications that use CLR can change quite often. The default is unselected.
Discover Microsoft BITS Jobs?	Select **Yes** to discover Microsoft BITS jobs. Selecting **Yes** might block successful completion of discovery if the BITS jobs are exceedingly high. The default is Yes.
Discover DFS Link Objects?	Select **Yes** to discover DFS link objects. Deselect the check box to skip the discovery of DFS link objects. The default is Yes.

3.2 Discovery_Cluster

Use this Knowledge Script to discover clustered applications on physical computers and virtual machines. This script also discovers the cluster alias and adds it as a top-level resource object. This script facilitates monitoring of clustered applications. It removes the need to run the SetResourceDependency Knowledge Script or run multiple jobs on each clustered node per instance of the application. Also, application failovers are not required to discover servers on each node.

Although virtual servers are viewed and monitored as objects, features such as pinging the physical computer are not available. Other settings such as maintenance mode, custom server properties and security information storage are not applicable for virtual servers.

Discovering and Monitoring Microsoft Cluster Resources with AppManager

NOTE: The terms 'node' and 'cluster node' are similar to 'agent computer' or 'computer upon which the Microsoft Cluster service is running and where you will install the AppManager agent software for monitoring.

To install AppManager components for discovery of cluster resources:

1. Create a server object in the TreeView pane for each cluster node. For example, create a server object for an AppManager agent computer.

   NOTE: Leave the option for discovering Windows components deselected.

2. Create a server group in the TreeView pane and move each newly created cluster node server object into that server group.
3. Install the AppManager agent software on each cluster node.
4. Install the following modules, which are required for monitoring, on each cluster node:

   4a AppManager for Microsoft Windows (WinOS) module. Automatic discovery is enabled for this module, so it will run by default at the end of its installation.

   4b Microsoft Cluster Service (MSCS) module. Then, run a Discovery_MSCS job on each cluster node in the TreeView pane.

   4c Other cluster-aware modules (Microsoft Exchange2007, Microsoft SQL Server) and any other modules that you need to have monitored.

5. Run Discovery for the following cluster resources:

   5a Run a Discovery_Cluster job on an active node in the cluster to identify the virtual server that the node is a part of and create a virtual Windows server in the console’s server group. Any Microsoft SQL or Microsoft Exchange virtual servers discovered will be added to the console’s server group.

   5b Run a Discovery_NT job on the Windows virtual server. The discovery job will determine the owner of the "network name" resource and only run on that node to determine the shared disks used across the cluster.

   IMPORTANT: Some shared disks used by the cluster nodes, including those mounted to remote computers, cannot be found at times because available programming interfaces do not show that information. These disks can still be monitored with the NT_DiskSpace Knowledge Script.

   5c Run a Discovery_SQL job against any Microsoft SQL Server virtual servers previously discovered and added to the server group.

   5d Run a Discovery_Exchange2007 job against any Microsoft Exchange virtual servers previously discovered and added to the server group.

   5e Run the Discovery Knowledge Scripts for other modules, as needed. For those that have their own active and passive cluster resources, ensure the Discovery job runs on active nodes, then failover the resources to the other nodes and run another Discovery job to the now-active nodes.

Considerations for monitoring cluster resources:

1. Only shared physical and logical disks that can be identified appropriately are discovered by Discovery_NT under the Windows virtual server object. Local disks are discovered when Discovery_NT runs on a physical agent computer in the TreeView pane.
2. For AppManager modules that are not fully cluster-aware (those other than Microsoft SQL Server and Exchange 2007), you must use the AMAdmin_SetResDependency Knowledge Script to control when to run a monitoring job on a given node. Use this Knowledge Script to identify a resource that is active when the node is active in the cluster.
3. When a job is created on a Windows virtual server object, Microsoft SQL Server virtual server object, or Microsoft Exchange virtual server object in the TreeView pane, child jobs will appear to be running on each node in the cluster as well as the virtual server. However, the job will apply data and events only to the virtual server child job.

Security Rights

To correctly discover and monitor a Microsoft cluster, this Knowledge Script requires local Administrator access to each node of the Microsoft cluster. To do this, run the `netiq` service as a domain user account and a member of the local Administrator group on each member of the cluster. Without this access, the discovery fails because it relies on the Microsoft Cluster API to properly access cluster resources.

Administering a Cluster

The Cluster Administrator can be used to administer a cluster, provided the account you are using has the required permissions and group memberships. The local Administrator account and local system account always have access to the cluster. You can use another account to administer a cluster with Cluster Administrator if the following requirements are true:

- The account has permission to administer the cluster. You must use Cluster Administrator to assign permissions, not Windows Group Administrator.
- The account is a domain account, which is a member of the local Administrators group.
- The account is a member of the local Administrators group on each node of the cluster.

The account can be a member of other groups, such as global groups, as long as it is a domain account.

The Discovery_Cluster script will only generate events if the *Raise event if condition occurs* option on the Advanced tab for the Discovery_Cluster script is set to raise an event one time within one job iteration.

By default, this Knowledge Script raises an event when discovery fails.

Prerequisite

Use the Discovery_NT Knowledge Script to discover Microsoft Windows resources on the server you want to monitor before you discover clustered applications.

If you delete or add a resource object, or if you make any other kind of change that might affect the monitoring of your resources, run both the Discovery_NT Knowledge Script and the Discovery_Cluster Knowledge Script again to update your list of resource objects. In addition, if you are running this module on AppManager 8 or higher, you can use the delta discovery feature in Control Center to run discovery on a schedule to more quickly detect changes to your environment.

Resource Objects

Microsoft Clustered Servers

Default Schedule

By default, this script is run once for each server.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event when discovery succeeds?	Set to **y** to raise an event when the job succeeds in discovering clustered resources. The default is n.
Event severity when discovery succeeds	Set the event severity level, from 1 to 40, to reflect the importance of an event in which discovery succeeds. The default is 25.
Event severity when discovery fails	Set the event severity level, from 1 to 40, to reflect the importance of an event in which discovery fails for any reason. The default is 5.
Event severity when discovery partially succeeds	Set the event severity level, from 1 to 40, to reflect the importance of an event in which discovery returns some data but also generates warning messages. The default is 10.
Event severity when discovery is not applicable	Set the event severity level, from 1 to 40, to reflect the importance of an event when discovery is not applicable, such as a situation in which there are no clustered applications on the servers on which you run this script. The default is 15.

3.3 Discovery_ReportAgent

Use this Knowledge Script to discover the NetIQ AppManager report agent and associated data sources.

There are some circumstances under which you must run the Discovery_ReportAgent Knowledge Script again:

- If you add a new data source (for example, Active Directory)
- If you install new applications for which AppManager provides application-specific reports (for example, if you install Oracle)

If you add a new data source, re-running Discovery_ReportAgent adds the data source as a child of the report agent, and displays the corresponding Report Knowledge Scripts in the Operator Console and Control Center.

If you install a new application (and discover it), re-running Discovery_ReportAgent adds the discovered application to the appropriate AppManager repository under the report agent, and displays the corresponding application-specific reports.

When you run Discovery_ReportAgent again, be sure to set the Knowledge Script to discover existing data sources. For example, if you previously discovered Active Directories, be sure to enable the *Discover Active Directories?* parameter. Failure to rediscover existing data sources will remove them from the report agent.

If you delete or add a resource object, or if you make any other kind of change that might affect the monitoring of your resources, run the Discovery_ReportAgent Knowledge Script again to update your list of resource objects. In addition, if you are running this module on AppManager 8 or higher, you can use the delta discovery feature in Control Center to run discovery on a schedule to more quickly detect changes to your environment.

Restrictions

This Knowledge Script is not supported in the Web Console.

You cannot use the Action_RunKS Knowledge Script to run Discovery_ReportAgent.

Resource Objects

Any computer with the AppManager report-enabled agent installed

Default Schedule

By default, this script is only run once for each computer.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Discovery Type	
Discover AppManager repository?	Set to **y** to discover the AppManager repository resource object. Successful discovery of this object allows you to render reports based on data in the AppManager repository. The default is y.
Discover Active Directories?	Set to **y** to discover Active Directory resource objects. Successful discovery of Active Directories allows you to render reports based on the data contained therein. The default is n.
Event Notification	
Raise event if discovery succeeds?	This Knowledge Script always raises an event when the job fails for any reason. In addition, you can set this parameter to y to raise an event when the job succeeds. The default is n.
Event severity when discovery succeeds	Set the event severity level, from 1 to 40, to reflect the importance for a successful discovery. The default is 25.
Event severity when discovery fails	Set the event severity level, from 1 to 40, to reflect the importance when the discovery fails. The default is 5.
Event severity when discovery is partially done	Set the event severity level, from 1 to 40, to reflect the importance when a discovery returns some data but also generates warning messages. The default is 10.
Event severity when discovery is not applicable	Set the event severity level, from 1 to 40, to reflect the importance when the discovery is not applicable. This type of failure usually occurs when the target computer does not have the AppManager report agent installed. The default is 15.

4 NT Knowledge Scripts

The NT category provides Knowledge Scripts for monitoring Microsoft Windows servers and workstations. This category also includes reporting scripts you can use to create meaningful reports about your Microsoft Windows servers and workstations.

From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. In the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
ConfigRemoteServiceDown	Loads the parameters specific to a local monitored computer and makes them available to the RemoteServiceDownLR Knowledge Script.
ConfigServiceDown	Loads the parameters specific to a local monitored computer and makes them available to the ServiceDownLR Knowledge Script.
CpuByProcess	Monitors CPU usage for each process and the total CPU usage for all processes.
CpuLoaded	Monitors total CPU usage and queue length to determine CPU load.
CpuResource	Monitors user CPU, the number of active processes, the number of threads, and the number of interrupts per second.
DiskSpace	Monitors logical drives for the percentage of disk space used, the amount of free space in megabytes, and the percentage of disk growth.
DNSConnectivity	Checks connectivity between a managed computer and its DNS server.
FailedLogon	Monitors the number of failed non-interactive logon attempts, for example, failed `net use` attempts, since the last interval.
FileChanged	Checks for changes to a specified file in the last monitoring interval.
FilesCompare	Compares the size, time stamp, and attributes of two files.
FileSizeSum	Monitors the total size of two specified files.
FilesOpen	Monitors the number of open files that were opened remotely, for example, by a user who remotely logged onto the computer.
FindFiles	Monitors logical drives for files that match your filtering criteria.
FolderFileCount	Monitors folders for the number of files that match your filtering criteria.
FolderSize	Monitors the size of folders containing files matching your filtering criteria.
IntervalCounter	Monitors the change in any performance monitor counter.

Knowledge Script	What It Does
LogicalDiskStats	Monitors logical disk transfers, disk reads, disk writes, operation time, and queue length.
MemByProcess	Monitors memory use for each process and total memory usage for all processes.
MemUtil	Monitors physical memory, virtual memory, and the paging files.
NetSession	Lists the network sessions connected to a computer.
NetworkBusy	Monitors the traffic on the network interface cards on a Windows computer.
PagingHigh	Monitors paging activity per second.
PhysicalDiskStats	Monitors physical disk transfers, disk reads, disk writes, operation time, and queue length.
PortHealth	Checks whether system ports are working properly.
PrinterHealth	Checks for print job problems, such as a paused or jammed printer, and the printer queue length.
PrinterQueue	Monitors a printer's queue length.
ProcessDown	Determines whether specified processes are running.
Processes	Monitors the number of processes.
ProcessUp	Checks whether a specified process is running and, optionally, terminates the process.
RegistryChange	Monitors changes in the registry.
RemoteServiceDown	Detects if any service on a remote computer is down.
RemoteServiceDownLR	Using parameters you specified with the ConfigRemoteServiceDown Knowledge Script, this script runs on a group of computers to detect whether services on remote computers are down.
Report_CPULoad	Generates a detailed report about CPU usage and queue length.
Report_CPULoadSummary	Generates a summary report about CPU usage and queue length.
Report_CPUResource	Generates a detailed report about the use of CPU resources, including the number of active processes, threads, and interrupts per second, and the utilization of CPU resources in user mode.
Report_CPUResourceSummary	Generates a summary report about the use of CPU resources.
Report_CPUUsageofProcessesSummary	Generates a summary report about CPU usage per named process, and total CPU usage by all named processes.
Report_FilesOpen	Generates a report about the number of files open during a specified period.
Report_LogicalDiskAvailSummary	Generates a summary report about the available space (in MB) for a logical disk.
Report_LogicalDiskUsageSummary	Generates a summary report about the percentage of disk space used and the amount of free space.

Knowledge Script	What It Does
Report_MemoryUtilization	Generates a detailed report about the use of physical and virtual memory, and paging files.
Report_MemoryUtilizationSummary	Generates a summary report about the use of physical and virtual memory, and paging files.
Report_NetworkBusy	Generates a report about the use of bandwidth on network interface cards.
Report_PagingHigh	Generates a report about the number of reads and writes per second to the page file.
Report_PhysicalDiskIO	Generates a report about the number of reads, writes, and transfers per second for a physical disk.
Report_PhysicalDiskQueueLength	Generates a report about physical disk queue length.
Report_PrinterHealth	Generates a report about print job problems and printer queue length.
Report_Process	Generates a report about the number of processes running during a specified period.
Report_TopCPUProcs	Generates a report about the total CPU used by all processes and which processes consume the most CPU resources.
Report_TopMemoryProcs	Generates a report about the total memory used by all processes and which processes consume the most memory.
RunAwayProcesses	Detects runaway processes by sampling CPU usage.
ServerBusy	Monitors the Windows server activity for network clients.
ServerBytes	Monitors the number of bytes per second transferred to and from a target computer.
ServerError	Monitors the number of sessions that errored out during the monitoring interval.
ServerTimeout	Monitors the number of sessions that timed out during the monitoring interval.
ServiceChange	Detects changes to the status and start type for Windows services.
ServiceDown	Monitors the stopped and started status of Microsoft Windows services and, optionally, starts services that are stopped.
ServiceDownLR	Using parameters you specified with the ConfigServiceDown Knowledge Script, this script can run on a group of computers to detect whether specified services are down and if so, optionally restart them.
ServiceHung	Checks whether any Windows services are hung.
ServiceRemove	Detects if any Windows services are added or removed in the monitoring interval.
SharedFiles	Monitors open network shared files.
SystemUpTime	Tracks the number of hours a computer has been operational since it was last rebooted.

Knowledge Script	What It Does
TopCpuProcs	Monitors total CPU used by all processes and which processes consume the most CPU resources.
TopMemoryProcs	Monitors the total memory used by all processes and which processes consume the most memory.
TrustRelationship	Tests the domain trust relationship from a trusting domain to specified trusted domains.
UnixRemoteProcessDown	Monitors applications on remote UNIX computers where you cannot easily install a UNIX agent.
BestPractices Knowledge Script Group	Collect and monitor the KPIs that will be shown in the NOC view for NT objects.

4.1 ConfigRemoteServiceDown

Use this Knowledge Script to set parameter values in the local repository of the computer where you run the script. The values are used by the RemoteServiceDownLR script when it runs on that computer.

Using this pair of scripts, you can set up the computers in a group so that when the RemoteServiceDownLR script runs on the group, it can run with different parameter values on each computer. This is particularly useful for enforcing monitoring policies.

Resource Objects

Windows 2003 Server and later

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event when job completes?	Set to **y** to raise an event when the job completes, with or without errors. The default is y.
List of computers	Specify one or more computers to monitor, separating each computer name with a comma (,) and no space.
Full path to file with a list of computers	Specify the full path to and name of a text file containing a list of computers. Put each computer on a separate line; no commas or spaces. The job supports a maximum file size of 32KB. If the file size exceeds 32KB, the job stops and raises an error event message: `Out of string space`.
List of services	Specify the name of one or more services to monitor, separating each name with a comma (,) and no space. The default is `NetIQmc`.

Description	How to Set It
Event severity when job completes	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the ConfigRemoteServiceDown job completes, with or without errors. The default is 25 (blue event indicator).

4.2 ConfigServiceDown

Use this Knowledge Script to set parameter values in the local repository of the computer on which you run the script. The values are used by the ServiceDownLR script when it runs on that computer. Using this pair of scripts, you can set up the computers in a group so that when the ServiceDownLR script runs on the group, it can run with different parameter values on each computer. This is particularly useful for enforcing monitoring policies.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event when job completes?	Select **Yes** to raise an event when the job completes, with or without errors. The default is Yes.
List of services	Specify one or more services to monitor. Use an asterisk (*) to monitor all “automatic” services. Use a comma to separate multiple service names. The default is `EventLog`.
List of excluded services	Specify one or more services to exclude from monitoring. Use an asterisk (*) to exclude all “automatic” services. Use a comma to separate multiple service names.
Event severity when job completes	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the ConfigServiceDown job completes, with or without errors. The default is 25 (blue event indicator).

4.3 CpuByProcess

Use this Knowledge Script to monitor whether specified processes have exceeded CPU thresholds. This script monitors CPU usage for each named process, as well as the total CPU usage for all named processes.

To determine CPU usage, this script checks the percentage of processor time that the threads for each process used to execute instructions.

If a process is not found, the script assumes that the process is not running, and reports zero as the CPU result.

NOTE

- This script does not detect invalid process names. If you enter an invalid process name, the script assumes that the process is not running, and reports zero as the CPU result.
- If the CPU usage for the named processes exceeds the threshold limit, an event is raised. However, this is not applicable for Windows System Idle Process. The System Idle Process indicates the percentage of idle CPU resources. If no applications are running, this process indicates a high idle capacity. The high percentage exceeds the threshold and raises an event indicating that the System Idle Process consumes high CPU resources. You can safely ignore this event message because the high percentage refers to the high idle capacity and not high CPU usage.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Create event for each process that exceeds the threshold?	Select **Yes** to raise an event if any individual process exceeds the CPU usage threshold you specify. The default is Yes.
Severity - Individual process CPU high	Set the event severity level, from 1 to 40, to indicate the importance of an event in which CPU usage exceeds the threshold. The default is 8 (red event indicator).
Create event if the sum of all processes exceeds the threshold?	Select **Yes** to raise an event if the CPU usage by all processes exceeds the threshold you specify. The default is Yes.
Severity - Total process CPU high	Set the event severity level, from 1 to 40, to indicate the importance of an event in which CPU usage for all processes exceeds the threshold. The default is 15 (yellow event indicator).
Severity - Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the CpuByProcess job fails unexpectedly. The default is 5 (red event indicator).
Data Collection	
Collect data for each process?	Select **Yes** to collect data for charts and reports, for each process you monitor. The default is unselected.

Description	How to Set It
Collect data for all processes?	Select **Yes** to collect data for charts and reports, for all processes you monitor. The default is unselected.
Monitoring	
Processes	Specify the names of the processes you want to monitor. Separate the names with commas (,) and no spaces.
Maximum threshold for CPU for each process	Specify the maximum CPU usage allowed for *each* monitored process before an event is raised. The default is 60%.
Maximum threshold for CPU for all processes	Specify the maximum CPU usage allowed for *all* monitored processes before an event is raised. The default is 95%.

4.4 CpuLoaded

Use this Knowledge Script to monitor total CPU usage and queue length to determine whether the CPU is overloaded. This script raises an event when CPU usage and CPU queue length values exceed the thresholds you set.

Resource Objects

CPU folder or any individual CPU icon (for multiprocessor systems)

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the CpuLoaded job fails. The default is 5 (red event indicator).

Description	How to Set It
Raise event if total system CPU exceeds threshold?	Select **Yes** to raise an event if total system CPU usage exceeds the threshold you set. The default is Yes. This script raises an event when the following occur: ◆ Total system CPU exceeds the threshold AND ◆ **Threshold - Maximum processor queue length** is exceeded if you enabled the `Use queue length in determining threshold crossings for events` parameter. When an event is raised, the event detail will contain the, top N using processes of the system processor usage from their Window counter values. If you select this parameter AND the **Use virtual machine performance counters if available?** parameter, the job will retrieve the total system CPU usage metric from the VMWare virtual machine's total processor usage counter, and individual process detail for the top processor-consuming processes will be included in event and datastream detail.
Event severity when total system CPU exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which system CPU usage exceeds the threshold. The default is 10 (red event indicator).
Raise event if any individual CPU exceeds threshold?	Select **Yes** to raise an event if CPU usage for any monitored server exceeds the usage threshold you set. The default is unselected. This script raises an event when the following occurs: ◆ Individual CPU exceeds the threshold AND ◆ *Threshold - Maximum processor queue length* is exceeded if you enabled the Use queue length in determining threshold crossings for events parameter.
Event severity when individual CPU exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which individual CPU usage exceeds the threshold. The default is 10 (red event indicator).
Monitoring	
Cap processor usage values at 100 percent?	Select **Yes** to make 100 (percent) the maximum value that will be stored as the overall and individual process usage datastream, and it will be used as the value for the overall or individual processor usage when that value is found to be over 100 (percent) This typically occurs in the VWMare virtual processor overall usage counter value when you selected the `Use virtual machine performance counters if available?` parameter. This caps the value, which is useful when reporting on AppManager data with other Micro Focus reporting products. The default is unselected.

Description	How to Set It
Use virtual machine performance counters if available?	Select **Yes** to monitor the total system CPU usage metric retrieved from the VMware performance counter. The default is Yes. If you select Yes for this parameter, and if on job iteration `1`, the VMWare counters are not present, that information is stored and subsequent iterations not attempt to use them. The virtual machine performance counter always collects and creates data detail for the top N processes that are using the CPU. **NOTE:** There are no VMWare virtual machine performance counters for individual process usage, the operating system counters still retrieve those values. **Important** VMware allows virtual machines to report more than 100% of its CPU, so if you select Yes for this parameter, you might see CPU utilization data that is greater than 100%. ◆ The %VM Processor Time counter value includes the % processor time for each virtual CPU plus 25% overhead for the virtual machine, so up to 125% could be returned for each processor. ◆ If your monitoring environment cannot tolerate % processor time values greater than 100%, deselect the parameter for using the virtual machine counters, or enable the **Cap processor usage values at 100 percent**? parameter.
Use queue length in determining threshold crossings for events	Select **Yes** for the queue length to be used to determine whether to raise an event for the total/system CPU usage by combining the overall processor usage value with the length of the processor queue. If the parameter is not selected, only the overall CPU usage threshold is used to raise events, if you selected the *Raise event if total system CPU exceeds threshold?* parameter. The default is Yes.
Number of processes to include in detail when total CPU threshold crossed	This value represents the number of processes that the % Processor Usage counter value will collect and include in event detail and datastream detail, if either of these parameters are respectively selected in the job configuration. To turn off inclusion of individual usage detail in events and datastreams, set this parameter to `0`.
Thresholds	
Threshold - Maximum total system CPU	Specify the maximum total system CPU usage allowed before an event is raised. The default is 95%.
Threshold - Maximum individual CPU	Specify the maximum individual CPU usage allowed before an event is raised. The default is 98%.
Threshold - Maximum processor queue length	Specify the maximum number of processes the CPU queue can contain before an event is raised. CPU queue length indicates how many processes are ready to run. The default is 2 processes.
Data Collection	

Description	How to Set It
Collect data for total system utilization?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the overall percentage of CPU time used. The default is unselected. The detail data contains information about the percentage of CPU usage, threshold for percentage of CPU usage, and top N using processes of the system processor usage and their Window counter values.
Collect data for individual processor utilization?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the percentage of CPU time used for each processor in one datastream per processor. The default is unselected. The detail data contains information about the percentage of CPU usage and the threshold for percentage of CPU usage.
Collect data for processor queue length?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the number of threads waiting to execute on all processors. The default is unselected. The detail data contains information about processor queue length and the threshold for processor queue length.

Example of How this Script Is Used

This script monitors both the percentage of CPU used and processor queue length. By itself, high CPU usage might not indicate a problem. Instead, consider the following factors:

- Queue length
- How you are using the computers monitored
- Your overall strategy for the environment

For example, in a **transactional** environment you can have a computer with CPU usage at 90% consistently. The computer has no room for growth, but if the queue length remains low and stable (never more than two or three threads waiting), the computer can be sized perfectly for maximum efficiency. If the queue length increases and threads are waiting, you may have a problem that needs to be addressed.

In a **batch** environment, however, you can set the script to run during off-peak hours when the batch jobs are not running. The script can raise an event if CPU usage is over 50% and any thread is waiting (queue length at 0) to ensure the computer has enough CPU headroom for batch jobs to run.

Other factors to consider are long range plans, such as the number of users you expect to support, how long you expect to support them, and how much room you need for growth. For example, you can set the CPU usage threshold lower to warn you to off-load some processing or order new systems.

Monitoring Multi-Processor Systems

On a multi-processor system, the total CPU utilization is the average percentage of time that all the processors on the system are busy executing non-idle threads. For example:

- If all processors are always busy, this is 100%.
- If all processors are 50% busy, this is 50%.
- If 25% of the processors are busy, this is 25%.

Monitoring Overall or Individual CPU Load

Monitor load for each CPU individually to gain more specific information about what is really happening on a system. For example, if you monitor overall load and see CPU usage is 100%, you do not know as much about the resource usage as seeing that CPU 0 is running at 90% and CPU 1 is running at 10%.

Handling Spikes

Because CPU and queue length are often subject to temporary spikes, set a short interval (two to five minutes), but raise an event only after thresholds are exceeded in three consecutive periods.

Collecting Data for Trend Analysis

This script can be set to collect data to help you identify usage trends for your servers. For example, if CPU usage increases, you can plan for growth. To perform this type of analysis, run a second job that collects data at a less-frequent interval.

4.5 CpuResource

Use this Knowledge Script to monitor CPU resource consumption for users, the number of active processes, the number of threads, and the number of interrupts per second. This script raises an event if a monitored value exceeds the threshold you specify.

Resource Object

CPU folder

Default Schedule

The default schedule for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if any threshold is exceeded. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. The default is n.
Maximum CPU user utilization threshold	Specify the maximum CPU usage allowed in user mode before an event is raised. The default is 90%.
Maximum number of processes threshold	Specify the maximum number of processes that can be running before an event is raised. The default is 80.
Maximum number of threads threshold	Specify the maximum number of threads that can be running before an event is raised. The default is 500.
Maximum Interrupts per second threshold	Specify the maximum number of interrupts per second allowed before an event is raised. The default is 600.

Description	How to Set It
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default is 5 (red event indicator).

4.6 DiskSpace

Use this Knowledge Script to monitor logical drives for the percentage of disk space used, the amount of free space in megabytes, and the percentage of disk growth between iterations.

Each time it runs, this script automatically monitors all logical disks on a server and all shared drives in a cluster. The owner of the quorum disk, which determines the current state of the cluster, monitors the space on shared drives. You can override automatic monitoring by providing a specific list of drives to monitor. Also, you can provide a list of drives to exclude from monitoring.

This script raises an event if the percentage of used space exceeds the threshold you set, if the amount of free space falls below the threshold you set, or if the percentage of disk growth exceeds the threshold you set.

NOTE

- In a cluster, this script runs on the primary node to monitor disk growth. If failover occurs, this script begins monitoring disk growth on the secondary node. The first iteration of the Knowledge Script job after failover reports the disk size of the secondary node as being the same as that of the primary node. The job begins reporting disk growth during the second and subsequent iterations after failover.
- Because clustered virtual servers do not support maintenance mode, the *Maintenance Mode* option is unavailable for clustered virtual servers in AppManager.

The ReportAM_CurrentDiskSpaceUsage report uses data collected by the NT_DiskSpace script. Ensure you archive data detail when running the Knowledge Scripts to collect data. You must disable the *Do not archive data detail* option in the Advanced tab of the Knowledge Script properties dialog box to allow automatic data archiving.

To use the DiskSpace Knowledge Script, you must install the AppManager for Microsoft Windows module, version 7.6.x.0 or later, on your agent computers.

Resource Object

Logical disk object

Default Schedule

By default, this script runs every five minutes.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity when job fails?	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the DiskSpace job fails. The default is 5.
Raise an event if threshold is crossed?	Select **Yes** to raise an event if the amount of available disk space falls below the threshold you set, if the percentage of disk utilization exceeds the threshold you set, and if the percentage of disk growth exceeds the threshold you set. The default is Yes. When you enable this parameter, the script raises one event that details the disk usage for all monitored logical drives. **NOTE:** If you run this script on a cluster, the script raises one event per monitored node in that cluster.
Raise a separate event for individual drives?	Select **Yes** to raise an event if the amount of available disk space falls below the threshold you set or the percentage of disk utilization exceeds the threshold you set. The default is unselected. When you enable this parameter, the script raises separate events that detail the disk usage for each monitored logical drive.
Event severity when a threshold is crossed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the amount of available disk space falls below the threshold you set or the percentage of disk utilization exceeds the threshold you set. The default is 5.
Do not raise events if disk growth thresholds are crossed?	Select **Yes** to limit the number of events that occur by not raising an event any time the disk growth thresholds are crossed. The default is unselected.
Only raise events when both disk space and disk utilization thresholds are crossed?	Select **Yes** to raise an event only when *both* disk space and disk utilization thresholds are crossed. The default is unselected.
Use XML format for event message?	Select **Yes** to format event detail messages in XML. Leave this parameter unselected to format event detail messages in plain text. Events formatted in XML display results in tables. Events in plain text display results in rows of unformatted text.
Disk Space Monitoring	
Drives to monitor	To monitor network drives and override the automatic monitoring, provide the UNC paths to the logical drives you want to monitor. Separate more than one path with a comma, such as `\\server01\C$,\\server02\D$` **NOTE:** The Log On As account under which the AppManager agent runs must have permission to access the UNC path. Leave this parameter blank to automatically monitor all logical drives.

Parameter	How to Set It
Drives to exclude	Provide a comma-separated list of the drives you do not want to monitor. This script automatically monitors all drives except those listed in this parameter. You can use regular expressions in this parameter. For more details about regular expressions, see the following Microsoft web pages: ◆ http://msdn.microsoft.com/en-us/library/6wzad2b2.aspx ◆ http://msdn.microsoft.com/en-us/library/1400241x.aspx You can also use the asterisk (*) as a wildcard at the end of a string.
Use volume label in Drives to exclude?	Select **Yes** to use volume label in the Drives to exclude. The default is unselected. **TIP:** To configure a drive (or volume) label, right-click the drive in Windows File Explorer, select Properties, then select the General tab. In the parameters for NT_DiskSpace, select the **use volume label**, then specify the drive label. This is in place of using the drive letter (e.g. “E:”)
Monitor mount points?	Select **Yes** to allow the script to monitor mount points (mapped drives). The default is Yes.
Ignore disks with minimal total size?	Select **Yes** to exclude disks of a certain size from monitoring. Use the *Threshold - Minimum disk size for monitoring* parameter to set the minimum monitoring requirement. The default is Yes.
Minimum size for disk monitoring	Specify the minimum size requirement for disk monitoring. Disks of less than *n* MB are excluded from monitoring The default is 100 MB.
Thresholds	
Global threshold - Minimum available disk space	Specify the minimum amount of disk space that must be available to prevent an event from being raised. The default is 100 MB. This threshold applies to all disks unless you provide a per-disk threshold value in the *Per-disk threshold - Minimum available disk space* parameter.
Global threshold - Maximum disk utilization	Specify the maximum percentage of disk utilization that can occur before an event is raised. The default is 90%. This threshold applies to all disks unless you provide a per-disk threshold value in the *Per-disk threshold - Maximum disk utilization in %* parameter.
Global threshold - Maximum percentage of disk growth	Specify the percentage of disk growth that can occur before an event is raised. The default is 25%. For example, if you set this parameter to 30%, this script raises an event if the size of the disk is 30% larger than it was during the previous script iteration. This threshold applies to all disks unless you provide a per-disk threshold value in the *Per-disk threshold - Maximum percentage of disk growth in %* parameter.
Apply per disk thresholds?	Select **Yes** to set different thresholds for individual disks. The default is unselected. **NOTE:** If you are monitoring mount points, label them in the following manner: `C:\MOUNT=100000,D:\MOUNT=90000`

Parameter	How to Set It
Per-disk threshold - Minimum available disk space in MB	Specify the minimum amount of disk space that must be available on individual disks to prevent an event from being raised. Use commas to separate multiple thresholds. For example: `C=90500,D=550` In this example, the threshold for minimum disk space on the C: disk is 90500 MB. The threshold for the D: disk is 550 MB.
Per-disk threshold - Maximum disk utilization in %	Specify the maximum percentage of disk utilization that can occur on individual disks before an event is raised. Use commas to separate multiple thresholds. For example: `C=50,D=80` In this example, the threshold for maximum disk utilization on the C: disk is 50%. The threshold for the D: disk is 80%.
Per-disk threshold - Maximum percentage of disk growth in %	Specify the maximum percentage of disk growth that can occur on individual disks before an event is raised. Use commas to separate multiple thresholds. For example: `C=5,D=10` In this example, the threshold for maximum disk growth on the C: disk is 5%. The threshold for the D: disk is 10%.
Data Collection	
Collect data for available disk space?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the amount of available disk space for the selected drives. The default is unselected.
Collect data for disk utilization?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns utilization details for logical disk space (%), used space (%), threshold (%), total space (MB), free space (MB). The default is unselected.
Collect data for disk growth?	Select Yes to collect data for charts and reports. If enabled, data collection returns the percentage of disk growth from the previous iteration to the current iteration. The default is unselected.

4.7 DNSConnectivity

Use this Knowledge Script to check connectivity between a managed computer and its DNS server. This script raises an event if the connection to the DNS server fails.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every hour**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the connection to the DNS server fails. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: - **100** -- the DNS lookup and connection to the DNS server were successful, or - **0** -- the connection was not successful. The default is n.
Remote DNS host name	Specify the name of the DNS server whose connection should be checked. If you do not enter a hostname, the default DNS server for the managed computer is used. The default is `wns1.HOME.net`.
DNS domain name	Provide the name of the DNS domain for the specified DNS server. Leave blank to use the local domain for the managed computer. The default is `HOME.net`.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the connection to the DNS server fails. The default is 8 (red event indicator).

4.8 FailedLogon

Use this Knowledge Script to monitor the number of failed non-interactive logon attempts to the server since the last interval. The result is always zero for the first interval so that the script can establish a baseline for subsequent checks.

For example, this script raises an event if you run this script on a computer and unsuccessfully attempt to log onto that computer using the `net use` command. This script does **not** raise an event for a failed interactive logon attempt, even a failed interactive login attempt from a remote desktop.

Use this script to determine whether password guessing programs are being used on the server. If you use this script to monitor events, the script raises an event for each failed logon attempt. If you choose to collect data, the script reports the total number of logon failures.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every hour**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the number of failed logon attempts exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the total number of logon failures. The default is n.
Failed logon threshold	Specify the maximum number of failed logon attempts allowed before an event is raised. The default is 0. If you are seeing too many insignificant events from users entering passwords incorrectly, determine a “typical” logon failure pattern (for example 5 per 24 hours) and set this parameter accordingly.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of failed logon attempts exceeds the threshold. The default is 5 (red event indicator).

4.9 FileChanged

Use this Knowledge Script to determine whether a specified file has changed since the last monitoring interval. This script compares the current size, time stamp, and attributes for a file to the size, time stamp, and attribute settings found for the file the last time the script ran.

You can choose to raise an event if the size, time stamp, or attribute indicates the file has been modified, or raise an event if any of these properties indicates the file has *not* been changed since the last monitoring interval.

Because this script checks the file properties rather than the file content, you can use this script with almost any file type.

Because this script can raise an event if a particular file has changed or when a file you expect to change has not been modified, you can use the script many different ways to monitor your environment.

For example, you might have an application that runs nightly regression tests and generates a report of the results. You can use this script to raise an event when the time stamp for the regression report is not modified, indicating that the test harness might have failed or other problems occurred in producing the expected report. If no event is raised, you can assume that a new report was generated successfully.

In addition, because you can selectively monitor file size, modification time, and attributes, and set severity levels for these properties independently, you can get clearer insight into the changes made to key files and respond accordingly. For example, you can monitor the file modification time for a file and receive a warning or raise an informational event when this property changes. You can raise a critical severity event or receive an e-mail message if a file’s attribute changes to read-only or suddenly becomes writable.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 24 hours**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Create event if file size changed?	Select **Yes** to raise an event if the file size has changed since the last time the job ran. The default is unselected.
Severity - Size changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the file size has changed. The default is 15 (yellow event indicator).
Create event if file time changed?	Select **Yes** to raise an event if the modification time of the file has changed since the last time the job ran. The default is Yes.
Severity - Time changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the modification time has changed. The default is 15 (yellow event indicator).
Create event if file attribute changed?	Select **Yes** to raise an event if an attribute of the file has changed since the last time the job ran. The default is unselected.
Severity - Attribute changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the file attribute has changed. The default is 15 (yellow event indicator).
Create event if file does not exist?	Select **Yes** to raise an event if the file you want to monitor does not exist. By default, events are not raised.
Severity - File does not exist	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the file does not exist. The default is 15 (yellow event indicator).
Severity - Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the FileChanged job fails unexpectedly. The default is 5 (red event indicator).
Monitoring	
File path	Specify the full path to the file you want to monitor. For example: `C:\Temp\myfile.txt`
Monitor file for...	**...changes**. Select this option to raise events when there are changes to the file. **...no changes**. Select this option to raise events when there are no changes to the file.

4.10 FilesCompare

Use this Knowledge Script to compare the sizes, time stamps, and attributes of two files. You can choose which properties to compare and the event severity if the script finds differences between the specified properties.

Because this script checks the file properties rather than the file content, you can use this script with almost any file type.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event if sizes are different?	Set to **y** to raise an event if the file size of File #1 is different from the file size for File #2. The default is y.
Raise event if modification times are different?	Set to **y** to raise an event if the file modification time for File #1 is different from the file modification time for File #2. The default is y.
Raise event if attributes are different?	Set to **y** to raise an event if the file attributes for File #1 are different from the file attributes for File #2. The default is y.
Compare file #1	Provide the full path to the first file to compare. For example: `C:\Temp\myfile.doc`.
Compare file #2	Provide the full path to the second file to compare.
Event severity level for size difference	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a size difference exists. The default is 5 (red event indicator).
Event severity level for time difference	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a time difference exists. The default is 5 (red event indicator).
Event severity level for attribute difference	Set the event severity level, from 1 to 40, to indicate the importance of an event in which an attribute difference exists. The default is 5 (red event indicator).

4.11 FileSizeSum

Use this Knowledge Script to monitor the total size of two files. This script raises an event if the total size of the two files exceeds the threshold you set.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Create event if the sum of the size of both files is above the threshold?	Select **Yes** to raise an event if the total size of both files exceeds the threshold you set. The default is Yes.
Severity - Sum size above threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the size of both files exceeds the threshold. The default is 15 (yellow event indicator).
Create event if any file is missing?	Select **Yes** to raise an event if either of the specified files is missing. The default is Yes.
Severity - File missing	Set the event severity level, from 1 to 40, to indicate the importance of an event if either of the specified files is missing. The default is 15 (yellow event indicator).
Severity - Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the FileSizeSum job fails unexpectedly. The default is 5 (red event indicator).
Data Collection	
Collect file size sum data?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the total size of the two files you specify. The default is unselected.
Monitoring	
Select first file...	Click **Browse [...]** to select the first of the two files to monitor.
Select second file...	Click **Browse [...]** to select the second of the two files to monitor.
Sum size threshold	Specify the maximum file size allowed before an event is raised. The default is 2. **NOTE:** Select units for the file size in the *File size scale* parameter.
File size scale	Select the unit of file size. The choices are: ◆ bytes ◆ kilobytes ◆ megabytes ◆ gigabytes ◆ terabytes The default is megabytes.

4.12 FilesOpen

Use this Knowledge Script to monitor the number of files currently open through a shared network drive or by a user who logged onto the computer remotely, for example, by using the `net use` command. This script does *not* raise an event if a file is opened by a user who interactively logged onto the computer.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the number of open files exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. The default is n.
Maximum number of files open threshold	Specify the maximum number of files that can be open before an event is raised. The default is 200.
Event severity level for open files	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of open files exceeds the threshold. The default is 5 (red event indicator).
Event severity level for an unexpected Knowledge Script error	Set the event severity level, from 1 to 40, to indicate the importance of an event in which FilesOpen job fails unexpectedly. The default is 35 (magenta event indicator).

4.13 FindFiles

Use this Knowledge Script to monitor the number of files that match a set of criteria. This script raises an event if the number of matching files exceeds the threshold you specify. This job fails if the time required to find a file exceeds the schedule interval.

Resource Object

Misc Device folder

Default Schedule

The default schedule for this script is **Every 24 hours**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	

Description	How to Set It
Raise event if threshold is exceeded?	Select **Yes** to raise an event if the number of files found that match your criteria exceeds the threshold you specify. The default is Yes.
Severity -- Exceeded threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of matching files exceeds the threshold. The default is 11 (yellow event indicator).
Include filename matches in event detail?	Select **Yes** to enable the inclusion of filenames and their paths matching the job criteria in the event detail. This parameter allows the inclusion of filenames and their paths matching the job criteria to be included in the event detail. This should be disabled when the files matching the job criteria exceeds a few thousand, as the event detail created with file matches in the detail can exceed a size AppManager can properly display. This parameter is selected by default.
Raise event if a folder cannot be accessed?	Select **Yes** to raise an event if the folder you specify in the *Root folder to begin the search* parameter does not exist or cannot be accessed because the account under which the NetIQ Client Resource Monitor service (`NetIQmc`) is running does not have permission to open the folder. The default is unselected.
Severity - Folder not accessible	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the specified folder does not exist or cannot be accessed. The default is 25 (blue event indicator).
Severity - Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the FindFiles job fails unexpectedly. The default is 5 (red event indicator).
Data Collection	
Collect file count data?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the number of files that match your filtering criteria. The default is unselected.
Include filename matches in data detail?	Select **Yes** to enable the inclusion of filenames and their paths matching the job criteria in the data detail when a datastream is collected. This parameter allows the inclusion of filenames and their paths matching the job criteria to be included in the data detail when data collection is enabled. This should be disabled when the files matching the job criteria exceeds a few thousand, as the data detail created with file matches in the detail can exceed a size AppManager can properly display. This parameter is selected by default.
Monitoring	
Logical drive letter(s) to search	Provide a comma-separated list, with no spaces, of the letters representing the logical drives you want to search. For example: `C,D`. Leave this field blank to specify a UNC (Universal Naming Convention) path from which the script will start monitoring files and specify the root UNC path in the **Root folder to begin the search** parameter.
Root folder to begin the search	Provide the path to the folder on each drive or specify the root Universal Naming Convention (UNC) path where the search should begin. For example, enter `Documents and Settings\Administrator\My Documents` to begin searching in the `My Documents` folder The default is `users`.

Description	How to Set It
File name(s), can use * and ? wildcards	Provide the filenames to search for. Use the `*` wildcard to represent any number of characters. Use the ? wildcard to represent any single character. The default is `*.*` (all files). You can enter only one filename here, but with the use of wildcards, you can search for multiple files.
Search subfolders?	Select **Yes** to search any subfolders of the root folder in which your search begins. The default is unselected.
File count threshold	Specify the maximum number of files that can be found that match your criteria before an event is raised. The default is 500.
File Filters	
File Attributes Filter	
File attribute operator	Select the operator (AND or OR) to apply to the criteria in the *File Attributes Filter* parameters. For example, instruct the script to search for files that have the *Archive attribute* AND the *Hidden attribute*. The default is OR.
Archive attribute	Select **Yes** to search for files that have the Archive attribute. The default is Yes.
Hidden attribute	Select **Yes** to search for files that have the Hidden attribute. The default is Yes.
Read-only attribute	Select **Yes** to search for files that have the Read-only attribute. The default is Yes.
System attribute	Select **Yes** to search for files that have the System attribute. The default is Yes.
Date Modified Filter	
Apply date modified filter?	Select **Yes** to apply a search filter that considers the date on which a file was modified. The default is unselected.
Select time range	Click **Browse [...]** to open the time browser. Set a specific or sliding date/time range for the date/time on which files were modified. The default is a sliding range of 1 Day with the End now option selected. A specific date/time range defines a specific start and end date and time, for example: `1/1/2004 12:00 AM to 1/31/2004 11:59 PM.` A sliding date/time range defines a time range relative to the start time of the Knowledge Script job. For example, a sliding date/time range of 1 Day extends from 12:00 AM of the previous day to 11:59 PM of the previous day (the entire 24-hour period of the day prior to the day the script runs). The **End now** option for the sliding date/time range extends the time range up to the start time of the Knowledge Script job. For example, if the job runs at 3:00 PM with a sliding range of 1 Day, then the time range covered is 12:00 AM of the previous day to 3:00 PM of the current day.
File Size Filter	
Apply file size filter?	Select **Yes** to apply a search filter that considers the size of a file. The default is unselected.
File size	Set the number of units that define the file size. The type of units are set in the *File size scale* parameter. The default is 2.
File size scale	Set the type of unit that defines the file size (for example, kilobytes). The default is megabytes.

Description	How to Set It
File size operator	Select the operator that defines the file size (for example, less than 2 megabytes). The default is greater than.

4.14 FolderFileCount

Use this Knowledge Script to monitor the number of files in a folder that match a set of criteria. This script raises an event if the number of matching files per folder exceeds the threshold you specify.

Resource Object

Misc Device folder

Default Schedule

The default schedule for this script is **Every 24 hours**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Raise event if threshold is exceeded?	Select **Yes** to raise an event if the number of files found matching your criteria exceeds the threshold you specify. The default is Yes.
Severity - Exceeded threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of matching files exceeds the threshold. The default is 11 (yellow event indicator).
Raise event if a folder cannot be accessed?	Select **Yes** to raise an event if the folder you specify in the *Root folder to begin the search* parameter does not exist or cannot be accessed because the account under which the NetIQ Client Resource Monitor service (`NetIQmc`) is running does not have permission to open the folder. The default is unselected.
Severity - Folder not accessible	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the specified folder does not exist or cannot be accessed. The default is 25 (blue event indicator).
Severity - Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the FolderFileCount job fails unexpectedly. The default is 5 (red event indicator).
Data Collection	
Collect file count data?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the number of folders with a file count that exceeds your threshold and the number of files per folder that match your filtering criteria. The default is unselected.
Monitoring	

Description	How to Set It
Logical drive letter(s) to search	Type a comma-separated list, with no spaces, of the letters representing the logical drives you want to search. For example: `C,D`.
Root folder to begin the search	Type the path to the folder on each drive where you want to begin searching. For example, `Documents and Settings\Administrator\My Documents` to begin searching in the `My Documents` folder. The default is `users`.
File name(s), can use * and ? wildcards	Type the filenames to search for. Use the `*` wildcard to represent any number of characters; use the `?` wildcard to represent any single character. The default is `*.*` (all files). You can enter only one filename here, but with the use of wildcards, you can search for multiple files.
Search subfolders?	Select **Yes** to search any subfolders of the root folder in which your search begins. The default is unselected.
File count threshold per folder	Specify the maximum number of files per folder that can be found that match your criteria before an event is raised. The default is 500.
File Filters	
File Attributes Filter	
File attribute operator	Select the operator (AND or OR) to apply to the criteria in the *File Attributes Filter* parameters. For example, instruct the script to search for files that have the *Archive attribute* AND the *Hidden attribute*. The default is OR.
Archive attribute	Select **Yes** to search for files that have the Archive attribute. The default is unselected.
Hidden attribute	Select **Yes** to search for files that have the Hidden attribute. The default is unselected.
Read-only attribute	Select **Yes** to search for files that have the Read-only attribute. The default is unselected.
System attribute	Select **Yes** to search for files that have the System attribute. The default is unselected.
Date Modified Filter	
Apply date modified filter?	Select **Yes** to apply a search filter that considers the date on which a file was modified. The default is unselected.

Description	How to Set It
Select time range	Click **Browse [...]** to set a specific or sliding date/time range for the date/time on which files were modified. The default is a sliding range of 1 Day with the End now option selected. A specific date/time range defines a specific start and end date and time, for example: 1/1/2004 12:00 AM to 1/31/2004 11:59 PM. A sliding date/time range defines a time range relative to the start time of the Knowledge Script job. For example, a sliding date/time range of 1 Day extends from 12:00 AM of the previous day to 11:59 PM of the previous day (the entire 24-hour period of the day prior to the day the script runs). The **End now** option for the sliding date/time range extends the time range up to the start time of the Knowledge Script job. For example, if the job runs at 3:00 PM with a sliding range of 1 Day, then the time range covered is 12:00 AM of the previous day to 3:00 PM of the current day.
File Size Filter	
Apply file size filter?	Select **Yes** to apply a search filter that considers the size of a file. The default is unselected.
File size	Specify the number of units that define the file size. The type of units are set in the *File size scale* parameter. The default is 10.
File size scale	Specify the type of unit that defines the file size (for example, kilobytes). The default is megabytes.
File size operator	Specify the operator that defines the file size (for example, less than 10 megabytes). The default is greater than.

4.15 FolderSize

Use this Knowledge Script to monitor the size of folders containing files that match a set of criteria. For example, you can monitor for folders over 100 MB that contain `MPG` or `JPG` files, and you can further refine your criteria to only include `MPG` or `JPG` files over a particular size. This script raises an event if the size of any folder exceeds the threshold you specify.

Resource Object

Misc Device folder

Default Schedule

The default schedule for this script is **Every 24 hours**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Raise event if threshold is exceeded?	Select **Yes** to raise an event if the size of a folder containing files matching your criteria exceeds the threshold you specify. The default is Yes.
Severity - Exceeded threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the folder size exceeds the threshold. The default is 11 (yellow event indicator).
Raise event if a folder cannot be accessed?	Select **Yes** to raise an event if the folder you specify in the *Root folder to begin the search* parameter does not exist or cannot be accessed because the account under which the NetIQ Client Resource Monitor service (`NetIQmc`) is running does not have permission to open the folder. The default is unselected.
Severity - Folder not accessible	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the specified folder does not exist or cannot be accessed. The default is 25 (blue event indicator).
Severity - Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the FolderSize job fails. The default is 5 (red event indicator).
Data Collection	
Collect folder count data?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the number of folders whose file size exceeds the threshold. The default is unselected.
Monitoring	
Logical drive letter(s) to search	Type a comma-separated list, with no spaces, of the letters representing the logical drives you want to search. For example: `C,D`.
Root folder to begin the search	Type the path to the folder on each drive in which you want to begin searching. For example, enter `Documents and Settings\Administrator\My Documents` to begin searching in the `My Documents` folder. The default is `users`.
File name(s), can use * and ? wildcards	Type the filenames to search for. Use the `*` wildcard to represent any number of characters; use the `?` wildcard to represent any single character. The default is `*.*` (all files). You can enter only one filename here, but with the use of wildcards, you can search for multiple files.
Search subfolders?	Select **Yes** to search any subfolders of the root folder in which your search begins. The default is Yes.
Folder size threshold	Specify the number of units that define the folder size. The type of units is set in the *Folder size scale* parameter. The default is 10.
Folder size scale	Specify the type of unit that defines the folder size (for example, kilobytes). The default is megabytes.
Folder size operator	Specify the operator that defines the folder size (for example, less than 10 megabytes). The default is greater than.

Description	How to Set It
File Filters	
File Attributes Filter	
File attribute operator	Select the operator (AND or OR) to apply to the criteria in the *File Attributes Filter* parameters. For example, instruct the script to search for files that have the *Archive attribute* AND the *Hidden attribute*. The default is OR.
Archive attribute	Select **Yes** to search for files that have the Archive attribute. The default is unselected.
Hidden attribute	Select **Yes** to search for files that have the Hidden attribute. The default is unselected.
Read-only attribute	Select **Yes** to search for files that have the Read-only attribute. The default is unselected.
System attribute	Select **Yes** to search for files that have the System attribute. The default is unselected.
Date Modified Filter	
Apply date modified filter?	Select **Yes** to apply a search filter that considers the date on which a file was modified. The default is unselected.
Select time range	Click **Browse [...]** to set a specific or sliding date/time range for the date on which files were modified. The default is a sliding range of 1 Day with the End now option selected. A specific date/time range defines a specific start and end date and time, for example: `1/1/2004 12:00 AM to 1/31/2004 11:59 PM.` A sliding date/time range defines a time range relative to the start time of the Knowledge Script job. For example, a sliding date/time range of 1 Day extends from 12:00 AM of the previous day to 11:59 PM of the previous day (the entire 24-hour period of the day prior to the day the script runs). The **End now** option for the sliding date/time range extends the time range up to the start time of the Knowledge Script job. For example, if the job runs at 3:00 PM with a sliding range of 1 Day, then the time range covered is 12:00 AM of the previous day to 3:00 PM of the current day.
File Size Filter	
Apply file size filter?	Select **Yes** to apply a search filter that considers the size of a file. The default is unselected.
File size	Specify the number of units that define the file size. The type of units are set in the *File size scale* parameter. The default is 5.
File size scale	Specify the type of unit that defines the file size (for example, kilobytes). The default is megabytes.
File size operator	Specify the operator that defines the file size (for example, less than 5 megabytes). The default is greater than.

4.16 IntervalCounter

Use this Knowledge Script to monitor changes in any performance monitor counter. You can specify a consecutive number of times that the *Counter delta value threshold* parameter must be exceeded before the script raises an event. This script automatically raises an event if it does not find the counter to monitor.

This script collects the counter value delta between script executions for the object\counter\instance you are monitoring. A negative counter value delta indicates that the counter value has decreased.

Prerequisites

Requirements for Windows Server 2012, Windows 8, Windows 7, Windows 2008 R2, and Windows 2008:

The Log On As account under which the AppManager agent runs for these Windows operating systems must be a domain account and belong to the Administrator local group.

Requirements for Windows Server 2003:

- The Log On As account under which the AppManager agent runs on Windows Server 2003 must belong to the Performance Monitor Users policy.
- If the Operator Console or Control Center is installed on Windows Server 2003, the user account under which the console application runs must belong to the Performance Monitor Users policy.

  To check the local policy:

  1. At a Command Prompt, type `gpedit.msc` and press `Enter`.
  2. In the Group Policy snap-in, double-click **Computer Configuration > Windows Settings > Security Settings > Local Policies > User Rights Assignment**.
  3. In the **Local Setting** column, ensure the appropriate user account belongs to the **Performance Monitor Users** policy.
- If the Operator Console or Control Center is installed on Windows Server 2003, the Remote Registry service on the console computer must be running. If the Remote Registry service is down when you attempt to configure this script by browsing counter information on the remote computer, the console displays an error message that indicates AppManager was unable to connect to the remote computer.

Requirements for Windows Vista:

If the Operator Console or Control Center is installed on Windows Vista, the Remote Registry service on the console computer must be running. If the Remote Registry service is down when you attempt to configure this script by browsing counter information on the remote computer, the console becomes unresponsive.

Resource Objects

Any discovered Windows computer or application server, such as Exchange Server, SQL Server, or Proxy Server

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Collect data?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the counter value delta between script executions for the object, counter, or instance you are monitoring. The default is n.
Event when over threshold?	Set to **y** to raise an event if the counter value exceeds the threshold. The default is y.
Counter delta value threshold	Specify the maximum allowed value for the difference between the counter value from the previous job iteration and the current job iteration. The default is 600.
Counter to monitor	Enter the object, counter, or instance name, or click **Browse [...]** to select the object, counter, and instances to monitor. The default is `Objects\Threads\`. **NOTE:** You can also start the System Monitor and click **Add [+]** in the toolbar. Use the format: `<object>\<counter>\<instance>\`. You can enter multiple instances, separated by commas. For example: `Process\% Privileged Time\mapisp32,mqsvc`. If the counter does not have an instance name, end the string with a backslash: `Process\% Privileged Time\`. If an instance is a parent of multiple instances (for example, if you have a logical disk 0 with partitions C: and D:), enter the complete instance name exactly as displayed in the Performance Monitor (for example, `"0 ==> C:"`).
Consecutive times	Specify the maximum number of consecutive times the counter must exceed the threshold before this script raises an event. The default is 1.
Event severity level - Over threshold	Set the event severity level, from 1 to 40 to indicate the importance of an event in which the counter value exceeds threshold. The default is 8 (red event indicator).
Event severity level...	Set the event severity level, from 1 to 40 to indicate the importance of an event in which the monitored counter or instance cannot be found. The default is 15 (yellow event indicator).

4.17 LogicalDiskStats

Use this Knowledge Script to monitor logical disk I/O and busy statistics gathered from performance counter values in Performance Monitor:

- Disk transfers per second
- Disk reads per second
- Disk writes per second
- Disk operation time in milliseconds
- Disk queue length

This script raises an event if a monitored value exceeds the threshold you specify.

Each time it runs, this script automatically monitors all logical disks on a server and all shared drives in a cluster. When this script runs on a cluster virtual server object, it monitors statistics for all shared drives that are active at the time. When this script runs on a physical Windows server, it monitors statistics for those drives that are not shared as part of a cluster, such as local fixed drives and removable drives.

NOTE: Because clustered virtual servers do not support maintenance mode, the *Maintenance Mode* option is unavailable for clustered virtual servers in AppManager.

You can choose to exclude any drive from monitoring or to monitor mount points configured on logical drives. Mount points must be configured as described in the following Microsoft Knowledge Base article: http://support.microsoft.com/kb/280297. Performance counter values are not created for mount points configured incorrectly. Therefore, this script raises an event when mounts points are not configured correctly, indicating an error when reading the disk.

This script ignores CD-ROMs, floppy drives, or other removable media whose size cannot be determined.

Resource Object

Logical disk object

Default Schedule

By default, this script runs every 30 minutes.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity when job fails	Set the event severity, from 1 to 40, to indicate the importance of an event in which the LogicalDiskStats job fails. The default is 15.
Logical Disk I/O	
Raise event if disk transfers exceed threshold?	Select **Yes** to raise an event if the number disk transfers per second exceeds the threshold you set. The default is Yes.
Event severity when disk transfers exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of the event in which the number of times data is transferred on a disk exceeds the threshold you set. The default is 8.
Threshold - Maximum disk transfers per second	Specify the maximum number of disk transfers that can occur per second before an event is raised. The default is 80.
Raise event if disk reads exceed threshold?	Select **Yes** to raise an event if the number of disk reads per second exceeds the threshold you set. The default is Yes.
Event severity when disk reads exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of the event in which the number of times data is read from a disk exceeds the threshold you set. The default is 8.
Threshold - Maximum disk reads per second	Specify the maximum number of disk reads that can occur per second before an event is raised. The default is 50.
Raise event if disk writes exceed threshold?	Select **Yes** to raise an event if the number of disk writes per second exceeds the threshold you set. The default is Yes.

Parameter	How to Set It
Event severity when disk writes exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of the event in which the number of times data is written to a disk exceeds the threshold you set. The default is 8.
Threshold - Maximum disk writes per second	Specify the maximum number of disk writes that can occur per second before an event is raised. The default is 50.
Logical Disk Busy	
Raise event if disk operation time exceeds threshold?	Select **Yes** to raise an event if the length of time per disk operation exceeds the threshold you set. The default is Yes.
Event severity when disk operation time exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the length of time it takes for a disk operation to complete exceeds the threshold you set. The default is 5.
Threshold - Maximum disk operation time	Specify the maximum length of time that a disk operation can take to complete before an event is raised. The default is 100 milliseconds.
Raise event if disk queue length exceeds threshold?	Select **Yes** to raise an event if the number of requests in the disk queue exceeds the threshold you set. The default is Yes.
Event severity when disk queue length exceeds threshold	Set the event severity, from 1 to 40, to indicate the importance of an event in which the number of requests in the disk queue exceeds the threshold you set. The default is 5
Threshold - Maximum disk queue length	Specify the maximum number of requests that can be in queue before an event is raised. The default is 1 request.
Monitoring	
Drives to exclude	Provide a comma-separated list of the drives you do not want to monitor. This script automatically monitors all drives except those listed in this parameter. The asterisk (*) is an acceptable wildcard. For example, to exclude the C: drive and disks or mount points that begin with `\crate`, specify the following: `c:,e:\crate*`
Monitor mount points?	Select **Yes** to allow the script to monitor mount points (mapped drives). The default is Yes.
Data Collection	
Logical Disk I/O	
Collect data for disk transfers per second?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the number of disk transfers per second for the monitoring period. The default is unselected.
Collect data for disk reads per second?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the number of disk reads per second for the monitoring period. The default is unselected.
Collect data for disk writes per second?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the number of disk writes per second for the monitoring period. The default is unselected.
Logical Disk Busy	

Parameter	How to Set It
Collect data for disk operation time?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the length of disk operations, in milliseconds, for the monitoring period. The default is unselected.
Collect data for disk queue length?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the number of requests in queue for the monitoring period. The default is unselected.

4.18 MemByProcess

Use this Knowledge Script to monitor process memory usage. This script monitors individual memory use for each specified process, and the total memory use for all specified processes. If a process is not found, this script assumes that the process is not currently running, and reports 0 as the memory result.

You can use this script to monitor multiple processes with the same name, such as the process spawned by each instance of `svchost.exe` running on the same computer.

NOTE

- This script does not detect invalid process names. If you type an invalid process name, the script assumes that the process is not running, and reports 0 as the result.
- This script raises an event if the memory usage for a named process exceeds the threshold, with one exception: the Windows System Idle process. The System Idle process indicates the percentage of idle CPU resources. If no applications are running, this process indicates a high idle capacity. The high percentage exceeds the threshold and raises an event indicating that the System Idle Process consumes high CPU resources. You can safely ignore this event message because the high percentage refers to the high idle capacity and not high CPU usage.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Create event for each process that exceeds the threshold?	Select **Yes** to raise an event if the memory for an individual process exceeds the threshold you specify. The default is Yes.

Description	How to Set It
Severity - Individual process memory high	Set the event severity level, from 1 to 40, to indicate the importance of an event in which individual process memory usage exceeds the threshold. The default is 8 (red event indicator).
Create event if the sum of all processes exceeds the threshold?	Select **Yes** to raise an event if the memory usage for all processes exceeds the threshold you specify. The default is Yes.
Severity -- Total process memory high	Set the event severity level, from 1 to 40, to indicate the importance of an event in which total process memory usage exceeds the threshold. The default is 15 (yellow event indicator).
Severity -- Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the MemByProcess job fails unexpectedly. The default is 5 (red event indicator).
Data Collection	
Collect data for each process?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns data for individual processes you are monitoring, including process name, memory utilization, and the memory utilization threshold you specified. The default is unselected. **NOTE:** If a process is not found, the script assumes that the process is not currently running, and reports 0 as the memory result.
Collect data for all processes?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns data for all processes you are monitoring, including the process count, total memory utilization, and the memory utilization threshold you specified. The default is unselected. **NOTE:** If a process is not found, the script assumes that the process is not currently running, and reports 0 as the memory result.
Monitoring	
Processes	Provide one or more process names, separated by commas (,) without spaces. The default is `explorer,lsass`. **NOTE:** Under circumstances where multiple instances of a process are running on a computer (for example, `svchost`), Windows adds a number to each successive instance of the process beginning with the second instance (for example, `svchost`, `svchost#1`, `svchost#2`). You can monitor each process instance by entering the process name, including the added number. **To see a list of distinct process names:** 1. From the Control Panel, double-click **Administrative Tools** and then double-click **Performance**. 2. Click **Add** (+). 3. From the Performance object list, select **Process**. 4. Click **Select instances from list** and then scroll to find the process names. 5. Click **Add**. **NOTE:** The value of the *Processes* parameter must be an exact match (including case-sensitivity) of the process name in the Performance Object field of PerfMon.
Maximum threshold for memory for each process	Specify the maximum amount of memory each process can use before an event is raised. The default is 20000. Use the *Memory scale* parameter to define the threshold scale.

Description	How to Set It
Maximum threshold for memory for all processes	Specify the maximum amount of memory all processes can use before an event is raised. The default is 32000. Use the *Memory scale* parameter to define the threshold scale.
Memory scale	Select the scale for the memory threshold you specify: bytes, kilobytes, megabytes, gigabytes, terabytes. The default is kilobytes.

4.19 MemUtil

Use this Knowledge Script to monitor usage of physical memory, virtual memory, and paging files. This script raises an event if the usage of a monitored item exceeds the threshold. In addition, this script generates datastreams for all monitored items.

Resource Objects

Physical memory object

Virtual memory object

Paging files folder

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the MemUtil job fails. The default is 5.
Raise event if physical memory usage exceeds threshold?	Select **Yes** to raise an event if physical memory usage exceeds the threshold you set. The default is Yes.
Event severity when physical memory usage exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which physical memory usage exceeds the threshold you set. The default is 5.
Raise event if virtual memory usage exceeds threshold?	Select **Yes** to raise an event if virtual memory usage exceeds the threshold you set. The default is Yes.
Event severity when virtual memory usage exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which virtual memory exceeds the threshold you set. The default is 5.

Description	How to Set It
Raise event if paging file usage exceeds threshold?	Select **Yes** to raise an event if paging file usage exceeds the threshold you set. The default is Yes.
Event severity when paging file usage exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which paging file usage exceeds the threshold. The default is 5.
Monitoring	
Use virtual machine performance counters if available?	Select **Yes** to monitor VMware performance counter data on virtual machines, if available, instead of physical host counters. The default is Yes. VMware performance counters do not provide virtual memory or paging file data. If you select Yes, only physical memory usage data is monitored.
List top N processes for physical usage	Set number of processes to monitor physical memory usage. Usage is shown in events and data.
Thresholds	
Threshold - Maximum physical memory usage	Specify the maximum percentage of physical memory that can be in use before an event is raised. The default is 90%.
Threshold - Maximum virtual memory usage	Specify the maximum percentage of virtual memory that can be in use before an event is raised. The default is 90%.
Threshold - Maximum paging file usage	Specify the maximum percentage of the paging file that can be in use before an event is raised. The default is 70%.
Data Collection	
Collect data for physical memory usage?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the percentage of physical memory usage during the monitoring period. The default is unselected.
Collect data for virtual memory usage?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the percentage of virtual memory usage during the monitoring period. The default is unselected.
Collect data for paging file usage?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the size of the paging file during the monitoring period. The default is unselected.

4.20 NetSession

Use this Knowledge Script to list the network sessions connected to a computer. This script raises an event if the number of sessions exceeds the threshold you specify. In addition, this script generates datastreams for the number of connected sessions.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every hour**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the number of sessions exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. The default is n.
Maximum connected sessions threshold	Specify the maximum number of sessions that can be connected at one time before an event is raised. The default is 100.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of sessions exceeds the threshold. The default is 8 (red event indicator).

4.21 NetworkBusy

Use this Knowledge Script to monitor the traffic on network interface cards (NICs). This script raises an event if the network interface's bandwidth utilization exceeds the threshold you specify. This script skips interface card number 1 as a loopback.

NOTE: This script uses the Network Interface Performance Monitor counter to perform its monitoring task. If this performance counter is not available, install the SNMP services to make the counter available.

Resource Objects

Non-WAN wrapper network interface cards

Network interface folder

Default Schedule

The default schedule for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the network interface's bandwidth utilization exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the percentage of network bandwidth in use. The default is n.
Ignore network interfaces with no bandwidth counter data?	Set to **y** to ignore network interfaces that have a bandwidth counter value of zero, which prevents the module from raising events on unplugged interfaces. The default is n.

Description	How to Set It
Maximum percentage network utilization threshold	Specify the maximum percentage of network bandwidth that can be in use before an event is raised. The default is 35%.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which bandwidth utilization exceeds the threshold. The default is 5 (red event indicator).
Network interfaces to exclude	List the display names of all active interfaces that you do not want to be monitored. Separate each interface name with a comma, without any spaces. You can use the asterisk (*) as a wildcard at the end of a string, along with regular expressions for interface names. The default is: `*Loopback*,*Pseudo*,*isatap*,*tunnel*`. These settings exclude the set of interfaces that are part of the operating system and do not map to actual network cards.

4.22 PagingHigh

Use this Knowledge Script to monitor reads and writes per second to the pagefile. This script raises an event if the number of reads and writes per second exceeds the threshold you specify. In addition, this script generates datastreams for the number of read and writes per second.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the number of reads and writes exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the number of reads and writes per second during the monitoring interval. The default is n.
Maximum pagefile activity per second threshold	Specify the maximum number of reads and writes to the pagefile allowed per second before an event is raised. The default is 200.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of reads and writes per second exceeds the threshold. The default is 5 (red event indicator).

4.23 PhysicalDiskStats

Use this Knowledge Script to monitor physical disk I/O and busy values:

- Disk transfers per second
- Disk reads per second
- Disk writes per second
- Disk operation time in milliseconds
- Disk queue length

This script raises an event if a monitored value exceeds the threshold you specify.

NOTE: Because clustered virtual servers do not support maintenance mode, the *Maintenance Mode* option is unavailable for clustered virtual servers in AppManager.

Resource Objects

Physical disk object

Default Schedule

By default, this script runs every 30 minutes.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Physical Disk I/O Notification	
Raise event if disk transfers exceed threshold?	Select **Yes** to raise an event if the number disk transfers per second exceeds the threshold you set. The default is Yes.
Event severity when disk transfers exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of the event in which the number of times data is transferred on a disk exceeds the threshold you set. The default is 8.
Raise event if disk reads exceed threshold?	Select **Yes** to raise an event if the number of disk reads per second exceeds the threshold you set. The default is Yes.
Event severity when disk reads exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of the event in which the number of times data is read from a disk exceeds the threshold you set. The default is 8.
Raise event if disk writes exceed threshold?	Select **Yes** to raise an event if the number of disk writes per second exceeds the threshold you set. The default is Yes.
Event severity when disk writes exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of the event in which the number of times data is written to a disk exceeds the threshold you set. The default is 8.

Parameter	How to Set It
Physical Disk Busy Notification	
Raise event if disk operation time exceeds threshold?	Select **Yes** to raise an event if the length of time per disk operation exceeds the threshold you set. The default is Yes.
Event severity when disk operation time exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the length of time it takes for a disk operation to complete exceeds the threshold you set. The default is 5.
Raise event if disk queue length exceeds threshold?	Select **Yes** to raise an event if the number of requests in the disk queue exceeds the threshold you set. The default is Yes.
Event severity when disk queue length exceeds threshold	Set the event severity, from 1 to 40, to indicate the importance of an event in which the number of requests in the disk queue exceeds the threshold you set. The default is 5
Raise event if job fails?	Select **Yes** to raise an event if the PhysicalDiskStats job fails. The default is Yes.
Event severity when job fails	Set the event severity, from 1 to 40, to indicate the importance of an event in which the PhysicalDiskStats job fails. The default is 15.
Data Collection	
Physical Disk I/O Data Collection	
Collect data for disk transfers per second?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the number of disk transfers per second for the monitoring period. The default is unselected.
Collect data for disk reads per second?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the number of disk reads per second for the monitoring period. The default is unselected.
Collect data for disk writes per second?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the number of disk writes per second for the monitoring period. The default is unselected.
Physical Disk Busy Data Collection	
Collect data for disk operation time?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the length of disk operations, in milliseconds, for the monitoring period. The default is unselected.
Collect data for disk queue length?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the number of requests in queue for the monitoring period. The default is unselected.
Monitoring	
Physical Disk I/O Monitoring	
Threshold - Maximum disk transfers per second	Specify the maximum number of disk transfers that can occur per second before an event is raised. The default is 80.
Threshold - Maximum disk reads per second	Specify the maximum number of disk reads that can occur per second before an event is raised. The default is 50.
Threshold - Maximum disk writes per second	Specify the maximum number of disk writes that can occur per second before an event is raised. The default is 50.
Physical Disk Busy Monitoring	

Parameter	How to Set It
Threshold - Maximum disk operation time	Specify the maximum length of time that a disk operation can take to complete before an event is raised. The default is 100 milliseconds.
Threshold - Maximum disk queue length	Specify the maximum number of requests that can be in queue before an event is raised. The default is 1 request.

4.24 PortHealth

Use this Knowledge Script to check whether system ports are working properly. This script raises an event if a port is not operating. In addition, this script generates datastreams for port availability.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a port is not operating. The default is 5 (red event indicator).
Raise event if port appears unavailable?	Select **Yes** to raise an event if a port is not operating. The default is Yes.
Event severity when port appears unavailable	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a port is not operating. The default is 5 (red event indicator).
Use monitored address as event source?	Select **Yes** to use the monitored address as the event source. The default is Yes.
Monitoring	
Network addresses to monitor	Provide the network addresses for which you want to check port availability. Separate the addresses with commas (,) and no spaces. Use the format: `<host_ID>:<port_number>`. The host ID can be a hostname or IP address. For example: `www.storm.com:8008,1.10.10.10:30`. The default is `www.netiq.com:80`.

Description	How to Set It
Network Protocol	Select the protocol to use to check the availability of the network address and port number specified in the **Network addresses to monitor** parameter. The default is TCP. The following is a list of values. - **TCP** - Select to test TCP connectivity to the target network address and port. - **UDP** - Select to test the ability of the target network address and port to receive UDP streams. When UDP is selected, a string of `Is anyone there?` is sent to the target host and port. Because no response is expected, the target network address and port is considered available if the send attempt does not encounter any errors.
Data Collection	
Collect data for port health?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns: - 100 - the port is operating properly - 0 - the port is not operating The default is No.

4.25 PrinterHealth

Use this Knowledge Script to monitor the health of printers. This script checks whether any specified printer is paused or if the printer queue length exceeds the threshold you specify, and raises an event if either condition exists. This script also raises an event if it finds general printing errors such as a printer that is out of paper or jammed.

You can set this script to discover printers dynamically each time it runs. If you discover printers dynamically, printers that were not discovered when you ran Discovery_NT are not reflected in the Navigation pane or the TreeView pane.

NOTE: To run this script successfully, avoid using special characters such as, /, -, and # when defining the printer name on the monitored computers. Also, if you run the Discovery_NT Knowledge Script and then delete a local or network printer, run Discovery_NT again.

Resource Objects

Printer folder, if dynamically enumerating printers. If you are not enumerating printers dynamically, you can run this script on the Printer folder or individual printer objects. You can run this script only on a local printer.

Default Schedule

The default schedule for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if a specified printer is paused, if the printer queue length exceeds the threshold you specify, or if the script finds general printing errors. The default is y. This script displays the following event messages when it detects a general printer error: Door Open, Paper Jam, Offline, No Paper, Toner Low, or Service Request.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of print jobs at each interval. The default is n.
Dynamically enumerate at each interval?	Set to **y** to dynamically observe connected printers at each monitoring interval. The default is y.
Maximum printer queue length threshold	Specify the maximum number of print jobs that can be waiting in the queue before an event is raised. The default is 10.
Maximum print job size threshold	Specify the maximum print job size allowed before an event is raised. The default is 200 KB.
Event severity level for printer paused	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the printer is not responding and no jobs are being processed. The printer is off-line, or out of paper or toner, for example. The default is 3 (red event indicator).
Event severity level for printer busy	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of jobs in the printer queue exceeds the threshold. The default is 5 (red event indicator).
Event severity level for job size threshold crossed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the size of the print job exceeds the threshold. The default is 1 (red event indicator).

4.26 PrinterQueue

Use this Knowledge Script to monitor a printer's queue length. This script checks the number of queued print jobs for a specified printer. You need to specify the name of the computer that serves the printer and the printer's share name. This script raises an event if the number of queued jobs exceeds the threshold you specify. In addition, this script generates datastreams for queue length.

NOTE

- Before running this script, ensure the AppManager agent service, `NetIQmc`, is set to run as a domain user account user in the same domain as, or a domain trusted by, the target computer. Use the **Services** Control Panel to identify an account for the service to run as.
- To run this Knowledge Script successfully, avoid using special characters such as **/**, **-**, and **#** when defining the printer name on the monitored computers.

  Also, if you run the Discovery_NT Knowledge Script and *then* delete a local or network printer, run Discovery_NT again.

Resource Objects

Windows 2003 Server or later

NOTE: This script is not supported on 64-bit systems.

Default Schedule

The default schedule for this script is **Every hour**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the number of queued jobs exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of queued print jobs at each interval. The default is n.
Printer server's hostname	Provide the name of the computer that serves the printer you want to monitor.
Server's share name for printer	Provide the name of the share used by the printer server for the printer you want to monitor.
Maximum print queue threshold	Specify the maximum number of print jobs that can be waiting in the queue before an event is raised. The default is 10.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of queued jobs exceeds the threshold. The default is 8 (red event indicator).

4.27 ProcessDown

Use this Knowledge Script to determine whether specified processes are running. This script raises an event if a specified process is not running. In addition, this script generates a datastream for process availability.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the process you selected for monitoring is not running. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns data for each named process, either: ◆ **100** -- the process is running; or ◆ **0** -- the process is not running. The default is n.
Processes	Provide one or more process names, separated by commas (,) and no spaces. Do not specify the file extension of the process. For example: `clock,tcpsvcs` **Tip** To monitor processes, AppManager retrieves the name of the performance counter instance associated with the process. If a PID (process identifier) is appended to the counter instance name, you do not need to indicate the PID in this parameter. Use an asterisk (*) instead. For example: ◆ Use an asterisk (*) after the process name to monitor all processes that begin with the string you provide. For example: `clock,tcpsvcs*` ◆ Use an asterisk (*) before the process name to monitor all processes that end with the string you provide. For example: `clock,*tcpsvcs`
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a selected process is not running. The default is 8 (red event indicator).

4.28 Processes

Use this Knowledge Script to monitor the number of active processes. This script raises an event if the number of active processes exceeds the threshold you specify. In addition, this script generates datastreams for active processes.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the number of active processes exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. The default is n.
Maximum total processes threshold	Specify the maximum number of active processes allowed before an event is raised. The default is 80.
Number of top processes to be displayed	Specify the number of top processes to display in the detail event or data message. Type 0 to display all processes. The default is 10.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of active processes exceeds the threshold. The default is 5 (red event indicator).

4.29 ProcessUp

Use this Knowledge Script to check whether a specified process is running. This script raises an event if the specified process is running, and can automatically terminate the process if you choose. In addition, this script generates datastreams for process status.

Resource Objects

Windows 2000 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Create event if process is found running?	Select **Yes** to raise an event if one or more of the processes you specified in the *Processes* parameter are running. The default is Yes.
Severity - Process running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a specified process is running. The default is 10 (red event indicator).
Create event if process is not running?	Select **Yes** to raise an event if a process you specified in the *Processes* parameter is not running. The default is unselected.
Severity - Process not running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a specified process is not running. The default is 10 (red event indicator).

Description	How to Set It
Create event if process is successfully terminated?	Select **Yes** to raise an event a process you specified in the *Processes* parameter is successfully terminated. The default is Yes. **NOTE:** If you set this parameter to **Yes**, also set the *Kill the running process?* parameter to **Yes**.
Severity - Process successfully terminated	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a process is successfully stopped. The default is 25 (blue event indicator).
Create event if process cannot be terminated?	Select **Yes** to raise an event if a process you specified in *Processes* cannot be terminated. The default is Yes. **NOTE:** If you set this parameter to **Yes**, also set the *Kill the running process?* parameter to **Yes**.
Severity - Failed to kill process	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a specified process cannot be terminated. The default is 10 (red event indicator).
Severity - Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the ProcessUp job fails unexpectedly. The default is 5 (red event indicator).
Data Collection	
Collect data for process status?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns: ◆ *n* - the number of specified processes that are running, or ◆ **0** - the process is not running. The default is unselected.
Monitoring	
Processes	Provide one or more process names, separated by commas (,) and no spaces. Specify a process name without an extension and use the following format for multiple instances of a process: `iexplore (first instance),iexplore#1 (second instance),iexplore#2 (third instance)`
Kill the running process?	Select **Yes** to automatically stop a specified process. If there are multiple instances of a specified process, all instances are stopped. The default is unselected.

4.30 RegistryChange

Use this Knowledge Script to monitor changes in the registry information on 32-bit and 64-bit Windows systems. This script raises an event if a key or value is added, deleted, or changed in the registry. In addition, this script generates datastreams for registry changes.

From a specified path, this script searches the registry for changes to registry keys and sub-keys. This information can be valuable in helping you understand the behavior and configuration of the computers you are monitoring, but it can also be expensive in terms of processing time. Because each registry level can contain many sub-keys to check, this script can require a significant period of time to run if you check two or three levels deep in the registry tree.

On 64-bit Windows systems, this script can be configured to monitor registry information for 32-bit or 64-bit programs. For example, to monitor changes to:

- The key that specifies what programs should be run at startup, set the value of the *Monitor 32-bit program registry keys on a 64-bit system?* parameter to **n**, and specify the following registry key:

  ```
  HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
  ```

- Keys associated with a 32-bit application such as AppManager, set the value of the *Monitor 32-bit program registry keys on a 64-bit system?* parameter to **y**, and specify the registry exactly as it would be specified on a 32-bit system. For example:

  ```
  HKEY_LOCAL_MACHINE\Software\NetIQ\AppManager\4.0
  ```

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 30 minutes**.

If you set this script to check sub-key levels, adjust the schedule. For example, if you are checking two or three sub-levels deep, set this script to run once a day during off-peak hours.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job Failure Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a job fails. The default is 5 (red event indicator).
Event Notification	
Raise event when a registry key or value is added, deleted, or changed?	Select **Yes** to raise an event when a registry key or value is added, deleted, or changed. The default is selected.
Event severity level for registry changes	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a change in the registry occurs. The default is 8 (red event indicator).
Registry Monitoring	
Registry Location	

Description	How to Set It
Registry Root	Type the registry root. Valid root options are: ◆ `HKEY_LOCAL_MACHINE` ◆ `HKEY_CLASSES_ROOT` ◆ `HKEY_CURRENT_USER` ◆ `HKEY_USERS` The default is `HKEY_LOCAL_MACHINE`.
Registry Path	Specify the path to the registry keys to monitor. The default path is `SYSTEM\CurrentControlSet\Services`. To specify the path to registry information for a 32-bit or 64-bit program, specify a path under `HKEY_LOCAL_MACHINE\Software`. Although the registry keys for 32-bit programs on a 64-bit system are stored under the `HKEY_LOCAL_MACHINE\Software\Wow6432Node` key, do **not** specify the `Wow6432Node` component of the path. Instead, specify the path without the `Wow6432Node` component, and set the value of the *Monitor 32-bit program registry keys on a 64-bit system?* parameter to `y`. **NOTE:** Use a specific path to the key you want to monitor. Any key can have many sub-levels, and the level specified by this path is always considered level 1.
Search Options	
Number of registry subtrees to search	Specify the number of registry subtrees to monitor, from 1 to 5, counting the path itself as level 1. The maximum number of key sub-levels you can monitor is 5. The default is 2.
Registry keys to exclude from monitoring (comma-separated, without spaces)	Specify the registry keys or values to exclude from monitoring. If you enter multiple registry paths, separate them by commas and do not include spaces. This allows for using a higher level registry path in the Registry Path parameter and directs the Knowledge Script to ignore certain sub-keys or values under this path. The registry keys or values to exclude should use the full registry path off of the root, for example: `SYSTEM\CurrentControlSet\Services\netiqmc\ErrorControl` In this example, the `ErrorControl` registry value is excluded from monitoring under the `netiqmc` service registry key.
Monitor 32-bit program registry hive on a 64-bit system?	On a 64-bit Windows system, select **Yes** to monitor registry information for 32-bit programs. The default is Yes. **Tip** To monitor registry information for 32-bit programs and 64-bit programs, configure separate Knowledge Script jobs.

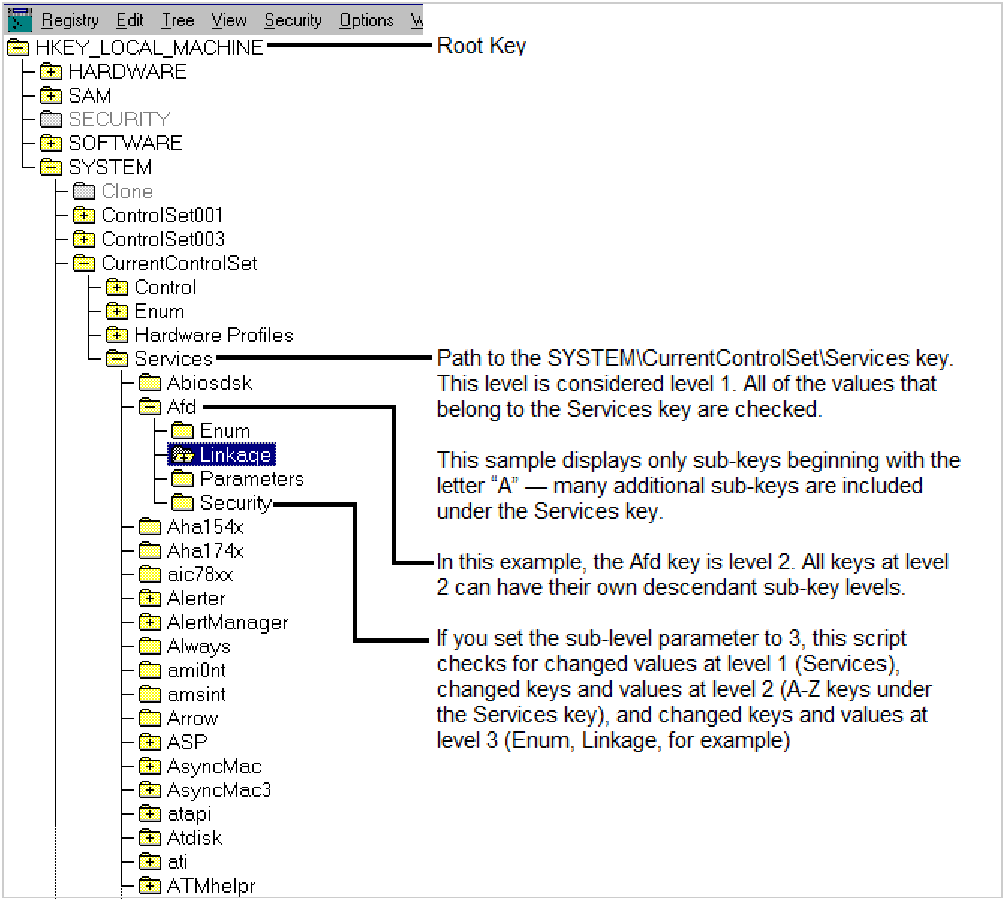
Description	How to Set It
Maintain initial key registry information across agent restarts?	Select **Yes** to indicate to the job to retain the information stored from the last job iteration for the monitored registry keys, subkeys, and values after a restart of the job. During the first job iteration that the job ever runs, a baseline of current registry key and value information is stored and updated during each job iteration. If at any point the job is restarted, either manually or automatically from an agent restart or agent computer restart, this option continues to use the last job iteration's information stored, thus avoiding a loss of comparison opportunity after the restart. To reset the baseline registry information stored using the same job instance of NT_RegistryChange, deselect this value, at which time the next iteration will take a new snapshot of the monitored registry keys, subkeys, and values and begin using the new snapshot in comparisons going forward.
Data Collection	
Collect data for changes in monitored registry keys and values?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the number of changes to the registry since the last time the job ran. The detail message includes specific information about each change. The default is unselected.

Example of How this Script Is Used

This script traverses the registry to check for changes to registry keys and sub-keys. This information can be extremely valuable in understanding the behavior and configuration of the computers you are monitoring but it can also be expensive in terms of processing time. To understand the impact of running this script, consider the following registry example.

If you set the *Path name* parameter to `SYSTEM\CurrentControlSet\Services`, the key becomes the first level of monitoring (sub-level 1) and all the keys at that level are checked for changes. If you set the *Sub-level* parameter to 3 for this job, the script then monitors all the values for all the sub-keys under the `SYSTEM\CurrentControlSet\Services` key and all the values for the sub-keys under the `SYSTEM\CurrentControlSet\Services` sub-key folders. With these settings, you can monitor a large number of key values but might put undue strain on your system.

As you can see in the following example, each sub-key level you monitor can contain many sub-keys and sub-key values:



Path to the SYSTEM\CurrentControlSet\Services key. This level is considered level 1. All of the values that belong to the Services key are checked.

This sample displays only sub-keys beginning with the letter “A” — many additional sub-keys are included under the Services key.

In this example, the Afd key is level 2. All keys at level 2 can have their own descendant sub-key levels.

If you set the sub-level parameter to 3, this script checks for changed values at level 1 (Services), changed keys and values at level 2 (A-Z keys under the Services key), and changed keys and values at level 3 (Enum, Linkage, for example)

One way to control the number of key values you monitor is by choosing the base path carefully. For example, you can set the `Path name` to a specific sub-key such as `SYSTEM\CurrentControlSet\Services\EventLog`. Depending on the number of sub-keys under the base path, however, you might also need to consider how best to set the sub-level parameter.

For example, if you set the *Path name* to `SYSTEM\CurrentControlSet\Services\EventLog` and the *Sub-level* parameter to `3`, the `EventLog` key becomes sub-level 1 and is checked for changes to values. The `EventLog` key contains the `Application`, `Security`, and `System` sub-keys, which as sub-level 2, are checked for new keys and values. Each of these sub-level 2 keys branches further, yielding dozens more keys at sub-level 3, each with values to check.

Because the number of monitored values can expand quickly, it is important to consider either narrowing the key path and sub-levels to check or lengthening the monitoring interval for this script to run effectively.

4.31 RemoteServiceDown

Use this Knowledge Script to monitor services on remote computers. You can specify the computers to monitor either directly using the *Machine list* parameter, or in a file containing a list of computer names or addresses.

This script tries to communicate with each of the remote computers in the *Machine list* parameter. This script raises an event if a named service is down or a specified computer cannot be reached from the computer where this script is running.

This script displays event information even if the remote computer is in maintenance mode.

NOTE: The AppManager agent does not need to be installed on the remote computer you want to monitor, nor does the remote computer need to exist in the Navigation pane or the TreeView pane.

When configuring an action for the RemoteServiceDown script, configure the Location to run on the MS (management server) or on a Proxy (a particular managed client computer).

If you instead configure an action to run on the managed client (MC), when a remotely monitored computer is placed into maintenance mode or scheduled maintenance mode, AppManager ignores any event conditions detected on the remote computer but does not disable the action.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if a specified service is down or if a specified computer cannot be reached from the computer where this script is running. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns data for each named service, either: ◆ **100** -- the service is running; or ◆ **0** -- the service is not running. The default is n.
Collect data only on down?	Set to **y** to collect data for charts and reports only when named services are down. If enabled, data collection returns a value of 0 when a service is down. Enable this parameter only if the *Collect data?* parameter is enabled. The default is n.
Machine list	Specify the names of the computers to be monitored. Separate multiple names with commas (,) and no spaces. For example: `AppSrvr,Storm,CorpSrvr`.

File name for machine list	Provide the full path to the file containing the list of computers to test. The file should contain the hostname or IP address for each computer in one or more lines. Each line can have multiple computer names, separated by commas and no spaces. For example:
	`NYC01,NYC02`
	`SALES01,10.15.221.5,SFO01`
	`LABMACH,QATEST`
	The job supports a maximum file size of 32KB. If the file size exceeds 32KB, the job stops and raises an error event message: `Out of string space`.

Description	How to Set It
Services	Provide the names of the services to be monitored. Separate multiple names with commas (,) and no spaces. The default is the NetIQ AppManager Agent service: `NetIQmc`.
Auto-start service?	Set to **y** to automatically restart down services. The default is y.
Event severity level for service down; auto-start failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is down and auto-start failed. The default is 5 (red event indicator).
Event severity level for service down; auto-start succeeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is down and auto-start succeeded. The default is 25 (blue event indicator).
Event severity level for service down; don't auto-start	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is down and you have selected not to restart it. The default is 18 (yellow event indicator).
Event severity level for service not found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a selected service cannot be found. The default is 8 (red event indicator).

4.32 RemoteServiceDownLR

Use this Knowledge Script to monitor services on remote computers. You specify the computers and services to monitor. A service that is detected as down can be restarted. The Windows services include those that are not discovered by AppManager, such as `WinLogon` or `NetIQ Corporationms`.

Before running this script, run the ConfigRemoteServiceDown Knowledge Script to store a list of computers and services in the local repository on the target computer.

After you run ConfigRemoteServiceDown on each target computer in a group, you can use RemoteServiceDownLR in a monitoring policy for the group. On each computer, RemoteServiceDownLR knows what to monitor because ConfigRemoteServiceDown has stored that information in the local repository.

This script displays event information even if the remote computer is in maintenance mode.

NOTE: The AppManager agent does not need to be installed on the remote computer you want to monitor, nor does the remote computer need to exist in the Navigation pane or the TreeView pane.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event if service is down?	Set to **y** to raise an event if a service is down. The default is y.
Collect data for service status?	Set to **y** to collect data for charts and reports. If enabled, data collection returns service status. The default is n.
Collect data only on service down?	Set to **y** to collect data only when a service is down. If enabled, and if the *Collect data?* parameter is also enabled, this script returns a value of 0 to indicate that a service is down. The default is n.
Auto-start service?	Set to **y** to automatically restart down services. The default is y.
Event severity when service down; auto-start failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is down and auto-start failed. The default is 5 (red event indicator).
Event severity when service down; auto-start succeeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is down and auto-start succeeded. The default is 25 (blue event indicator).
Event severity when service down; auto-start disabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is down and you have selected not to restart it. The default is 18 (yellow event indicator).
Event severity for service not found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a selected service cannot be found. The default is 8 (red event indicator).

4.33 Report_CPULoad

Use this Knowledge Script to generate a detailed report about CPU usage and queue length. Using this report, you can aggregate the data by minute, hour, or day, and calculate statistics for each period, for example, the average value per hour.

This report uses data collected by the CpuLoaded script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select the style	Select the style for the first page of your report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** provides all the datastreams on a single page The default is By computer.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Aggregation by	Select the time period by which the data in your report is presented. Select **Minute**, **Hour**, or **Day**. The default is Hour.
Aggregation interval	Specify the intervals to use to aggregate the data in the report. You can specify 1-5, 7, 8, 10, 12, 14, 15, 24, 28, 30, 60, or 90. The default is 1.

Description	How to Set It
Statistics to show per period	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the aggregation interval (for example, the average value for 1 Hour). ◆ **Minimum**. The minimum value of data points for the aggregation interval. ◆ **Maximum**. The maximum value of data points for the aggregation interval. ◆ **Count**. The number of data points for the aggregation interval. ◆ **Sum**. The total value of data points for the aggregation interval. ◆ **3Sigma**. The average + (3 * standard deviation) and average - (3 * standard deviation). ◆ **Std**. The standard deviation. The measure of how widely values are dispersed from the mean. ◆ **Box**. Lower fence, 25% point, median, 75% point, and upper fence for the aggregation interval. ◆ **Open**. The first value for the aggregation interval. ◆ **Close**. The last value for the aggregation interval. The default is Average.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a **Table** or **Chart** of datastream values in the report, or choose **Both**. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder name is NT_CPULoad.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. The default is n. The job ID helps you correlate a specific instance of a Report script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title for your report is NT CPU Load.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The default is n. The time stamp is made up of the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).

Description	How to Set It
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.34 Report_CPULoadSummary

Use this Knowledge Script to generate a summary report about CPU usage and queue length. Using this report, you can make a statistical analysis of the data point values, for example, the average or maximum value over a period.

This report uses data collected by the CpuLoaded script.

Resource Objects

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Select the style	Select the style for the first page of your report: ◆ **By computer** displays one value for each computer you selected. ◆ **By legend** displays one value for each different legend (the legend is a truncated form of the datastream legend visible in the Operator Console). ◆ **By computer and legend** displays one value for each unique legend from each computer. The default is By computer.
Data settings	

Description	How to Set It
Statistics to show	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the time range of the report. ◆ **Minimum**. The minimum value of data points for the time range of the report. ◆ **Maximum**. The maximum value of data points for the time range of the report. ◆ **Min/Avg/Max**. The minimum, average, and maximum values of data points for the time range of the report. ◆ **Range**. The range of values in the datastream (maximum - minimum = range). ◆ **StandardDeviation**. The measure of how widely values are dispersed from the mean. ◆ **Sum**. The total value of data points for the time range of the report. ◆ **Close**. The last value for the time range of the report. ◆ **Change**. The difference between the first and last values for the time range of the report (close - open = change). ◆ **Count**. The number of data points for the time range of the report. The default is Average.
Select sorting or display options	Specify whether to sort data in your report or how to display the data: ◆ **No sort**. Data is not sorted. ◆ **Sort**. Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right). ◆ **Top %**. Chart only the top N % of selected data (sorted by default). ◆ **Top N**. Chart only the top N of selected data (sorted by default). ◆ **Bottom %**. Chart only the bottom N % of data (sorted by default). ◆ **Bottom N**. Chart only the bottom N of selected data (sorted by default). The default is No sort.
Percentage (%) or count for top or bottom of chart	Specify a number for either the percent or count defined in *Select sorting or display options* (for example, Top 10%, or Top 10). The default is 25.
Truncate top or bottom?	Set to **yes** to truncate the top or bottom data in your report. If set to yes, the data table displays only the top or bottom N or % (for example, only the top 10%). If set to no, the table displays all data. The default is no.
Show totals on the table?	Set to **yes** to display additional calculations for each column of numbers in a table. If enabled, the following values are listed at the end of the table: ◆ **Report Average**. An average of all values in a column. ◆ **Report Minimum**. The minimum value in a column. ◆ **Report Maximum**. The maximum value in a column. ◆ **Report Total**: The total of all values in a column. The default is no.
Report settings	

Description	How to Set It
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for the report's output folder. The default folder prefix is NT_CPULoadSummary.
Add job ID to output folder name?	Set to **yes** to add the job ID to the name of the output folder. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT CPU Load Summary.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.35 Report_CPUResource

Use this Knowledge Script to generate a detailed report about the use of CPU resources, including the number of active processes, threads, and interrupts per second, and the utilization of CPU resources in user mode. Using this report, you can aggregate the data by minute, hour, or day, and calculate statistics for each period, for example, the average value per hour.

This report uses data collected by the CpuResource script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select the style	Select the style for the first page of your report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers. Each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer. ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** provides all the datastreams on a single page The default is By computer.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Aggregation by	Select the time period by which the data in your report is presented. Select **Minute**, **Hour**, or **Day**. The default is Hour.
Aggregation interval	Specify the intervals to use to aggregate the data in your report. You can specify 1-5, 7, 8, 10, 12, 14, 15, 24, 28, 30, 60, or 90. The default is 1.

Description	How to Set It
Statistics to show per period	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the aggregation interval (for example, the average value for 1 Hour). ◆ **Minimum**. The minimum value of data points for the aggregation interval. ◆ **Maximum**. The maximum value of data points for the aggregation interval. ◆ **Count**. The number of data points for the aggregation interval. ◆ **Sum**. The total value of data points for the aggregation interval. ◆ **3Sigma**. The average + (3 * standard deviation) and average - (3 * standard deviation). ◆ **Std**. The standard deviation. The measure of how widely values are dispersed from the mean. ◆ **Box**. Lower fence, 25% point, median, 75% point, and upper fence for the aggregation interval. ◆ **Open**. The first value for the aggregation interval. ◆ **Close**. The last value for the aggregation interval. The default is Average.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_CPUResource.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT CPU Resource.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).

Description	How to Set It
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.36 Report_CPUResourceSummary

Use this Knowledge Script to generate a summary report about the use of CPU resources, including the number of active processes, threads, and interrupts per second, and the utilization of CPU resources in user mode. Using this report, you can develop a statistical summary of the data you select, for example, the average value of data points over the period you define for the report.

This report uses data collected by the CpuResource script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Select the style	Select the style for the first page of your report: ◆ **By computer** displays one value for each computer you selected. ◆ **By legend** displays one value for each different legend (the legend is a truncated form of the datastream legend visible in the Operator Console). ◆ **By computer and legend** displays one value for each unique legend from each computer. The default is By computer and legend.
Data settings	

Description	How to Set It
Statistics to show	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the time range of the report. ◆ **Minimum**. The minimum value of data points for the time range of the report. ◆ **Maximum**. The maximum value of data points for the time range of the report. ◆ **Min/Avg/Max**. The minimum, average, and maximum values of data points for the time range of the report. ◆ **Range**. The range of values in the datastream (maximum - minimum = range). ◆ **StandardDeviation**. The measure of how widely values are dispersed from the mean. ◆ **Sum**. The total value of data points for the time range of the report. ◆ **Close**. The last value for the time range of the report. ◆ **Change**. The difference between the first and last values for the time range of the report (close - open = change). ◆ **Count**. The number of data points for the time range of the report. The default is Average.
Select sorting or display options	Specify whether to sort data in your report or how to display the data: ◆ **No sort**. Data is not sorted. ◆ **Sort**. Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right). ◆ **Top %**. Chart only the top N % of selected data (sorted by default). ◆ **Top N**. Chart only the top N of selected data (sorted by default). ◆ **Bottom %**. Chart only the bottom N % of data (sorted by default). ◆ **Bottom N**. Chart only the bottom N of selected data (sorted by default). The default is No sort.
Percentage (%) or count for top or bottom of chart	Specify a number for either the percent or count defined in *Select sorting or display options* (for example, Top 10%, or Top 10). The default is 25.
Truncate top or bottom?	Set to **yes** to truncate the top or bottom data in your report. If set to yes, the data table displays only the top or bottom N or % (for example, only the top 10%). If set to no, the table displays all data. The default is no.
Show totals on the table?	Set to **yes** to display additional calculations for each column of numbers in a table. If set to yes, the following values are listed at the end of the table: ◆ **Report Average**. An average of all values in a column. ◆ **Report Minimum**. The minimum value in a column. ◆ **Report Maximum**. The maximum value in a column. ◆ **Report Total**: The total of all values in a column. The default is no.
Report settings	

Description	How to Set It
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_CPUResourceSummary.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT CPU Resource Summary.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.37 Report_CPUUsageofProcessesSummary

Use this Knowledge Script to generate a summary report about CPU usage per named process, and total CPU usage by all named processes. Processes are named when you configure the CpuByProcess script. Using this report, you can develop a statistical summary of the data you select, for example, the average value of data points over the period you define for the report.

This report uses data collected by the CpuByProcess script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Select the style	Select the style for the first page of your report: ◆ **By computer** displays one value for each computer you selected. ◆ **By legend** displays one value for each different legend (the legend is a truncated form of the datastream legend visible in the Operator Console). ◆ **By computer and legend** displays one value for each unique legend from each computer. The default is By computer and legend.
Data settings	
Statistics to show	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the time range of the report. ◆ **Minimum**. The minimum value of data points for the time range of the report. ◆ **Maximum**. The maximum value of data points for the time range of the report. ◆ **Min/Avg/Max**. The minimum, average, and maximum values of data points for the time range of the report. ◆ **Range**. The range of values in the datastream (maximum - minimum = range). ◆ **StandardDeviation**. The measure of how widely values are dispersed from the mean. ◆ **Sum**. The total value of data points for the time range of the report. ◆ **Close**. The last value for the time range of the report. ◆ **Change**. The difference between the first and last values for the time range of the report (close - open = change). ◆ **Count**. The number of data points for the time range of the report. The default is Average.

Description	How to Set It
Select sorting or display options	Specify whether to sort data in your report or how to display the data: ◆ **No sort**. Data is not sorted. ◆ **Sort**. Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right). ◆ **Top %**. Chart only the top N % of selected data (sorted by default). ◆ **Top N**. Chart only the top N of selected data (sorted by default). ◆ **Bottom %**. Chart only the bottom N % of data (sorted by default). ◆ **Bottom N**. Chart only the bottom N of selected data (sorted by default). The default is No sort.
Percentage (%) or count for top or bottom of chart	Specify a number for either the percent or count defined in *Select sorting or display options* (for example, Top 10%, or Top 10). The default is 25.
Truncate top or bottom?	Set to **yes** to truncate the top or bottom data in your report. If set to yes, the data table displays only the top or bottom N or % (for example, only the top 10%). If set to no, the table displays all data. The default is no.
Show totals on the table?	Set to **yes** to display additional calculations for each column of numbers in a table. If set to yes, the following values are listed at the end of the table: ◆ **Report Average**. An average of all values in a column. ◆ **Report Minimum**. The minimum value in a column. ◆ **Report Maximum**. The maximum value in a column. ◆ **Report Total**: The total of all values in a column. The default is no.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_CPUUsageofProcessesSummary.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT CPU Usage of Processes Summary.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.

Description	How to Set It
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.38 Report_FilesOpen

Use this Knowledge Script to generate a report about the number of files open during a specified period. Using this report, you can aggregate data by minute, hour, or day, and calculate statistics for each period, for example, the average value per hour.

This report uses data collected by the FilesOpen script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.

Description	How to Set It
Select the style	Select the style for the first page of your report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** provides all the datastreams on a single page The default is By computer.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Aggregation by	Select the time period by which the data in your report is presented. Select **Minute**, **Hour**, or **Day**. The default is Hour.
Aggregation interval	Specify the intervals to use to aggregate the data in your report. You can specify 1-5, 7, 8, 10, 12, 14, 15, 24, 28, 30, 60, or 90. The default is 1.
Statistics to show per period	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the aggregation interval (for example, the average value for 1 Hour). ◆ **Minimum**. The minimum value of data points for the aggregation interval. ◆ **Maximum**. The maximum value of data points for the aggregation interval. ◆ **Count**. The number of data points for the aggregation interval. ◆ **Sum**. The total value of data points for the aggregation interval. ◆ **3Sigma**. The average + (3 * standard deviation) and average - (3 * standard deviation). ◆ **Std**. The standard deviation. The measure of how widely values are dispersed from the mean. ◆ **Box**. Lower fence, 25% point, median, 75% point, and upper fence for the aggregation interval. ◆ **Open**. The first value for the aggregation interval. ◆ **Close**. The last value for the aggregation interval. The default is Average.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.

Description	How to Set It
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_FilesOpen.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters.The default title is NT Files Open.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of the report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.39 Report_LogicalDiskAvailSummary

Use this Knowledge Script to generate a summary report about the amount of free space (in MB) on a logical disk. Using this report, you can develop a statistical summary of the data you select, for example, the average value of data points over the period you define for the report.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Select the style	Select the style for the first page of your report: ◆ **By computer** displays one value for each computer you selected. ◆ **By legend** displays one value for each different legend (the legend is a truncated form of the datastream legend visible in the Operator Console). ◆ **By computer and legend** displays one value for each unique legend from each computer. The default is By computer and legend.
Data settings	
Statistics to show	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the time range of the report. ◆ **Minimum**. The minimum value of data points for the time range of the report. ◆ **Maximum**. The maximum value of data points for the time range of the report. ◆ **Min/Avg/Max**. The minimum, average, and maximum values of data points for the time range of the report. ◆ **Range**. The range of values in the datastream (maximum - minimum = range). ◆ **StandardDeviation**. The measure of how widely values are dispersed from the mean. ◆ **Sum**. The total value of data points for the time range of the report. ◆ **Close**. The last value for the time range of the report. ◆ **Change**. The difference between the first and last values for the time range of the report (close - open = change). ◆ **Count**. The number of data points for the time range of the report. The default is Average.

Description	How to Set It
Select sorting or display options	Specify whether to sort data in your report or how to display the data: ◆ **No sort**. Data is not sorted. ◆ **Sort**. Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right). ◆ **Top %**. Chart only the top N % of selected data (sorted by default). ◆ **Top N**. Chart only the top N of selected data (sorted by default). ◆ **Bottom %**. Chart only the bottom N % of data (sorted by default). ◆ **Bottom N**. Chart only the bottom N of selected data (sorted by default). The default is No sort.
Percentage (%) or count for top or bottom of chart	Specify a number for either the percent or count defined in *Select sorting or display options* (for example, Top 10%, or Top 10). The default is 25.
Truncate top or bottom?	Set to **yes** to truncate the top or bottom data in your report. If set to yes, the data table displays only the top or bottom N or % (for example, only the top 10%). If set to no, the table displays all data. The default is no.
Show totals on the table?	Set to **yes** to display additional calculations for each column of numbers in a table. If set to yes, the following values are listed at the end of the table: ◆ **Report Average**. An average of all values in a column. ◆ **Report Minimum**. The minimum value in a column. ◆ **Report Maximum**. The maximum value in a column. ◆ **Report Total**: The total of all values in a column. The default is no.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_LogicalDiskAvailSummary.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT Logical Disk Available Summary.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.

Description	How to Set It
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.40 Report_LogicalDiskUsageSummary

Use this Knowledge Script to generate a summary report about the percentage of disk space used and the amount of free space (in MB). Using this report, you can develop a statistical summary of the data you select, for example, the average value of data points over the period you define for the report.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.

Description	How to Set It
Select the style	Select the style for the first page of your report: ◆ **By computer** displays one value for each computer you selected. ◆ **By legend** displays one value for each different legend (the legend is a truncated form of the datastream legend visible in the Operator Console). ◆ **By computer and legend** displays one value for each unique legend from each computer. The default is By computer and legend.
Data settings	
Statistics to show	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the time range of the report. ◆ **Minimum**. The minimum value of data points for the time range of the report. ◆ **Maximum**. The maximum value of data points for the time range of the report. ◆ **Min/Avg/Max**. The minimum, average, and maximum values of data points for the time range of the report. ◆ **Range**. The range of values in the datastream (maximum - minimum = range). ◆ **StandardDeviation**. The measure of how widely values are dispersed from the mean. ◆ **Sum**. The total value of data points for the time range of the report. ◆ **Close**. The last value for the time range of the report. ◆ **Change**. The difference between the first and last values for the time range of the report (close - open = change). ◆ **Count**. The number of data points for the time range of the report. The default is Average.
Select sorting or display options	Specify whether to sort data in your report or how to display the data: ◆ **No sort**. Data is not sorted. ◆ **Sort**. Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right). ◆ **Top %**. Chart only the top N % of selected data (sorted by default). ◆ **Top N**. Chart only the top N of selected data (sorted by default). ◆ **Bottom %**. Chart only the bottom N % of data (sorted by default). ◆ **Bottom N**. Chart only the bottom N of selected data (sorted by default). The default is No sort.
Percentage (%) or count for top or bottom of chart	Specify a number for either the percent or count defined in *Select sorting or display options* (for example, Top 10%, or Top 10). The default is 25.
Truncate top or bottom?	Set to **yes** to truncate the top or bottom data in your report. If set to yes, the data table displays only the top or bottom N or % (for example, only the top 10%). If set to no, the table displays all data. The default is no.

Description	How to Set It
Show totals on the table?	Set to **yes** to display additional calculations for each column of numbers in a table. If set to yes, the following values are listed at the end of the table: ◆ **Report Average**. An average of all values in a column. ◆ **Report Minimum**. The minimum value in a column. ◆ **Report Maximum**. The maximum value in a column. ◆ **Report Total**: The total of all values in a column. The default is no.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_LogicalDiskUsageSummary.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT Logical Disk Usage Summary.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.41 Report_MemoryUtilization

Use this Knowledge Script to generate a report about the use of physical and virtual memory, and paging files. Using this report, you can aggregate data by minute, hour, or day, and calculate statistics for each period (for example, the average value per hour).

This report uses data collected by the MemUtil script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select the style	Select the style for the first page of your report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** provides all the datastreams on a single page The default is By computer.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Aggregation by	Select the time period by which the data in your report is presented. Select **Minute**, **Hour**, or **Day**. The default is Hour.
Aggregation interval	Specify the intervals to use to aggregate the data in your report. You can specify 1-5, 7, 8, 10, 12, 14, 15, 24, 28, 30, 60, or 90. The default is 1.

Description	How to Set It
Statistics to show per period	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the aggregation interval (for example, the average value for 1 Hour). ◆ **Minimum**. The minimum value of data points for the aggregation interval. ◆ **Maximum**. The maximum value of data points for the aggregation interval. ◆ **Count**. The number of data points for the aggregation interval. ◆ **Sum**. The total value of data points for the aggregation interval. ◆ **3Sigma**. The average + (3 * standard deviation) and average - (3 * standard deviation). ◆ **Std**. The standard deviation. The measure of how widely values are dispersed from the mean. ◆ **Box**. Lower fence, 25% point, median, 75% point, and upper fence for the aggregation interval. ◆ **Open**. The first value for the aggregation interval. ◆ **Close**. The last value for the aggregation interval. The default is Average.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_MemoryUtilization.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT Memory Utilization.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).

Description	How to Set It
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.42 Report_MemoryUtilizationSummary

Use this Knowledge Script to generate a summary report about the use of physical and virtual memory, and paging files. Using this report, you can develop a statistical summary of the data you select, for example, the average value of data points over the period you define for the report.

This report uses data collected by the MemUtil script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Select the style	Select the style for the first page of your report: ◆ **By computer** displays one value for each computer you selected. ◆ **By legend** displays one value for each different legend (the legend is a truncated form of the datastream legend visible in the Operator Console). ◆ **By computer and legend** displays one value for each unique legend from each computer. The default is By computer and legend.
Data settings	

Description	How to Set It
Statistics to show	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the time range of the report. ◆ **Minimum**. The minimum value of data points for the time range of the report. ◆ **Maximum**. The maximum value of data points for the time range of the report. ◆ **Min/Avg/Max**. The minimum, average, and maximum values of data points for the time range of the report. ◆ **Range**. The range of values in the datastream (maximum - minimum = range). ◆ **StandardDeviation**. The measure of how widely values are dispersed from the mean. ◆ **Sum**. The total value of data points for the time range of the report. ◆ **Close**. The last value for the time range of the report. ◆ **Change**. The difference between the first and last values for the time range of the report (close - open = change). ◆ **Count**. The number of data points for the time range of the report. The default is Average.
Select sorting or display options	Specify whether to sort data in your report or how to display the data: ◆ **No sort**. Data is not sorted. ◆ **Sort**. Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right). ◆ **Top %**. Chart only the top N % of selected data (sorted by default). ◆ **Top N**. Chart only the top N of selected data (sorted by default). ◆ **Bottom %**. Chart only the bottom N % of data (sorted by default). ◆ **Bottom N**. Chart only the bottom N of selected data (sorted by default). The default is No sort.
Percentage (%) or count for top or bottom of chart	Specify a number for either the percent or count defined in *Select sorting or display options* (for example, Top 10%, or Top 10). The default is 25.
Truncate top or bottom?	Set to **yes** to truncate the top or bottom data in your report. If set to yes, the data table displays only the top or bottom N or % (for example, only the top 10%). If set to no, the table displays all data. The default is no.
Show totals on the table?	Set to **yes** to display additional calculations for each column of numbers in a table. If set to yes, the following values are listed at the end of the table: ◆ **Report Average**. An average of all values in a column. ◆ **Report Minimum**. The minimum value in a column. ◆ **Report Maximum**. The maximum value in a column. ◆ **Report Total**: The total of all values in a column. The default is no.
Report settings	

Description	How to Set It
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_MemoryUtilizationSummary.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT Memory Utilization Summary.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.43 Report_NetworkBusy

Use this Knowledge Script to generate a report about the use of bandwidth on network interface cards. Using this report, you can aggregate data by minute, hour, or day, and calculate statistics for each period, for example, the average value per hour.

This report uses data collected by the NetworkBusy script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select the style	Select the style for the first page of your report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** provides all the datastreams on a single page The default is By computer.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Aggregation by	Select the time period by which the data in your report is presented. Select **Minute**, **Hour**, or **Day**. The default is Hour.
Aggregation interval	Specify the intervals to use to aggregate the data in your report. You can specify 1-5, 7, 8, 10, 12, 14, 15, 24, 28, 30, 60, or 90. The default is 1.

Description	How to Set It
Statistics to show per period	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the aggregation interval (for example, the average value for 1 Hour). ◆ **Minimum**. The minimum value of data points for the aggregation interval. ◆ **Maximum**. The maximum value of data points for the aggregation interval. ◆ **Count**. The number of data points for the aggregation interval. ◆ **Sum**. The total value of data points for the aggregation interval. ◆ **3Sigma**. The average + (3 * standard deviation) and average - (3 * standard deviation). ◆ **Std**. The standard deviation. The measure of how widely values are dispersed from the mean. ◆ **Box**. Lower fence, 25% point, median, 75% point, and upper fence for the aggregation interval. ◆ **Open**. The first value for the aggregation interval. ◆ **Close**. The last value for the aggregation interval. The default is Average.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_NetworkBusy.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT Network Busy.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).

Description	How to Set It
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.44 Report_PagingHigh

Use this Knowledge Script to generate a report about the number of reads and writes per second to the pagefile. Using this report, you can aggregate data by minute, hour, or day, and calculate statistics for each period, for example, the average value per hour.

This report uses data collected by the PagingHigh script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select the style	Select the style for the first page of your report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** provides all the datastreams on a single page The default is By computer.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.

Description	How to Set It
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Aggregation by	Select the time period by which the data in your report is presented. Select **Minute**, **Hour**, or **Day**. The default is Hour.
Aggregation interval	Specify the intervals to use to aggregate the data in your report. You can specify 1-5, 7, 8, 10, 12, 14, 15, 24, 28, 30, 60, or 90. The default is 1.
Statistics to show per period	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the aggregation interval (for example, the average value for 1 Hour). ◆ **Minimum**. The minimum value of data points for the aggregation interval. ◆ **Maximum**. The maximum value of data points for the aggregation interval. ◆ **Count**. The number of data points for the aggregation interval. ◆ **Sum**. The total value of data points for the aggregation interval. ◆ **3Sigma**. The average + (3 * standard deviation) and average - (3 * standard deviation). ◆ **Std**. The standard deviation. The measure of how widely values are dispersed from the mean. ◆ **Box**. Lower fence, 25% point, median, 75% point, and upper fence for the aggregation interval. ◆ **Open**. The first value for the aggregation interval. ◆ **Close**. The last value for the aggregation interval. The default is Average.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report’s output folder. The default folder prefix is NT_PagingHigh.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report’s output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT Paging High.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.

Description	How to Set It
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.45 Report_PhysicalDiskIO

Use this Knowledge Script to generate a report about the number of reads, writes, and transfers per second for a physical disk. Using this report, you can aggregate data by minute, hour, or day, and calculate statistics for each period, for example, the average value per hour.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.

Description	How to Set It
Select the style	Select the style for the first page of your report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** provides all the datastreams on a single page The default is By computer.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Aggregation by	Select the time period by which the data in your report is presented. Select **Minute**, **Hour**, or **Day**. The default is Hour.
Aggregation interval	Specify the intervals to use to aggregate the data in your report. You can specify 1-5, 7, 8, 10, 12, 14, 15, 24, 28, 30, 60, or 90. The default is 1.
Statistics to show per period	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the aggregation interval (for example, the average value for 1 Hour). ◆ **Minimum**. The minimum value of data points for the aggregation interval. ◆ **Maximum**. The maximum value of data points for the aggregation interval. ◆ **Count**. The number of data points for the aggregation interval. ◆ **Sum**. The total value of data points for the aggregation interval. ◆ **3Sigma**. The average + (3 * standard deviation) and average - (3 * standard deviation). ◆ **Std**. The standard deviation. The measure of how widely values are dispersed from the mean. ◆ **Box**. Lower fence, 25% point, median, 75% point, and upper fence for the aggregation interval. ◆ **Open**. The first value for the aggregation interval. ◆ **Close**. The last value for the aggregation interval. The default is Average.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.

Description	How to Set It
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_PhysicalDiskIO.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters.The default title is NT Physical Disk IO.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.46 Report_PhysicalDiskQueueLength

Use this Knowledge Script to generate a report about physical disk queue length. Using this report, you can aggregate data by minute, hour, or day, and calculate statistics for each period, for example, the average value per hour.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select the style	Select the style for the first page of your report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** provides all the datastreams on a single page The default is By computer.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Aggregation by	Select the time period by which the data in your report is presented. Select **Minute**, **Hour**, or **Day**. The default is Hour.
Aggregation interval	Specify the intervals to use to aggregate the data in your report. You can specify 1-5, 7, 8, 10, 12, 14, 15, 24, 28, 30, 60, or 90. The default is 1.

Description	How to Set It
Statistics to show per period	Select a statistical method by which to display data in your report: - **Average**. The average value of data points for the aggregation interval (for example, the average value for 1 Hour). - **Minimum**. The minimum value of data points for the aggregation interval. - **Maximum**. The maximum value of data points for the aggregation interval. - **Count**. The number of data points for the aggregation interval. - **Sum**. The total value of data points for the aggregation interval. - **3Sigma**. The average + (3 * standard deviation) and average - (3 * standard deviation). - **Std**. The standard deviation. The measure of how widely values are dispersed from the mean. - **Box**. Lower fence, 25% point, median, 75% point, and upper fence for the aggregation interval. - **Open**. The first value for the aggregation interval. - **Close**. The last value for the aggregation interval. The default is Average.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report’s output folder. The default folder prefix is NT_PhysicalDiskQueueLength.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report’s output folder name. By default, the job ID is not included. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT Physical Disk Queue Length.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).

Description	How to Set It
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.47 Report_PrinterHealth

Use this Knowledge Script to generate a report about printer health. Printer health is determined by whether or not the printer is paused, and the queue length for printer jobs. Using this report, you can aggregate data by minute, hour, or day, and calculate statistics for each period, for example, the average value per hour.

This report uses data collected by the PrinterHealth script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select the style	Select the style for the first page of your report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** provides all the datastreams on a single page The default is By computer.

Description	How to Set It
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Aggregation by	Select the time period by which the data in your report is presented. Select **Minute**, **Hour**, or **Day**. The default is Hour.
Aggregation interval	Specify the intervals to use to aggregate the data in your report. You can specify 1-5, 7, 8, 10, 12, 14, 15, 24, 28, 30, 60, or 90. The default is 1.
Statistics to show per period	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the aggregation interval (for example, the average value for 1 Hour). ◆ **Minimum**. The minimum value of data points for the aggregation interval. ◆ **Maximum**. The maximum value of data points for the aggregation interval. ◆ **Count**. The number of data points for the aggregation interval. ◆ **Sum**. The total value of data points for the aggregation interval. ◆ **3Sigma**. The average + (3 * standard deviation) and average - (3 * standard deviation). ◆ **Std**. The standard deviation. The measure of how widely values are dispersed from the mean. ◆ **Box**. Lower fence, 25% point, median, 75% point, and upper fence for the aggregation interval. ◆ **Open**. The first value for the aggregation interval. ◆ **Close**. The last value for the aggregation interval. The default is Average.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_PrinterHealth.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. By default, the job ID is not included. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT Printer Health.

Description	How to Set It
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.48 Report_Process

Use this Knowledge Script to generate a report about the number of processes running during a specified period. Using this report, you can aggregate data by minute, hour, or day, and calculate statistics for each period, for example, the average value per hour.

This report uses data collected by the Processes script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.

Description	How to Set It
Select the style	Select the style for the first page of your report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** provides all the datastreams on a single page The default is By computer.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Select peak weekday(s)	Select the days of the week to include in your report. The default is every day of the week.
Aggregation by	Select the time period by which the data in your report is presented. Select **Minute**, **Hour**, or **Day**. The default is Hour.
Aggregation interval	Specify the intervals to use to aggregate the data in your report. You can specify 1-5, 7, 8, 10, 12, 14, 15, 24, 28, 30, 60, or 90. The default is 1.
Statistics to show per period	Select a statistical method by which to display data in your report: ◆ **Average**. The average value of data points for the aggregation interval (for example, the average value for 1 Hour). ◆ **Minimum**. The minimum value of data points for the aggregation interval. ◆ **Maximum**. The maximum value of data points for the aggregation interval. ◆ **Count**. The number of data points for the aggregation interval. ◆ **Sum**. The total value of data points for the aggregation interval. ◆ **3Sigma**. The average + (3 * standard deviation) and average - (3 * standard deviation). ◆ **Std**. The standard deviation. The measure of how widely values are dispersed from the mean. ◆ **Box**. Lower fence, 25% point, median, 75% point, and upper fence for the aggregation interval. ◆ **Open**. The first value for the aggregation interval. ◆ **Close**. The last value for the aggregation interval. The default is Average.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Specify whether to include a table or chart, or both, of datastream values in the report. The default is Table.

Description	How to Set It
Select chart style	Select the graphic properties for the charts in your report. The default chart style is Bar.
Select output folder	Select the parameters for your report's output folder. The default folder prefix is NT_Process.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default title is NT Process.
Add time stamp to title?	Set to **yes** to add a time stamp to the title of your report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp lets you run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Event severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Event severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Event severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.49 Report_TopCPUProcs

Use this Knowledge Script to generate a report about the total CPU used by all processes and which processes consume the most CPU resources.

This report uses data collected by the TopCpuProcs script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report. **NOTE:** For this report, select only one View, and up to 15 computers or server groups. The data wizard allows you to select more, but if you do, the Finish button is disabled. This mechanism prevents you from selecting too much data for the report.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Select output folder	Select the parameters for your report's output folder. The default prefix for the folder name is NT_TopCPUProcs.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Severity for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Severity for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Severity for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.50 Report_TopMemoryProcs

Use this Knowledge Script to generate a report about the total memory used by all processes and which processes consume the most memory resources.

This report uses data collected by the TopMemoryProcs script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report. **NOTE:** Select only one View, and up to 15 computers or server groups. The data wizard allows you to select more, but if you do, the Finish button is disabled. This mechanism prevents you from selecting too much data for the report.
Select time range	Select a **Specific** or **Sliding** date/time range from which the report should pull data. The default is Sliding.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Select output folder	Select the parameters for your report's output folder. The default prefix for the folder name is NT_TopMemoryProcs.
Add job ID to output folder name?	Set to **yes** to add the job ID to the report's output folder name. The default is no. A job ID lets you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters.
Event notification	
Event for report success?	Set to **yes** to raise an event if the report is successfully generated. The default is yes.
Severity for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a report is generated successfully. The default is 35 (magenta event indicator).
Severity for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a generated report contains no data. The default is 25 (blue event indicator).
Severity for report failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which report generation fails. The default is 5 (red event indicator).

4.51 RunAwayProcesses

Use this Knowledge Script to detect runaway processes on the specified computer by repeatedly sampling CPU usage for processes. This script raises an event if a process exceeds the CPU usage threshold in the number of consecutive samples taken (one at each interval).

For example, if this script detects that the process `cmd` has exceeded the CPU usage threshold for five consecutive monitoring periods, it might indicate that the process is trapped in an infinite loop or has encountered other problems. In addition to raising an event to notify you of the problem, you can stop any detected runaway processes. The detail message shows the list of processes being sampled.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if a process exceeds the CPU usage threshold in the number of consecutive samples taken (one at each interval). The default is y.
Collect data?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the CPU usage for runaway processes. The default is n.
Maximum CPU usage threshold for runaway processes	Specify the maximum percentage of CPU time any process can be using when sampled before an event is raised. The default is 90%.
Number of consecutive samples	Specify the number of consecutive samples to take before raising an event. The default is 3.
Number of runaway processes	Specify the number of processes to display in a detail event or data message. Type 0 for all processes. The default is 5.
Ignore these processes (comma separated, without spaces)	Specify the names of any processes to exclude from sampling. Separate the names with commas (,) and no space. The default is `SQLSERVR`.
Never kill these processes (comma separated, without spaces)	Specify the names of any processes that should never be stopped. Separate the names with commas (,) and no spaces. The default is `EXPLORER,NetIQmc,NetIQccm,NetIQms,SERVICES,LSASS,WINLOGON,svchost`. If you stop these processes, your computer restarts.
Kill runaway process when detected?	Set to **y** to automatically stop a process. AppManager does not stop any process you specify in the *Never kill these processes* parameter. The default is n.

Description	How to Set It
Event severity level for runaway processes detected	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a runaway process is detected. The default is 5 (red event indicator).
Event severity level for killed runaway process	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a runaway process is stopped. The default is 10 (red event indicator).
Event severity level for failed to kill runaway process	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a runaway process cannot be stopped. The default is 10 (red event indicator).

4.52 ServerBusy

Use this Knowledge Script to monitor Windows server activity for network clients. This script raises an event if the number of active sessions or the number of open files exceeds the threshold you specify.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the number of active sessions or the number of open files exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the number of active sessions and the number of open files. The default is n.
Maximum number of active sessions threshold	Specify the maximum number of client computers that can be connected to the server before an event is raised. The default is 50.
Maximum number of opened files threshold	Specify the maximum number of files that can be opened by network clients before an event is raised. The default is 200.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default is 5 (red event indicator).

4.53 ServerBytes

Use this Knowledge Script to monitor the rate of bytes transferred to and from the target computer. Because the transfer rate can vary dramatically depending on the activity being performed, you can click the **Advanced** tab and set the number of consecutive occurrences to a value greater than 1 to filter out insignificant spikes. The detail message includes the number of bytes sent and received per second.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the number of bytes transferred per second exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the number of bytes transferred per second. The default is n.
Maximum bytes transferred per second threshold	Specify the maximum number of bytes that can be transferred (sent and received) per second before an event is raised. The default is 800,000 bytes.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of bytes transferred per second exceeds the threshold. The default is 25 (blue event indicator).

4.54 ServerError

Use this Knowledge Script to monitor the number of sessions that errored out during the monitoring interval. This script tracks only the number of sessions that failed and were closed and dropped in an error state since the last time the script ran (delta value). This script raises an event if the number of the errored-out sessions exceeds the threshold you specify.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the number of errored-out sessions exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the number of errored-out sessions. The default is n.
Maximum errored-out sessions threshold	Specify the maximum number of errored-out sessions allowed before an event is raised. The default is 2.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default is 8 (red event indicator).

4.55 ServerTimeout

Use this Knowledge Script to monitor the number of sessions that timed out during the monitoring interval. To conserve resources, Windows servers automatically disconnect sessions that are idle for a set period. If a session's idle time exceeds the autodisconnect parameter for the server, the session times out and is closed. This script tracks the number of sessions that timed out since the last time the script ran (delta value) and raises an event if the number of the timed-out sessions exceeds the threshold you specify.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the number of the timed-out sessions exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the number of timed-out sessions. The default is n.
Maximum timed-out sessions threshold	Specify the maximum number of timed-out sessions allowed before an event is raised. The default is 2.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default is 25 (blue event indicator).

4.56 ServiceChange

Use this Knowledge Script to detect changes to the status and startup type of Windows services. This script raises an event if the status (such as running, stopped, or pending), or startup type (such as manual, automatic, or disabled) of any service changes. For example, this script raises an event if a service startup type changes from Automatic to Manual.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job Failure Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the ServiceChange job fails unexpectedly. The default is 5 (red event indicator).
Monitor Services	
Services to monitor	Specify the names of the services you want to monitor, separating multiple names with commas and no spaces. You can specify the internal service names or the service names displayed in the Control Panel. Type an asterisk (*) to check all services.The default is EventLog.
Services to exclude	Specify the names of the services you want to exclude, separating multiple names with commas and no spaces. You can specify a maximum of 1500 characters.
Event Notification	
Raise event if the start-type of a monitored service changes?	Set to **Yes** to raise an event if the startup type of a monitored service changes. The default is Yes.
Event severity for service start-type changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service start-type changes. The default is 10 (red event indicator).
Raise event if the status of a monitored service changes?	Set to **Yes** to raise an event if the status of a monitored service changes. The default is Yes.
Event severity for service status changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service status changes. The default is 10 (red event indicator).
Data Collection	

Description	How to Set It
Collect data for monitored services?	Set to **Yes** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- service is unchanged, or ◆ **0** -- service is not running or has changed. The default is unselected.

4.57 ServiceDown

Use this Knowledge Script to monitor whether specified Microsoft Windows services are stopped or started, and, optionally, start any service that is stopped. This script allows you to monitor Windows Services that are not discovered by AppManager, such as the `WinLogon` or `NetIQms` services.

TIP: Use the General_ServiceDown script to monitor services that are discovered by AppManager.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job Failure Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the ServiceDown job fails unexpectedly. The default is 5 (red event indicator).
Monitor Services	

Description	How to Set It
Services to monitor	Specify the names of the services you want to monitor, separating multiple names with commas. For example: `MSSQLServer,SQLServerAgent,PrintSpooler`. The default is `EventLog,WinMgmt`. You can specify the internal short service names (short service name), or the service name displayed in the Control Panel (long service name). Type an asterisk (`*`) to monitor all service startup types that are specified in the **Service start-type filter** parameter. To monitor multiple instances of the same instance name, use one of the following methods: ◆ Use an asterisk (*) before the service name to monitor all processes that end with the string you provide. For example, typing `*SQL` will monitor all processes that end with SQL. ◆ Use an asterisk (*) after the process name to monitor all processes that begin with the string you provide. For example, typing `SQL*` will monitor all processes that begin with SQL. ◆ Specify part of the name of a service with asterisks (*) on either side of the partial name. For example, typing `*SQL*` will monitor all processes that have SQL anywhere in the name.

Description	How to Set It
Service start-type filter	Select the startup type for the services you want to monitor. This parameter is used only when monitoring all services or when you use an asterisk (*) to monitor services using a wildcard with partial service names entered in the **Services to monitor** parameter. The startup types include: ◆ All ◆ Automatic (services that start during the boot process) ◆ Automatic - Delayed Start (services that start shortly after the boot process) ◆ Automatic - Trigger Start (services that start after a specified triggering event) ◆ Automatic [excludes Delayed and/or Trigger Start] ◆ Manual (services that you manually start) ◆ Manual - Trigger Start (services that you manually start after a specified triggering event) ◆ Manual [excludes Trigger Start] ◆ Disabled (services that have been set to not start) Use this parameter along with the *Services to monitor* parameter to monitor services by their startup types instead of their names. For example, if you enter "*" in the *Services to monitor* parameter and then select **Manual**, the script monitors all services whose startup type is Manual. The default is Automatic. **Notes** ◆ The script uses this parameter when you enter an asterisk (*) in the *Services to monitor* parameter. ◆ Be careful when using an asterisk (*) in the *Services to monitor* parameter along with the **Manual** option. In this situation, if you also select the *Also start services that were last stopped normally* parameter, AppManager starts all Manual services.
Services to exclude	Specify the names of the services you want to exclude from monitoring, separating multiple names with commas. You can specify the internal service names or the service names displayed in the Control Panel.
Start services that stopped abnormally?	Select **Yes** to automatically start services that stopped abnormally. The default is Yes.
Also start services that were last stopped normally?	Select **Yes** to automatically start a service that stopped normally, perhaps as the result of a stop request from another process or human interaction. The default is unselected.
Service start timeout	Set the maximum number of seconds to wait after initiating the start command before reporting that the service could not be started. If the service is not running after the specified amount of time, this script reports that the service did not start in the time you specified. The default is 30 seconds.
Monitor Dependent Services	
Monitor dependent services?	Select **Yes** to monitor services that are dependent upon the other services you chose to monitor. The default is Yes.

Description	How to Set It
Start dependent services that are stopped?	Select **Yes** to automatically start a stopped service that is dependent upon a stopped monitored service. AppManager starts dependent services after the monitored service is started. The default is unselected. The options you choose for start-type filter and service start timeout apply to dependent services as well. **NOTE:** If you enable this parameter, you must also enable the *Monitor dependent services?* parameter.
Event Notification	
Raise event if a service stopped abnormally and will not be restarted?	Select **Yes** to raise an event if a service stopped abnormally, and you have selected this script to restart the service. The default is Yes.
Event severity when a service stopped abnormally and will not be restarted	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service stopped abnormally, and you have selected this script to restart the service. The default is 5 (red event indicator).
Raise event if a service is stopped and cannot be started?	Select **Yes** to raise an event if a service is stopped and this script cannot start the service. The default is Yes.
Event severity when a service is stopped and cannot be started	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a stopped service cannot be started. The default is 10 (red event indicator).
Raise event if a stopped service is started successfully?	Select **Yes** to raise an event if AppManager successfully starts a stopped service. The default is Yes.
Event severity when a stopped service is started successfully	Set the event severity level, from 1 to 40, to indicate the importance of an event in which AppManager successfully starts a stopped service. The default is 25 (blue event indicator).
Raise event if a service was last stopped normally?	Select **Yes** to raise an event if a service was last stopped normally. If enabled, an event is raised only if *Also start services that were last stopped normally* is not selected. The default is Yes.
Event severity when a services was last stopped normally	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service stopped normally and was not set to be started. The default is 30 (blue event indicator).
Raise event if a monitored service does not exist?	Select **Yes** to raise an event if a service specified in the *Services to monitor* parameter does not exist. The default is Yes.
Event severity when a monitored service does not exist	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a monitored service cannot be found. The default is 8 (red event indicator).
Raise event if a monitored service is disabled?	Select **Yes** to raise an event if a service is disabled. The default is Yes. AppManager cannot automatically start disabled services.
Event severity when a monitored service is disabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a monitored service is disabled. The default is 12 (yellow event indicator).
Data Collection	

Description	How to Set It
Collect data for monitored services?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns a separate datastream for each service you monitor: ◆ **100** -- the service is started ◆ **0** -- the service is stopped ◆ **50** -- the service is stopped and was successfully started The data detail message includes the name of the service, the start type, and the status. The default is unselected.
Collect data for dependent services?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns a separate datastream for each dependent service you monitor: ◆ **100** -- the service is started ◆ **0** -- the service is stopped ◆ **50** -- the service is stopped and was successfully started The data detail message includes the name of the dependent service, the start type, and the status. The default is unselected.

4.58 ServiceDownLR

Use this Knowledge Script to detect whether specified services on the computer on which you run the script are down. A service detected as down can be restarted.The Windows services include those that are not discovered by AppManager, such as `WinLogon` or `NetIQms`.

This script requires that you first use the ConfigServiceDown Knowledge Script to store a list of services in the local repository on the computer where ServiceDownLR runs.

Once you have run ConfigServiceDown on each computer in a group, you can use ServiceDownLR in a monitoring policy for the group. On each computer, ServiceDownLR knows what to monitor because ConfigServiceDown previously stored that information in the local repository.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event if any service is down?	Set to **y** to raise an event if services are down. The default is y.
Collect data for service status?	Set to **y** to collect data for charts and reports. If enabled, data collection returns a separate datastream for each service you monitor: ◆ **100** -- the service is started ◆ **0** -- the service is stopped ◆ **50** -- the service is stopped and was successfully started The data detail message includes the name of the service, the start type, and the status. The default is n.
Collect data only on service down?	Set to **y** to collect data only when a service is down. If set to y, returns a value of 0. Enable this parameter only if the *Collect data?* parameter is enabled. The default is n.
Auto-start service?	Set to **y** to automatically restart down services. The default is y.
Event severity when service down; auto-start failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is down and auto-start fails. The default is 5 (red event indicator).
Event severity when service down; auto-start succeeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is down and auto-start succeeds. The default is 25 (blue event indicator).
Event severity when service down; auto-start disabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is down and the *Auto-start service?* parameter is set to n. The default is 18 (yellow event indicator).
Event severity when service not found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a specified service is not found. The default is 8 (red event indicator).
Event severity when Knowledge Script error occurs	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the ServiceDownLR job fails unexpectedly. The default is 35 (magenta event indicator).

4.59 ServiceHung

Use this Knowledge Script to detect if a Windows service is hung. A hung service is a service in a Start-Pending, Stop-Pending, Continue-Pending, or Pause-Pending state for a number of consecutive intervals. This script raises an event if any service is detected as hung. The service can then be stopped or restarted.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job Failure Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event when a job fails. The default is 5.
Event Notification	
Raise event when a service appears to be hung?	Select to **Yes** to raise an event if a service is detected as hung. The default is Yes.
Event severity level for failure to restart service	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is hung and an attempt to restart it was not successful. The default is 5 (red indicator).
Event severity level for success to restart service	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is hung and was restarted successfully. The default is 25 (blue event).
Event severity level when service stopped but not being restarted due to job configuration	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is hung and the **Restart hung service that the job kills?** parameter is unselected. The default is 18 (yellow event indicator).
Event severity level when service process cannot be killed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is hung and could not be stopped. The default is 10 (red event indicator).
Monitor Services	
Services to monitor (comma-separated list, use * to monitor all services on the agent)	Specify the names of the services you want to monitor, separating the names with commas (,) and no spaces. You can specify the internal service names or the service names displayed in the Control Panel. Type an asterisk (*) to monitor all services on the agent with a startup type of Automatic. The default is `EventLog`.

Description	How to Set It
Service start-type filter (used only when * used in Services to monitor	Select the startup type for the services you want to monitor. This parameter is used only when monitoring all services in the **Services to monitor** parameter. The start-up types include: - All - Automatic (services that start during the boot process) - Automatic - Delayed Start (services that start shortly after the boot process) - Automatic - Trigger Start (services that start after a specified triggering event) - Automatic - Delayed/Trigger Start - Automatic [excludes Delayed and/or Trigger Start] - Manual (services that you manually start) - Manual - Trigger Start (services that you manually start after a specified triggering event) - Manual [excludes Trigger Start] - Disabled (services that have been set to not start) Use this parameter along with the **Services to monitor** parameter to monitor services by their startup types instead of their names. For example, if you enter an asterisk (*) in the Services to monitor parameter and then select **Manual**, the script monitors all services whose startup type is Manual. The default is Automatic.
Services to exclude (comma-separated list)	Specify the names of the services you want to exclude, separating the names with commas (,) and no spaces. You should specify the short (internal) name of the service, which can be found as `Service Name` in the Properties dialog of the respective service in the Services console in the Control Panel.
Maximum number of consecutive iterations before service is considered hung	Specify the maximum number of consecutive times a service can be in a Start-Pending, Stop-Pending, Continue-Pending, or Pause-Pending state before it is considered hung. The default is 2.
Kill the hung service? (y/n)	Select **Yes** to kill the hung service. The default is Yes.
Restart hung service that the job kills? (y/n)	Select **Yes** to restart a hung service that the job kills. The default is Yes.
Data Collection	
Collect data for monitored services?	Select to **Yes** to collect data for charts and reports. If enabled, data collection returns: - **100** -- the service is not hung - **0** -- the service is hung. The default is unselected.

4.60 ServiceRemove

Use this Knowledge Script to detect when Windows services are added or removed in the monitoring interval. This script raises an event when changes are made to the services installed on the target computer. The script monitors only changes in the list of automatic services (startup type), including installations and uninstallations of services in automatic startup and changing a service from automatic to disabled/manual or vice-versa.

NOTE: If a computer is restarted after a service is removed, the agent counters are reset. Once the counters are reset, AppManager cannot detect the service removal and does not raise an event.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event when changes are made to the services installed on the target computer. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of services currently installed. The default is n.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which service changes are detected. The default is 12 (yellow event indicator).

4.61 SharedFiles

Use this Knowledge Script to monitor and list shared files that are open. This script raises an event if the number of open shared files exceeds the threshold you specify. Run this script on the server that hosts the shared files.

Prerequisite

Before running this script, ensure that the AppManager Client Resource Monitor service (`netiqmc.exe`) is set to log on as a domain user account in the same domain as, or a domain trusted by, the target computer.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is to **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if the number of open shared files exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of shared files currently open. The detail message lists the shared files with file IDs and lock information. The default is n.
Maximum number of shared files open threshold	Specify the maximum number of shared files that can be open at one time before an event is raised. The default is 100.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of open shared files exceeds the threshold. The default is 8 (red event indicator).

4.62 SystemUpTime

Use this Knowledge Script to monitor the system up time for a Windows server or workstation. This script tracks the number of hours that the computer has been operational since it was last rebooted. This script raises an event if the computer was rebooted within the monitoring interval.

To monitor whether a computer has gone down for reasons other than being placed in maintenance mode, use the General_MachineDown Knowledge Script.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event when reboot has occurred since last job iteration (y/n)?	Set to **y** to raise an event if the computer was rebooted within the monitoring interval. The default is y.

Description	How to Set It
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of hours the system has been up. The default is n.
Event severity level for system reboot	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the computer was rebooted. The default is 8 (red event indicator).
Event severity level for an unexpected Knowledge Script error	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the SystemUpTime job fails unexpectedly. The default is 35 (magenta event indicator).

4.63 TopCpuProcs

Use this Knowledge Script to monitor total CPU resources used by all processes and which processes consume the most CPU resources. This script raises an event if the percentage of CPU usage exceeds the threshold you specify. In addition, this script generates a datastream for processor usage.

Resource Object

CPU folder

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Create event if processor utilization is over the threshold?	Set to **Yes** to raise an event if the percentage of CPU time used exceeds the threshold you specify. The default is Yes.
Severity - Processor utilization over the threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which CPU usage exceeds the threshold. The default is 5 (red event indicator).
Number of processes to include in detail message	Specify the number of top processes to display in the detail message (event or data). Enter 0 to display all processes. The default is 10. **NOTE:** Limit the number of processes included in the detail message to the top five to ten processes. In most cases, including all processes increases the size of the detail message without providing much more useful information. Typically, the top few processes are the most significant and the most userful for troubleshooting purposes.
Severity - Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the TopCpuProcs job fails unexpectedly. The default is 5 (red event indicator).

Description	How to Set It
Data Collection	
Collect processor utilization data?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the process name, ID, and utilization percentage (%) for the number of processes you set in *Number of processes to include in detail message*. The default is unselected. **NOTE:** If the value you set in *Number of processes to include in detail message* is greater than the number of processes running on the computer, the event detail message only contains the list of running processes; AppManager does not include blank lines to represent the non-running processes.
Monitoring	
Threshold - Total processor utilization	Specify the maximum percentage of CPU resources that can be in use for all processes before an event is raised. The default is 85%.

4.64 TopMemoryProcs

Use this Knowledge Script to monitor total memory (in KB) usage for all processes and to identify which processes consume the most memory. This script raises an event if memory usage exceeds the threshold you specify. In addition, this script generates a datastream for memory utilization.

Resource Object

Memory folder

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	
Create event if memory utilization exceeds the threshold?	Select **Yes** to raise an event if memory usage exceeds the threshold you specify. The default is Yes.
Severity - Memory utilization over the threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which memory usage exceeds the threshold. The default is 5 (red event indicator).

Description	How to Set It
Number of processes to include in detail message	Specify the number of top processes to display in the detail message (event or data). Enter 0 to display all processes. The default is 10. **NOTE:** Limit the number of processes included in the detail message to the top five to ten processes. In most cases, including all processes increases the size of the detail message without providing much more useful information. Typically, the top few processes are the most significant and the most useful for troubleshooting purposes.
Severity - Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the TopMemoryProcs job fails unexpectedly. The default is 5 (red event indicator).
Data Collection	
Collect memory utilization data?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the process name, ID, and memory utilization (in KB), as well as job configuration information for each data point value. The default is unselected.
Monitoring	
Threshold - Total memory utilization	Specify the maximum amount of memory that can be in use for all processes before an event is raised. The default is 5120 KB.
Threshold - Size scale	Select the scale for the total memory utilization threshold you specify (kilobytes, megabytes, gigabytes, terabytes). The default is kilobytes.

4.65 TrustRelationship

Use this Knowledge Script to test the domain trust relationship from the computer where you run this script to a specified domain. The domain of the managed computer running this script is considered the *trusting* or resource domain. The domains you specify as script properties are the domains you expect to be trusted domains. This script raises an event if there are problems with the domain trust, such as when a trusted password is no longer valid or the Primary Domain Controller of the trusting domain cannot be located.

NOTE: Before running this script, be sure the `netiqmc` and `netiqccm` services are set to run as a domain user account with Administrator privileges in both the trusting and trusted domains. For example, to test whether Domain A still trusts Domain B, the agent services must run as an account with domain Administrator privileges in both Domain A and Domain B. Use the Services Control Panel to identify an account for the service to run as.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Raise event?	Set to **y** to raise an event if there are problems with the domain trust relationships. The default is y.
Collect data?	Specify whether to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the domain of the managed computer trusts the domains entered, or ◆ **0** -- the trust relationship is broken. The default is n.
Trusted domains	Provide a list of trusted domains, separated by commas with no spaces. Trusted domains contain resources that computers in other domains can use.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which there are problems with the domain trust relationships. The default is 35 (red event indicator).

4.66 UnixRemoteProcessDown

Use this Knowledge Script to monitor applications on remote UNIX computers where you cannot easily install a UNIX agent. This script uses a proxy UNIX agent, installed on another computer, to monitor processes on the remote UNIX computer.

If a monitored process is found to be down, this script can restart it automatically, using a script or command you supply. Be sure to read the help for the *Scripts or commands to restart processes* parameter before proceeding.

You can specify the process names to be monitored in the *Processes to monitor* parameter, or you can provide a configuration file in XML format to specify processes to monitor and what steps to take to restart them if they are down. For more information, see “Remote Process Monitoring Using a Configuration File” on page 169.

Resource Object

UNIX Machine folder

Default Schedule

The default interval for this script is **Every 20 seconds**.

Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
Event Notification	

Description	How to Set It
Raise event if process down?	Select **Yes** to raise an event if a monitored process is down. The default is Yes.
Event severity when process is down	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a process is down. The default is 10.
Raise event if process is running?	Select **Yes** to raise an event if a monitored process is running. The default is unselected.
Event severity when process is running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a monitored process is running. The default is 25.
Remote Host Connection Configure access to the remote managed computers by specifying their root password. All of the remote computers must use the same root password. This script can use SSH with root password authentication or Telnet to communicate with the remote managed computer.	
Password for root user account	To use Secure Shell (SSH) for the connection to the remote computers, ensure that SSH with root authentication is enabled on the remote UNIX computers where you want to install the UNIX agent. For this parameter, specify the password for the root user to securely access the remote UNIX computers. This script does not support SSH root authentication with an RSA key.

Connection Transport	This script can use SSH with root password authentication or Telnet to communicate with the remote managed computers.
	If you select the **Telnet/FTP** option (the default), the Telnet prompt on the remote computer must end with a space or one of the following characters: %, >, #, $
	The following is an example of a supported Telnet prompt:
	`user@hostname>`
	Here is an example of an unsupported Telnet prompt:
	`<user@hostname:/tmp - 2005-Mar-09>`
	`->`
	In the example above, the last character in the first line of the two-line prompt is a line feed character, which is not supported.

Telnet non-root user account	If you selected Telnet to connect to the remote UNIX computers, specify a non-root user account to use for the connection. When connecting to a remote UNIX computer using Telnet and FTP, this script switches from the non-root user to the root user.
Telnet non-root user password	If you selected Telnet as the connection transport medium, specify the password for the non-root user account to connect to the remote UNIX computers.
Monitoring Source Configuration	
Full path to configuration file for remote monitoring	Supply a full directory path to an XML file to use for communications with the remote UNIX computer. The default is c:\temp\`config.xml`
Manual Configuration	

Description	How to Set It
Hostnames or IP addresses where processes are to be monitored	Supply a list of hostnames or IP addresses of the UNIX computers where processes are to be monitored. Separate multiple hostnames with commas (,) and no spaces. Supply IP addresses in dotted notation, such as 23.45.678.9. Separate multiple IP addresses with commas and no spaces.
Processes to monitor	Supply the names of the UNIX application processes to monitor. Separate multiple process names with commas and no spaces. You can also enter a Perl regular expression here if you want to exclude and include processes on various platforms using one argument. See “Running this Knowledge Script” on page 169 for more information.
Scripts or commands to restart processes	Supply one of the following: ◆ a list of full directory paths to script files to use to restart any processes that are found to be down, or ◆ a list of commands to use to restart these processes. There is no need to supply a value for this parameter if you specify “n” for the *Restart process if down?* parameter. If you configure this script to restart a process, specify a list of restart commands or shell scripts that contain the restart commands. Do not execute restart commands in the foreground. When executing a restart command in the foreground, this script cannot run at its next scheduled interval until after all the restart commands have completed. When specifying: ◆ A list of commands to run on the remote computer, run each command in the background by adding an ampersand (&) and separating each command with a comma. If this script is configured to use Telnet/FTP, you can restart a process in the background by adding an ampersand (&) to each command. If this script is configured to use SSH/SFTP, use a shell script on the remote computer to restart the processes in the background and ensure that `stdout` and `stderr` are redirected to a log file. When configured to use SSH/SFTP, this script always executes a command to restart a process in the foreground. ◆ A shell script on the remote computer that restarts the processes you want, in the shell script, add an ampersand (&) to each restart command—and ensure that `stdout` and `stderr` are redirected to a log file—to restart a process in the background.
Restart process if down?	Provide a list specifying “y” or “n” for each process in the *Processes to monitor* parameter. Specify y for a process if you want this script to restart it on the remote computer if it is found to be down. The commands or scripts you specified for the previous parameter will be used. Separate the list of Ys and Ns with commas and no spaces.

Running this Knowledge Script

The UnixRemoteProcessDown script requires the proxy UNIX agent to run as the root user account. To enable this script, you must run the AppManager installation program (the `AMxx_UNIX_setup.exe` file) on the proxy computer. An extra "helper" file will be installed: `UnixRemoteProcessDown.exe`.

To use this script to monitor the up and down status of the UNIX agent, specify `nqmagt` in the list of processes to monitor. If the `nqmagt` process is down, you can specify a restart command to restart the agent:

```
/etc/init.d/nqmdaemon start
```

This script can use either the Secure Shell (SSH) program with root password authentication or Telnet to make a secure connection to the remote UNIX computer. By default, Telnet is used, but you can select SSH/SFTP from the *Connection Transport* parameter to use Secure Shell instead. If you choose to use Telnet, you must supply a non-root user account name and password.

NOTE: Proxy monitoring with this script is possible only if the SSH program is installed on the target computer, or if the Telnet protocol is enabled on it.

You can also supply a Perl regular expression for the *Names of processes to monitor* parameter to check for a specific string. For example, you can exclude and include processes on various platforms using one argument. A process may be running out of the `/usr`, the `/opt`, or the `/var` directory, but you are not sure where. Or perhaps a process is running out of different locations on different platforms. If you enter

```
(/usr|/opt)/[processname]
```

for the *Names of processes to monitor* parameter, the script would monitor the process that is running in `/usr` OR in `/opt` but NOT in `/var`.

Remote Process Monitoring Using a Configuration File

The UnixRemoteProcessDown script includes an option to use a configuration file in XML format to supply monitoring instructions to the agent. In such a file, you can supply a list of processes to monitor on a given remote UNIX computer, specify how to restart these processes, and indicate whether to restart these processes.

By default, the script looks for the following configuration file:

```
c:\temp\config.xml
```

However, you can supply a different file as the value for the *Full path to configuration file for remote monitoring* parameter.

Following is an example of a valid XML configuration file that tells the UNIX agent which processes to monitor and what to do if the processes are not running:

```
<?xml version="1.0" encoding="utf-8" ?>
<SERVERS>
  <SERVER name="uws3">
    <PROCESS name="nqmagt" startupscript="/etc/init.d/nqmdaemon start" restart="y"/>
    <PROCESS name="xntpd"  startupscript="/etc/init.d/xntpd start" restart="n"/>
  </SERVER>
  <SERVER name="uws19">
    <PROCESS name="inetd"  startupscript="/etc/init.d/inetsvc start" restart="n"/>
    <PROCESS name="init"  startupscript="/etc/init.d/init start" restart="n"/>
  </SERVER>
</SERVERS>
```

4.67 BestPractices Knowledge Script Group

The AppManager for Microsoft Windows BestPractices Knowledge Script Group (KSG) enables a “best practices” usage for monitoring your NT environment. You can use this KSG with AppManager monitoring policies. A monitoring policy enables you to efficiently and consistently monitor all the resources in your environment using a set of pre-configured Knowledge Scripts. For more information, see “About Policy-Based Monitoring” in the AppManager Help.

You can find the BestPractices KSG in the **NT** tab of the Knowledge Script pane in the Control Console.

All the scripts in the KSG have their parameters set to recommended values. To run all of the recommended scripts in the KSG at one time, click the **NT** tab, and then run the KSG on an AppManager resource.

The KSG is a subset of a module of a module’s Knowledge Scripts. The script that belongs to a KSG is a differently copy of the original script you access from the NT tab. If you modify a script that belongs to a KSG, the parameter settings of the original script in the NT tab are not affected.

In few cases, default script parameter settings are different when the script is deployed as part of a KSG, and when it is deployed alone. The default settings of a script within a group depend on its monitoring purpose within the larger group, and on the intended monitoring scope of that group.

If you modify or remove a script associated with the NT KSG and want to restore it to its original form, you can reinstall the AppManager for Microsoft Windows module on the repository computer or check in the appropriate script from the `AppManager\qdb\kp\WinOS` directory.

The following Knowledge Scripts are members of the NT BestPractices KSG to monitor

- CpuLoaded
- DiskSpace
- MemUtil
- NetworkBusy
- ServiceDown
- SystemUpTime
- TopCpuProcs

5 WIN2000 Knowledge Scripts

The Windows Server (WIN2000) category provides Knowledge Scripts for monitoring Windows servers.

From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. Or in the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
ADDNSRegistrationEventLog	Scans the Event Log for Active Directory-related DNS registration problems.
DFSLinkDown	Monitors the up and down status of a DFS root, link, or replica.
DFSReplicationBacklog	Monitors the number of backlog files in the DFS folder.
DFSServiceDown	Monitors the up and down status of the DFS service.
DiskQuotaStatus	Monitors disk quota status for the specified logical drive.
DNSAXFRStat	Monitors changes to AXFR statistics for a DNS since the last interval.
DNSDatabaseNodeMemory	Monitors the total memory used by a DNS service for database nodes.
DNSDynaUpdateError	Monitors the number of dynamic update rejections and timeouts during the monitoring interval.
DNSDynaUpdateStat	Monitors the total number of dynamic updates queued by the DNS server.
DNSEventLog	Scans the Windows event log for DNS entries matching your selection criteria.
DNSRecursiveQuery	Monitors recursive queries for the DNS server and checks the number of recursive query errors or timeouts per second.
DNSSecureUpdate	Monitors secure updates for the DNS server and checks the percentage of attempted updates that failed.
DNSTotalQuery	Monitors total query activity for a DNS server and checks the number of queries received per second and the number of responses sent per second.
DNSWINSStat	Monitors the WINS activity for a DNS server.
DNSZoneTransfer	Monitors DNS zone transfer activity and zone transfer failures.
FrsBusy	Monitors the CPU utilization and various statistics of FRS.
FrsEventLog	Scans the Windows FRS log for file replication events.
FrsReplicaError	Monitors the various error statistics of the FRS, including errors in authentication, bindings, and packets sent.
FrsServiceDown	Monitors the up and down status of the FRS.
GroupPolicyAddRemove	Determines whether a Group Policy has been added to or removed from a computer.

Knowledge Script	What It Does
GroupPolicyCount	Counts the number of Group Policies for the target server.
GroupPolicyLinkSnapshot	Lists the links associated with a group policy object.
GroupPolicyRefresh	Refreshes the computer or user group policy on the target server on demand.
GroupPolicySnapshot	Lists all the group policies on the target server.
IASServiceDown	Monitors the up and down status of the Internet Authentication Service.
LSASSWatch	Checks whether the Kerberos Key Distribution Center service process, LSASS, is running or hung.
MSIPackagesChange	Monitors the programs and components installed or uninstalled using the Microsoft Windows Installer (MSI).
PrinterErrors	Monitors printers for problems with printing jobs.
PrinterEventLog	Scans the Windows System log for printer-related events.
PrinterQueue	Monitors the printer queue length.
PrinterUtil	Monitors the bytes per second being handled by the printer.
RemoteStorageEventLog	Scans the Windows Application log for Remote Storage-related events.
RemoteStorageServiceDown	Monitors the up and down status of Remote Storage services.
RSVPEventLog	Scans the Windows Application log for QoS/RSVP-related events.
RSVPServiceDown	Monitors the up and down status of the QoS/RSVP service.
SMTPEventLog	Scans the Windows System log for SMTP-related events.
SMTPQueues	Monitors the queue length for the SMTP queues.
SMTPServiceDown	Monitors the up and down status of the SMTP service.

5.1 ADDNSRegistrationEventLog

Use this Knowledge Script to scan the Event Log for Active Directory-related DNS registration problems. Each time this script runs, it checks the Event Log for entries matching your selection criteria and raises an event if matching entries are found.

Resource Object

DNS folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if log entries match your selection criteria. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns information based on the other parameter values you enabled. The default is n.
Start with events in past N hours	Set this parameter to determine which events are searched for the *first* time the Knowledge Script is run. Subsequent searches begin where the last search finished. The following entries are valid: ◆ Enter **-1** to search all current and previous System Log events during the first interval. ◆ Enter **0** to search only for current events; previous events are not searched. ◆ Enter the number of hours to go back in the System Log to scan for matching events. For example, enter 8 to scan the last 8 hours of the System Log for matching entries. The default is 0.
Monitor events of type:	Set to **y** for each type of event you want to monitor: ◆ Error ◆ Warning ◆ Information ◆ Success Audit ◆ Failure Audit If you enable data collection or events, and set any of these parameters to n, this script does not raise an event or collect data for that type of log entry. The default is y.
Filter the [...] field for	To limit the types of entries that raise AppManager events and the type of data that is collected, enter a search string that filters the following fields in the Windows Event Log: ◆ **Category**. Specify one or more text strings to look for in the Category field. Separate multiple strings with commas. ◆ **User**. Specify a search string to look for events associated with a particular user, for example, `<domain name>\<user name>`. Separate multiple strings with commas. For example: `USA\Tom,USA\Chris,EUROPE\Alex`. ◆ **Computer**. Specify computer names to look for. Separate multiple entries with commas. For example: SHASTA,MARS. ◆ **Description**. Specify a detail description or keywords in the description. A string can contain spaces, underscores, and periods. Separate multiple entries with commas. For example: `no domain,critical error from the Active Directory`. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you are specifying only include criteria, the colon is not necessary.

Parameter	How to Set It
Maximum number of events per event message	Specify the maximum number of DNS Registration Event Log events that can be returned in each event report. For example, if this value is set to 30, and 67 Registration Event Log events are found, then three event reports are raised: two reports containing 30 events and one report containing seven events. The Message column on the Events tab displays the number of events in the event report, the type of log the events are from, and the event report batch number. The batch number is the sequential number of the event report. Batch numbers start at 1 for each Knowledge Script iteration. The default is 30.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of the an event in which log entries match your selection criteria. You can adjust the severity based on the types of events you are checking. The default severity level is 8 (red event indicator).

5.2 DFSLinkDown

Use this Knowledge Script to monitor Distributed File System (DFS) roots, links, and replicas. For each root replica and link replica, you can check whether the directory for the corresponding replica exists. This script raises an event if any root, root replica, link, or link replica is down.

By default, this script checks the DFS roots and links found during discovery. You can, however, set this script to discover DFS links dynamically each time it runs.

Resource Objects

DFS folder

DFS Root folder

DFS Link object

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Create event if root, link, or replica is down?	Set to **Yes** to raise an event when a DFS root, link, or replica is down. The default is Yes.
Severity - DFS volume down	Set the severity level, from 1 to 40, to indicate the importance of an event in which DFS volume is down. The default is 5 (red event indicator).

Parameter	How to Set It
Severity - DFS path not found	Set the severity level, from 1 to 40, to indicate the importance of an event in which the DFS path cannot be found. The default is 22 (blue event indicator).
Severity - DFS root down	Set the severity level, from 1 to 40, to indicate the importance of an event in which the DFS root is down. The default is 5 (red event indicator).
Severity - DFS root replica down	Set the severity level, from 1 to 40, to indicate the importance of an event in which the DFS root replica is down. The default is 6 (red event indicator).
Severity - DFS link down	Set the severity level, from 1 to 40, to indicate the importance of an event in which the DFS link is down. The default is 7 (red event indicator).
Severity - DFS link replica down	Set the severity level, from 1 to 40, to indicate the importance of an event in which the DFS link replica is down. The default is 8 (red event indicator).
Severity - Job Failure	Set the severity level, from 1 to 40, to indicate the importance of an event in which the DFSLinkDown job fails unexpectedly. The default is 5 (red event indicator).
Data Collection	
Collect data?	Set to **Yes** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- a root, link, or replica is up, or ◆ **0** -- a root, link, or replica is down. The default is unselected.
Monitoring	
Dynamically enumerate DFS root/links?	Set to **Yes** to dynamically enumerate DFS links at each monitoring interval. The default is unselected. **NOTE:** If you select Yes, run this script on the DFS folder object.
Check if the replica directory exists?	Set to **Yes** to check for replica directories for each root or link found. The default is unselected. **NOTE:** If the AppManager agent services, NetIQ AppManager Client Resource Monitor (`NetIQmc.exe`) and NetIQ AppManager Client Communication Manager (`NetIQccm.exe`), are running under the Local System account or under a user account that does not have permission to read some directories, checking for replica directories may fail and cause this script to return an error.

5.3 DFSReplicationBacklog

Use this Knowledge Script to monitor status of Distributed File System (DFS) replication backlog. This script raises an event if the number of files in the backlog crosses the threshold value. By default, this knowledge script monitors only incoming backlog from remote servers.

NOTE: NetIQ MC service needs to run under domain user for the Job to execute.

Resource Objects

DFS folder

Default Schedule

The default interval for this script is **1 day**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Setting	
Monitor outgoing backlog?	Set to Yes to monitor the outgoing backlog files. The default is No.
Job failure event notification	
Event severity when job fails	Set the level between 1 and 40, to indicate the importance of an event in which the DFSReplicationBacklog job fails unexpectedly. The default is 5 (red event indicator).
Event Notification	
Raise event if backlog file count crosses the threshold?	Set to Yes to raise an event when the number of files in the backlog crosses the threshold. The default is Yes.
Threshold for number of files in backlog	Set the threshold count for the number of files in backlog. The default is 50.
Event severity when number of files in backlog crosses threshold	Set the severity level, from 1 to 40, to indicate the importance of an event in which the backlog files crosses the threshold count. The default is 15.
Event severity when failed to collect backlog	Set the severity level, from 1 to 40, to indicate the importance of an event in which the collection of backlog file fails. The default is 10 (red event indicator).
List the backlog file names in detail message	Lists the names of the backlog file in detail message. The default is Yes.
Data Collection	
Collect data?	Set to **Yes** to collect data for charts and reports. If enabled, data collection returns. The default is No.

5.4 DFSServiceDown

Use this Knowledge Script to monitor the up and down status of Distributed File System (DFS) roots, links, and replicas. You can set this script to automatically attempt to restart roots, links, and replicas when they are not running. This script raises an event when auto-start fails, succeeds, or when roots, links, or replicas are down but the *Auto-start service?* parameter is set to n.

Resource Objects

DFS Namespace Service object

DFS Replication Service object

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if auto-start fails, succeeds, or is disabled. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the Distributed File System service is up, or ◆ **0** -- the service is down. These values are used to report the percentage of time the service is up in any given period. The default is n.
Restart the service(s) if stopped?	Set to **y** to automatically restart the DFS service when it is down. The default is y.
Event severity when restart fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and AppManager cannot restart it. The default is 5 (red event indicator).
Event severity when restart succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service was down and AppManager successfully restarted it. The default is 25 (blue event indicator).
Event severity when restart is set to n	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and the *Auto-start service?* parameter has been disabled. The default is 18 (yellow event indicator).
Severity for an unexpected KS error	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the DFSServiceDown job fails unexpectedly. The default is 35 (magenta event indicator).

5.5 DiskQuotaStatus

Use this Knowledge Script to monitor disk quota status. You can set disk quotas to limit the amount of file server disk space for each user. And you can determine how much disk space can be used before a warning is generated. This value is called the warning level.

For example, if the quota limit is set to 10 MB and the warning level is set to 8 MB, a warning message is generated when a user consumes 8 or more megabytes on the file server. When the user reaches the 10 MB quota limit, the user may or may not be able to save any more files, depending on how disk quotas are configured.

This script raises an event if the specified number or percentage of users reaches the warning level or quota limit.

Resource Objects

Disk Quota settings

Default Schedule

The default interval for this script is **Every hour**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event when a threshold is exceeded. The default is y.
Collect data - number of users over quota limit?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of users over the quota limit. The default is n.
Collect data - number of users over warning level?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of users over the warning level. The default is n.
Collect data - number of users below warning level?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of users under the warning level. The default is n.
Collect data—percentage of users over quota limit?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the percentage of users over the quota limit. The default is n.
Collect data—percentage of users over warning level?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the percentage of users over the warning level. The default is n.
Collect data - percentage of users below warning level?	Set to **y t**o collect data for charts and reports. If enabled, data collection returns the percentage of users below the warning level. The default is n.
Collect data - disk space used for each user? (WARNING - heavy load)	Set to **y** to collect data for charts and reports. If enabled, data collection returns the amount of space being used by each user. The default is n. **NOTE:** Large amounts of CPU resources are required for this parameter to return results.
Number of users over quota limit	Specify the maximum number of users who can be over the quota limit before an event is raised. The default is 1 user.
Number of users over warning level	Specify the maximum number of users who can be over the warning level before an event is raised. The default is 5 users.
Percentage of users over quota limit	Specify the maximum percentage of users who can be over the quota limit before an event is raised. The default is 10%.
Percentage of users over warning level	Specify the maximum percentage of users who can be over the warning level before an event is raised. The default is 20%.
Event severity - number of users over quota limit	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of users exceeded the quota threshold. The default is 8 (red event indicator).
Event severity - number of users over warning level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of users exceeds the warning threshold. The default is 15 (yellow event indicator).

Parameter	How to Set It
Event severity - percentage of users over quota limit	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the percentage of users exceeds the quota threshold. The default is 8 (red event indicator).
Event severity - percentage of users over warning level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the percentage of users exceeds the warning threshold. The default is 15 (yellow event indicator).
Event severity - API error	Set the event severity level, from 1 to 40, to indicate the importance of an event in which network system errors occur. These errors can be hardware, software, or network related. The default is 7 (red event indicator).

5.6 DNSAXFRStat

Use this Knowledge Script to monitor the following AXFR (*zone transfer*, a database replication mechanism) statistics for Master and Secondary Domain Name System (DNS) servers:

- AXFR Request Received (Master)
- AXFR Success Sent (Master)
- AXFR Request Sent (Secondary)
- AXFR Response Received (Secondary)
- AXFR Success Received (Secondary)

This script monitors changes to these statistics since the last interval (delta value), and raises an event if a monitored value exceeds the threshold you set. If you collect data with this script, the AXFR statistics are returned as separate datastreams.

Resource Object

DNS folder

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the number of transfer requests exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns information about the transfer requests sent and received. The default is n.
Transfer requests	Specify the maximum number of transfer requests allowed before an event is raised. The default is 100.

Parameter	How to Set It
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of transfer requests exceeds the threshold. The default is 8 (red event indicator).

5.7 DNSDatabaseNodeMemory

Use this Knowledge Script to monitor the total memory used by the Domain Name Service (DNS) service for database nodes. This script raises an event if the memory used by DNS database nodes (in KB) exceeds the threshold you set.

Resource Object

DNS folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the amount of memory used by the DNS service exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the size of database node memory. The default is n.
Memory used (KB)	Specify the maximum amount of memory that can be used by DNS database nodes before an event is raised. The default is 800 KB.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the amount of memory used by the DNS service exceeds the threshold. The default is 8 (red event indicator).

5.8 DNSDynaUpdateError

Use this Knowledge Script to monitor the number of Domain Name Service (DNS) dynamic update errors. This script checks for two types of errors:

- Dynamic updates rejected by the DNS server
- Dynamic update timeouts of the DNS server

This script raises an event if the number of dynamic update errors in the interval exceeds the threshold you set.

The dynamic update mechanism allows clients and servers to register DNS domain names and IP address mappings to a DNS server.

Resource Object

DNS folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the number of dynamic update errors exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of dynamic updates rejected and the number of dynamic update timeouts in the interval. The default is n.
Dynamic update errors	Specify the maximum number of dynamic update errors that can occur in an interval before an event is raised. The default is 2.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of dynamic update errors exceeds the threshold. The default is 8 (red event indicator).

5.9 DNSDynaUpdateStat

Use this Knowledge Script to monitor DNS server dynamic update activity. This script raises an event if the dynamic update queue length exceeds the threshold you set. A long queue usually indicates that the DNS server is overloaded and cannot process the update in a timely manner.

The dynamic update mechanism allows clients and servers to register DNS domain names and IP address mappings to a DNS server.

Resource Object

DNS folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the number of dynamic updates in the queue exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the queue length for dynamic updates. To further control the data returned, specify the data collection mode to use. By default, data is not collected.
Data collection mode to use	Specify the type of data you want to collect. The following entries are valid: ◆ **1** - to generate one datastream that records the update queue length. The data detail message describes the number updates received and written per second. ◆ **2** - to generate one datastream that records the update queue length, but without the detail message. ◆ **3** - to generate one datastream that tracks the update queue length, and three additional datastreams for the No Operation rate, the rate at which updates are being received, and the rate at which updates are being written to the database. The default is 1 (one datastream and detail message).
Dynamic update queue length	Specify the maximum number of dynamic updates that can be in the queue in an interval before an event is raised. The default is 5 updates.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event if the number of dynamic updates in the queue exceeds the threshold. The default is 8 (red event indicator).

5.10 DNSEventLog

Use this Knowledge Script to periodically scan the Domain Name Service (DNS) Server log for DNS events matching the criteria you specify.

In the first interval, the value you specify for the *Start with events in past N hours* parameter determines how far back in the log to check for matching entries. As the script continues to run at subsequent intervals, it checks for any new entries created since the last time the log was checked.

You can further restrict the types of log entries that generate an event in two ways:

- Use the *Monitor for events of type [...]* parameters to search only certain types of events, such as Warning events.
- Use the *Filter the [...] field for* parameters to search only for specific information, such as events associated with a specific user or computer name.

Each time this script runs, it checks the DNS Server log for entries matching your selection criteria and raises an event if matching entries are found. The event detail message returns the text of the log entries found. When this Knowledge Script is set to collect data, it returns the number of log entries found, and the data point detail message returns the text of the log entries.

Resource Object

DNS folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if log entries match your search criteria. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of new event log entries. The graph data detail message contains the text of the log entries. The default is n.
Start with events in past N hours	Set this parameter to determine which events are searched for the *first* time the script is run. Subsequent searches begin where the last search finished. The following entries are valid: ◆ Enter **-1** to search all current and previous DNS Log events during the first interval. ◆ Enter **0** to search only for current events; previous events are not searched. ◆ Enter the number of hours to go back in the DNS Log to scan for matching events. For example, enter 8 to scan the last 8 hours of the DNS Log for matching entries. The default is 0.
Monitor events of type:	Set to **y** for each type of event you want to monitor: ◆ Error ◆ Warning ◆ Information ◆ Success Audit ◆ Failure Audit If you enable data collection or events, and set any of these parameters to n, this script does not raise an event or collect data for that type of log entry. The default is y.

Parameter	How to Set It
Filter the [...] field for	To limit the types of entries that raise events and the type of data that is collected, enter a search string that filters the following fields in the event log: ◆ **Source**. Specify one or more text strings to look for in the Source field. Separate multiple strings with commas. For example: `DNS Server,general`. ◆ **Category**. Specify one or more text strings to look for in the Category field. Separate multiple strings with commas. ◆ **Event ID**. Specify a single event ID or a range of event IDs. Separate multiple entries by commas. For example: `414,1028-1400,4015`. ◆ **User**. Specify a search string to look for events associated with a particular user, for example, `<domain name>\<user name>`. Separate multiple strings with commas. For example: `USA\Tom,USA\Chris,EUROPE\Alex`. ◆ **Computer**. Specify a single or multiple computer names to look for. Separate multiple entries by commas. For example: `SHASTA,MARS`. ◆ **Event Description**. Specify a detail description or keywords in the description. The string can contain spaces, underscores, and periods. Separate multiple entries with commas. For example: `no domain,critical error from the Active Directory` The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:).
Maximum number of entries per event message	Specify the maximum number of entries to be recorded in each event's detail message. If, during any interval when it scans the log, this script finds more entries in the log than can be put into a single event message, it raises multiple events to return all the log entries. The default is 30 entries.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which log entries match your search criteria. You can adjust the severity level based on the types of events you are checking. The default is 8 (red event indicator).

5.11 DNSRecursiveQuery

Use this Knowledge Script to monitor Domain Name Service (DNS) server recursive query activity. This script checks the number of recursive query errors or timeouts per second, and raises an event if the number of recursive query error or timeouts per second exceeds the rate threshold you set.

Resource Object

DNS folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the number of recursive query errors or timeouts exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of recursive queries, failures, and timeouts per second. The default is n.
Recursive query errors/ timeouts	Specify the maximum number of recursive query errors or timeouts per second that can occur before an event is raised. The default is 2.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of recursive query errors or timeouts exceeds the threshold. The default is 8 (red event indicator).

5.12 DNSSecureUpdate

Use this Knowledge Script to monitor secure updates for the Domain Name Service (DNS) server, and check the percentage of update attempts that failed. This script raises an event if the percentage of secure update failures exceeds the threshold you set.

Resource Object

DNS folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event in which the percentage of secure updates exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, returns the following information in separate datastreams: ◆ Secure update failure rate ◆ Secure update failure number ◆ Secure update total number ◆ Secure update received per second The default is n.

Parameter	How to Set It
Secure update failures	Specify the maximum percentage of secure updates that are allowed to fail before raising an event. The default is 2%.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the percentage of secure updates exceeds the threshold. The default is 8 (red event indicator).

5.13 DNSTotalQuery

Use this Knowledge Script to monitor total query activity for a Domain Name Server (DNS) server. This script checks the number of queries received per second and the number of responses sent per second, and raises an event if the query-received rate or the response-sent rate exceeds the threshold you set.

Resource Object

DNS folder

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the number of query transactions exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the total number of queries received per second and the total number of responses sent per second. The default is n.
Queries received/sent per second	Specify the maximum number of queries that can be received or responses that can be sent per second before an event is raised. The default is 10.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of query transactions exceeds the threshold. The default is 15 (yellow event indicator).

5.14 DNSWINSStat

Use this Knowledge Script to monitor the Windows Internet Name Service (WINS) activity for a Domain Name Service (DNS) server. This script checks the number of lookup requests received per second and the number of responses sent per second. This script raises an event if the lookup-received rate, reverse lookup-received rate, response-sent rate, or reverse response-sent rate exceeds the threshold you set.

Resource Object

DNS folder

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the sent or received rates exceed the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns four datastreams: ◆ WINS lookups received per second ◆ WINS responses sent per second ◆ WINS reverse lookups received per second ◆ WINS reverse responses sent per second The default is n.
Lookups received/sent per second	Specify the maximum number of lookups that can be received or responses that can be sent per second before an event is raised. The default is 20 per second.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default is 15 (yellow event indicator).

5.15 DNSZoneTransfer

Use this Knowledge Script to monitor Domain Name Service (DNS) zone transfer activity. You can set a threshold for the number of zone transfer failures in an interval and a threshold for the percentage of zone transfers attempted that fail in an interval. This script raises an event if either the number or the percentage of zone transfer failures exceeds the threshold.

Resource Object

DNS folder

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the number or percentage of zone transfer failures exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns in five separate datastreams: ◆ Number of zone transfer failures ◆ Number of successful zone transfers ◆ Number of requests received ◆ Number of SOA (Start of Authority) requests sent ◆ Percentage of zone transfers that failed The default is n.
Number of zone transfer failures	Specify the maximum number of zone transfer failures that can occur before an event is raised. The default is 5.
Percentage of zone transfer failures	Specify the maximum percentage of zone transfer that can fail before an event is raised. The default is 20%.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number or percentage of zone transfer failures exceeds the threshold. The default severity level is 8 (red event indicator).

5.16 FrsBusy

Use this Knowledge Script to monitor the CPU utilization and various statistics of the File Replication Service (FRS). This script raises an event if a threshold is exceeded.

Resource Object

File Replication Service folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if a threshold is exceeded. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, returns data based on the threshold values you set. The default is n.

Parameter	How to Set It
CPU utilization of the File Replication Service	Specify the maximum percentage of CPU resources the FRS can utilize before an event is raised. The default is 5%.
Number of change orders received	Specify the maximum number of change orders that can be received before an event is raised. The default is 5.
Number of change orders sent	Specify the maximum number of change orders that can be sent before an event is raised. The default is 5.
Number of files installed	Specify the maximum number of replicated files that can be installed locally before an event is raised. The default is 2.
KB of free staging space	Specify the minimum amount of free space in the staging directory that must be available to prevent an event from being raised. The default is 10 KB.
Percentage of free staging space	Specify the minimum percentage of free space in the staging directory that must be available to prevent an event from being raised. The default is 10%.
Number of packets received	Specify the maximum number of packets that can be received before an event is raised. The default is 10.
Number of packets sent	Specify the maximum number of packets that can be sent before an event is raised. The default is 10.
Number of USN records accepted	Specify the maximum number of USN records that can be accepted for replication before an event is raised. The default is 5.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a monitored value exceeds or falls below the threshold. The default is 8 (red event indicator).
Severity for unexpected KS error	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the FrsBusy job fails unexpectedly. The default is 35 (magenta event indicator).

5.17 FrsEventLog

Use this Knowledge Script to periodically scan the Windows File Replication Service (FRS) log for file replication events matching the criteria you specify.

Each time this script runs, it checks the FRS log for entries matching the criteria you specify and raises an event if matching entries are found. The event detail message returns the text of the log entries found. When this script is set to collect data, it returns the number of log entries found, and the data point detail message returns the text of the log entries.

In the first interval, the value you specify for the *Start with events in past N hours* parameter determines how far back in the log to check for matching entries. As the script continues to run at subsequent intervals, it checks for any new entries created since the last time the log was checked.

You can further restrict the types of log entries that generate an event in two ways:

- Use the *Monitor for events of type [...]* parameters to search only certain types of events, such as Warning events.
- Use the *Filter the [...] field for* parameters to search only for specific information, such as events associated with a specific user or computer name.

Resource Object

File Replication Service folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if FRS log entries match your search criteria. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of new FRS log entries. The default is n.
Start with events in past N hours	Set this parameter to determine which events are searched for the *first* time this script is run. Subsequent searches begin where the last search finished. The following entries are valid: ◆ Enter **-1** to search all current and previous File Replication Log events during the first interval. ◆ Enter **0** to search only for current events; previous events are not searched. ◆ Enter the number of hours to go back in the FRS Log to scan for matching events. For example, enter 8 to scan the last 8 hours of the File Replication Log for matching entries. The default is 0.
Monitor for events of type:	Set to **y** for each type of event you want to monitor: ◆ Error ◆ Warning ◆ Information ◆ Success Audit ◆ Failure Audit If you enable data collection or events, and set any of these parameters to n, this script does not raise an event or collect data for that type of log entry. The default is y.

Parameter	How to Set It
Filter the [...] field for	To limit the types of entries that raise events and the type of data that is collected, enter a search string that filters the following fields in the event log: ◆ **Source**. Specify text strings to look for in the Source field. Separate multiple strings with commas. ◆ **Category**. Specify text strings to look for in the Category field. Separate multiple strings with commas. ◆ **Event ID**. Specify a single event ID or a range of event IDs. Separate multiple entries with commas. For example: 414,1028-1400,4015. ◆ **User**. Specify a search string to look for events associated with a particular use, for example, `<domain name>\<user name>`. Separate multiple strings with commas. For example: `USA\Tom,USA\Chris,EUROPE\Alex`. ◆ **Computer**. Specify computer names to look for. Separate multiple entries by commas. For example: SHASTA,MARS. ◆ **Event Description**. Specify a detail description or keywords in the description. The string can contain spaces, underscores, and periods. Separate multiple entries with commas. For example: `no domain,critical error from the Active Directory`. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you are specifying only include criteria, the colon is not necessary.
Maximum number of entries per event message	Specify the maximum number of entries to be recorded in each event's detail message. If, during any interval when it scans the log, this script finds more entries in the log than can be put into a single event message, it raises multiple events to return all the log entries. The default is 30 entries.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which FRS log entries match your search criteria. You can adjust the severity level based on the types of events you are checking for. The default is 8 (red event indicator).

5.18 FrsReplicaError

Use this Knowledge Script to monitor the various error statistics of the File Replication Service (FRS), including errors in authentication, bindings, and packets sent. This script monitors two objects within the FRS:

- The `FileRelicaConn` object monitors performance statistics for the `Replicaconn` object, which defines replica connections for the DFS roots.
- The `FileReplicaSet` object monitors performance statistics for the `Replicaset` object, which defines a replica set.

This script raises an event if a monitored value exceeds the threshold you set.

Resource Object

File Replication Service folder

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if a threshold is exceeded. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns data based on the thresholds you set. The default is n.
The number of FileReplicaConn...	Specify the maximum number of errors that can be encountered by the `FileReplicaConn` object before an event is raised in each of these categories: ...authentications in error ...bindings in error ...packets sent in error
The number of FileReplicaSet...	Specify the maximum number of errors that can be encountered by the `FileReplicaSet` object before an event is raised in each of these categories: ...Authentications in error ...Bindings in error ...DS bindings in error ...DS objects in error ...DS searches in error ...Files installed with errors ...Packets received in error ...Packets sent in error ...Staging Files Generated with errors
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default severity level is 8 (red event indicator).

5.19 FrsServiceDown

Use this Knowledge Script to monitor the up and down status of the File Replication Service (FRS). You can set this script to automatically attempt to restart the service when it is not running. This script raises an event when auto-start fails or succeeds, or when the service is down but the *Auto-start service?*

Resource Object

File Replication Service object

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event when auto-start fails, succeeds, or when the service is down but the *Auto-start service?* parameter is disabled. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the FRS is up ◆ **0** -- the FRS is down These values are used to report the percentage of time the service is up in any given period. The default is n.
Auto-start service?	Set to **y** to automatically restart FRS when it is down. The default is y.
Event severity when auto-start fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and AppManager cannot restart it. The default is 5 (red event indicator).
Event severity when auto-start succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service was down and AppManager successfully restarted it. The default is 25 (blue event indicator).
Event severity when auto-start is set to n	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and the *Auto-start service?* parameter has been disabled. The default is 18 (yellow event indicator).
Severity for an unexpected KS error	Set the level between 1 and 40, to indicate the importance of an event in which the FrsServiceDown job fails unexpectedly. The default is 35 (magenta event indicator).

5.20 GroupPolicyAddRemove

Use this Knowledge Script to check whether a Group Policy has been added to or removed from a target computer. This script raises an event if a Group Policy is added or removed during the monitoring interval.

Resource Object

Group Policy top-level folder

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if a Group Policy is added or removed during the monitoring interval. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- there were no Group Policy changes, or ◆ **0** -- a Group Policy was added or removed during the interval. The default is n.
Event severity when a Group Policy is added	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a Group Policy was added during the monitoring period. The default is 5 (red event indicator).
Event severity when a Group Policy is removed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a Group Policy was removed during the monitoring period. The default is 5 (red event indicator).

5.21 GroupPolicyCount

Use this Knowledge Script to count the number of Group Policies associated with the target server in Active Directory. This script raises an event if the number of Group Policies associated with the server exceeds the threshold you set.

Resource Object

Group Policy top-level folder

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the number of Group Policies exceeds the threshold you set. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of Group Policies found. The default is n.
Number of Group Policies	Specify the maximum number of Group Policies that can be associated with a computer in Active Directory before an event is raised. The default is 5 Group Policies.

Parameter	How to Set It
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of Group Policies exceeds the threshold. The default is 5 (blue event indicator).

5.22 GroupPolicyLinkSnapshot

Use this Knowledge Script to list the links associated with one or multiple Group Policy objects. This script raises an event if Group Policy links are found or not found.

Resource Object

Windows Server 2003 Group Policy object

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if Group Policy links are found or not found. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of Group Policy links found. The default is n.
Event severity when Group Policy link is found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which Group Policy links have been detected and returned in the event detail message. The default is 25 (blue event indicator).
Event severity when Group Policy link is not found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which no Group Policy links have been detected. The default is 5 (red event indicator).

5.23 GroupPolicyRefresh

Use this Knowledge Script to refresh the computer Group Policy or the user Group Policy without restarting the computer or re-entering login information. This script refreshes the computer or user policy for all the Group Policies associated with the target computer.

Resource Object

Group Policy folder

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event for a successful refresh policy?	Set to **y** to raise an event when the refresh succeeds. This script always raises an event when the refresh fails. The default is y.
Refresh computer group policy?	Set to **y** to refresh the computer Group Policy. The default is y.
Refresh user group policy?	Set to **y** to refresh the user Group Policy. The default is n.
Event severity when a refresh is successful	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a refresh of the computer or user Group Policy succeeds. The default is 25 (blue event indicator).
Event severity when a refresh fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a refresh of the computer or user Group Policy fails. The default is 25 (blue event indicator).

5.24 GroupPolicySnapshot

Use this Knowledge Script to list all the Group Policies on the target server in priority order. This script raises an event if Group Policies are found. If you enable data collection, the event detail message lists all the Group Policies sorted in the following priority order:

- Local Group Policies are listed first
- Default Domain Group Policy are listed second
- Default Domain Controller Group Policy are listed last

No event is raised if no Group Policies are found for the target computer.

Resource Object

Group Policy top-level folder

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if Group Policies are found. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of Group Policies found. The default is n.
Event severity when Group Policies are found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which Group Policies are found. The default is 25 (blue event indicator).

5.25 IASServiceDown

Use this Knowledge Script to monitor up and down status of the Internet Authentication Service (IAS). You can set this script to automatically attempt a service restart when it is not running. This script raises an event when auto-start fails, succeeds, or is disabled.

Resource Object

Internet Authentication Service object

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if auto-starts fails, succeeds, or is disabled. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the Internet Authentication Service is up, or ◆ **0** -- the service is down. These values are used to report the percentage of time the service is up in any given period. The default is n.
Auto-start service?	Set to **y** to automatically restart IAS when it is down. The default is y.
Event severity when auto-start fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and AppManager cannot restart it. The default is 5 (red event indicator).
Event severity when auto-start succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service was down and AppManager successfully restarted it. The default is 25 (blue event indicator).

Parameter	How to Set It
Event severity when auto-start is set to n	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and the *Auto-start service?* parameter has been disabled. The default is 18 (yellow event indicator).
Severity for an unexpected KS error	Set the level between 1 and 40, to indicate the importance of an event in which the IASServiceDown job fails unexpectedly. The default is 35 (magenta event indicator).

5.26 LSASSWatch

Use this Knowledge Script to check whether the Kerberos Key Distribution Center service (specifically, the LSASS process) is running or hung. This script also monitors the amount of CPU time the LSASS process is using. You specify the threshold for CPU usage and the number of consecutive times the threshold can be exceeded before raising an event.

This script raises an event if the LSASS process is not running or if the process is using a large amount of CPU over a consecutive number of intervals (known as looping).

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if CPU usage exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the process is running, or ◆ **0** -- the process is down. The default is n.
High CPU usage	Specify the maximum amount of CPU that the LSASS process can consume before an event is raised. The default is 90%.
Consecutive times LSASS has high CPU usage	Specify the consecutive number of intervals the CPU usage of the LSASS process can exceed the threshold before an event is raised. The default is 3 times.
Event severity - Process appears to be hung	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the LSASS process is hung, or looping. The default is 5 (yellow event indicator).

Parameter	How to Set It
Event severity - Process is not running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the LSASS process is not running. The default is 15 (yellow event indicator).

5.27 MSIPackagesChange

Use this Knowledge Script to monitor the programs or components installed or uninstalled using the Microsoft Windows Installer (MSI). This script tracks the changes recorded in the registry under `SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall` and reports the number of MSI packages that were installed, uninstalled, or updated during the monitoring period.

If you enable detailed data collection, the name, version, publication name, and installation date are returned for each MSI package.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Daily**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if an MSI package is installed, uninstalled, or updated. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of currently installed packages. The default is n.
Create data detail message with package information	Set to **y** to generate the data detail message including package name, version, publication name, and installation date. The default is n.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which an MSI package is installed, uninstalled, or updated. The default is 8 (red event indicator).

5.28 PrinterErrors

Use this Knowledge Script to monitor printer-related errors:

- Print job errors (such as, print jobs that are hung because of data transfer problems or paper jams)
- Printer not ready errors
- Printer out of paper errors

You can set a separate threshold for each type of printer error. This script raises an event if any error type exceeds the threshold you set.

NOTE: To run this script successfully, avoid using special characters such as, /, -, and # when defining the printer name on the monitored computers. Also, if you run the Discovery_NT Knowledge Script and then delete a local or network printer, run Discovery_NT again.

Resource Object

Printer object

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if a threshold is exceeded. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the process is running, or ◆ **0** -- the process is down. The default is n.
‘Print job’ errors	Specify the maximum number of print job errors that can occur in an interval before an event is raised. The default is 5 errors.
‘Printer not ready’ errors	Specify the maximum number of printer “not ready” errors that can occur in an interval before an event is raised. The default is 5 errors.
‘Out of paper’ errors	Specify the maximum number of “out of paper” errors that can occur in an interval before an event is raised. The default is 10 errors.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default is 8 (red event indicator).

5.29 PrinterEventLog

Use this Knowledge Script to periodically scan the Windows System log for printer-related events matching the criteria you specify.

Each time this script runs, it checks the System log for entries matching your selection criteria and raises an event if matching entries are found. The event detail message returns the text of the log entries found. When this script is set to collect data, it returns the number of log entries found, and the data point detail message returns the text of the log entries.

In the first interval, the value you specify for the *Start with events in past N hours* parameter determines how far back in the log to check for matching entries. As the script continues to run at subsequent intervals, it checks for any new entries created since the last time the log was checked.

You can further restrict the types of log entries that generate an event in two ways:

- Use the *Monitor for events of type [...]* parameters to search only certain types of events, such as Warning events.
- Use the *Filter the [...] field for* parameters to search only for specific information, such as events associated with a specific user or computer name.

NOTE: To run this script successfully, avoid using special characters such as /, -, and # when defining the printer name on the monitored computers. Also, if you run the Discovery_NT Knowledge Script and then delete a local or network printer, you must run Discovery_NT again.

Resource Object

Printer folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	**How to Set It**
Event?	Set to **y** to raise an event if log entries match your search criteria. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of new event log entries. The default is n.
Start with events in past N hours	Set this parameter to determine which events are searched for the first time the script is run. Subsequent searches begin where the last search finished. The following entries are valid: ◆ Enter -1 to search all current and previous System Log events during the first interval. ◆ Enter 0 to search only for current events; previous events are not searched. ◆ Enter the number of hours to go back in the System Log to scan for matching events. For example, enter 8 to scan the last 8 hours of the System Log for matching entries. The default is 0.

Parameter	How to Set It
Monitor for events of type:	Set to **y** for each type of event you want to monitor: ◆ Error ◆ Warning ◆ Information ◆ Success Audit ◆ Failure Audit If you enable data collection or events, and set any of these parameters to n, this script does not raise an event or collect data for that type of log entry. The default is y.
Filter the [...] field for	To limit the types of entries that raise events and the type of data that is collected, enter a search string that filters the following fields in the event log: ◆ **Category**. Specify text strings to look for in the Category field. Separate multiple strings with commas. ◆ **Event ID**. Specify a single event ID or a range of event IDs. Separate multiple entries by commas. For example: 414,1028-1400,4015. ◆ **User**. Specify a search string to look for events associated with a particular user, for example, `<domain name>\<user name>`. Separate multiple strings with commas. For example: `USA\Tom,USA\Chris,EUROPE\Alex`. ◆ **Computer**. Specify computer names to look for. Separate multiple entries by commas. For example: SHASTA,MARS. ◆ **Event Description**. Specify a detail description or keywords in the description. The string can contain spaces, underscores, and periods. Separate multiple entries with commas. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you are specifying only include criteria, the colon is not necessary.
Maximum number of entries per event message	Specify the maximum number of entries to be recorded in each event's detail message. If, during any interval when it scans the log, the script finds more entries in the log than can be put into a single event message, it raises multiple events to return all the log entries. Default is 30 entries.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the log contains entries that match your search criteria. You can adjust the severity based on the types of events you are checking. The default is 8 (red event indicator).

5.30 PrinterQueue

Use this Knowledge Script to monitor the printer queue length. This script raises an event if the printer queue length exceeds the threshold you set.

This script is not supported on 64-bit systems.

NOTE: To run this script successfully, avoid using special characters such as, /, -, and # when defining the printer name on the monitored computers. Also, if you run the Discovery_NT Knowledge Script and then delete a local or network printer, run Discovery_NT again.

Resource Object

Local or cluster printer object

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the number of print jobs in the queue exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the current printer queue length. The default is n.
Printer queue length	Specify the maximum number of print jobs that can be in the printer queue before an event is raised. The default is 5 jobs.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of print jobs in the queue exceeds the threshold. The default is 8 (red event indicator).

5.31 PrinterUtil

Use this Knowledge Script to monitor printer utilization by tracking the number of bytes printed per second. This script raises an event if the number of bytes per second exceeds the threshold you set.

NOTE: To run this script successfully, avoid using special characters such as, /, -, and # when defining the printer name on the monitored computers. Also, if you run the Discovery_NT Knowledge Script and then delete a local or network printer, run Discovery_NT again.

Resource Object

Local or cluster printer object

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the number of bytes printed per second exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of bytes printed per second. The default is n.
Bytes printed per second	Specify the maximum number of bytes that can be printed per second before an event is raised. The default is 2000 bytes.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of bytes printed per second exceeds the threshold. The default is 8 (red event indicator).

5.32 RemoteStorageEventLog

Use this Knowledge Script to periodically scan the Windows Application log for Remote Storage-related events matching the criteria you specify.

Each time this script runs, it checks the Application log for entries matching your selection criteria and raises an event if matching entries are found. The event detail message returns the text of the log entries found. When this script is set to collect data, it returns the number of log entries found, and the data point detail message returns the text of the log entries.

In the first interval, the value you specify for the *Start with events in past N hours* parameter determines how far back in the log to check for matching entries. As the script continues to run at subsequent intervals, it checks for any new entries created since the last time the log was checked.

You can further restrict the types of log entries that generate an event in two ways:

- Use the *Monitor for events of type [...]* parameters to search only certain types of events, such as Warning events.
- Use the *Filter the [...] field for* parameters to search only for specific information, such as events associated with a specific user or computer name.

Resource Object

Remote Storage folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if log entries match your search criteria. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of new event log entries. The default is n.
Start with events in past N hours	Set this parameter to determine which events are searched for the *first* time the script is run. Subsequent searches begin where the last search finished. The following entries are valid: ◆ Enter -1 to search all current and previous Application Log events during the first interval. ◆ Enter 0 to search only for current events; previous events are not searched. ◆ Enter the number of hours to go back in the Application Log to scan for matching events. For example, enter 8 to scan the last 8 hours of the Application Log for matching entries. The default is 0.
Monitor for events of type:	Set to **y** for each type of event you want to monitor: ◆ Error ◆ Warning ◆ Information ◆ Success Audit ◆ Failure Audit If you enable data collection or events, and set any of these parameters to n, this script does not raise an event or collect data for that type of log entry. The default is y.

Parameter	How to Set It
Filter the [...] field for	To limit the types of entries that raise events and the type of data that is collected, enter a search string that filters the following fields in the event log: ◆ **Category**. Specify text strings to look for in the Category field. Separate multiple strings with commas. ◆ **Event ID**. Specify a single event ID or a range of event IDs. Separate multiple entries by commas. For example: 414,1028-1400,4015. ◆ **User**. Specify a search string to look for events associated with a particular user, for example, `<domain name>\<user name>`. Separate multiple strings with commas. For example: `USA\Tom,USA\Chris,EUROPE\Alex`. ◆ **Computer**. Specify a single or multiple computer names to look for. Separate multiple entries by commas. For example: SHASTA,MARS. ◆ **Event Description**. Specify a detail description or keywords in the description. The string can contain spaces, underscores, and periods. Separate multiple entries with commas. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you are specifying only include criteria, the colon is not necessary.
Maximum number of entries per event message	Specify the maximum number of entries to be recorded in each event's detail message. If, during any interval when it scans the log, this script finds more entries in the log than can be put into a single event message, it raises multiple events to return all the log entries. The default is 30 entries.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which log entries match your search criteria. You can adjust the severity based on the types of events you are checking. The default is 8 (red event indicator).

5.33 RemoteStorageServiceDown

Use this Knowledge Script to monitor the up and down status of the Remote Storage Service. You can set this script to automatically attempt to restart the service when it is not running. This script raises an event when auto-start fails, succeeds, or is disabled.

Resource Object

Remote Storage folder

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if auto-start fails, succeeds, or is disabled. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the Remote Storage services are up, or ◆ **0** -- any service is down. These values are used to report the percentage of time the service is up in any given period. The default is n.
Auto-start service?	Set to **y** to automatically restart any Remote Storage service when it is down. The default is y.
Event severity when auto-start fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which service is down and AppManager cannot restart it. The default is 5 (red event indicator).
Event severity when auto-start succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service was down and AppManager successfully restarted it. The default is 25 (blue event indicator).
Event severity when auto-start is set to n	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and the *Auto-start service?* parameter has been disabled. The default is 18 (yellow event indicator).
Severity for an unexpected KS error	Set the level between 1 and 40, to indicate the importance of an event in which the RemoteStorageServiceDown job fails unexpectedly. The default is 35 (magenta event indicator).

5.34 RSVPEventLog

Use this Knowledge Script to periodically scan the Windows Application log for QoS/RSVP-related events matching the criteria you specify.

Each time this script runs, it checks the Application log for entries matching your selection criteria and raises an event if matching entries are found. The event detail message returns the text of the log entries found. When data collection is enabled, the job returns the number of log entries found, and the data point detail message returns the text of the log entries.

In the first interval, the value you specify for the *Start with events in past N hours* parameter determines how far back in the log to check for matching entries. As the script continues to run at subsequent intervals, it checks for any new entries created since the last time the log was checked.

You can further restrict the types of log entries that generate an event in two ways:

- Use the *Monitor for events of type [...]*

  parameters to search only certain types of events, such as Warning events.
- Use the *Filter the [...] field for* parameters to search only for specific information, such as events associated with a specific user or computer name.

Resource Object

QoS folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if log entries match your search criteria. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of new event log entries. The graph data detail message returns the text of the log entries. The default is n.
Start with events in past N hours	Set this parameter to determine which events are searched for the *first* time the Knowledge Script is run. Subsequent searches begin where the last search finished. The following entries are valid: ◆ Enter -1 to search all current and previous Application Log events during the first interval. ◆ Enter 0 to search only for current events; previous events are not searched. ◆ Enter the number of hours to go back in the Application Log to scan for matching events. For example, enter 8 to scan the last 8 hours of the Application Log for matching entries. The default is 0.
Monitor for events of type:	Set to **y** for each type of event you want to monitor: ◆ Error ◆ Warning ◆ Information ◆ Success Audit ◆ Failure Audit If you enable data collection or events, and set any of these parameters to n, this script does not raise an event or collect data for that type of log entry. The default is y.

Parameter	How to Set It
Filter the [...] field for	To limit the types of entries that raise events and the type of data that is collected, enter a search string that filters the following fields in the event log: ◆ **Category**. Specify text strings to look for in the Category field. Separate multiple strings with commas. ◆ **Event ID**. Specify a single event ID or a range of event IDs. Separate multiple entries by commas. For example: 414,1028-1400,4015. ◆ **User**. Specify a search string to look for events associated with a particular user, for example, `<domain name>\<user name>`. Separate multiple strings with commas. For example: `USA\Tom,USA\Chris,EUROPE\Alex`. ◆ **Computer**. Specify computer names to look for. Separate multiple entries by commas. For example: SHASTA,MARS. ◆ **Event Description**. Specify a detail description or keywords in the description. The string can contain spaces, underscores, and periods. Separate multiple entries with commas. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you are specifying only include criteria, the colon is not necessary.
Maximum number of entries per event message	Specify the maximum number of entries to be recorded in each event's detail message. If, during any interval when it scans the log, the script finds more entries in the log than can be put into a single event message, it raises multiple events to return all the log entries. The default is 30 entries.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which log entries match your search criteria. You can adjust the severity based on the types of events you are checking. The default is 8 (red event indicator).

5.35 RSVPServiceDown

Use this Knowledge Script to monitor the Windows QoS/RSVP service. You can set this script to automatically attempt to restart the service when it is not running. This script raises an event if the service is not running, if the attempt to restart it fails, or if the service is down and the *Auto-start service?* parameter is set to n.

Resource Object

QoS folder

Default Schedule

The default interval for this script is **Every 5 minutes.**

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the service is not running, if the restart attempt fails, or if the service is down and the *Auto-start service?* parameter is set to n. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: - **100** -- the QoS/RSVP service is up, or - **0** -- the service is down. These values are used to report the percentage of time the service is up in any given period. The default is n.
Auto-start service?	Set to **y** to automatically restart the QoS/RSVP service when it is down. The default is y.
Event severity when auto-start fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which service is down and AppManager cannot restart it. The default is 5 (red event indicator).
Event severity when auto-start succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service was down and AppManager successfully restarted it. The default is 25 (blue event indicator).
Event severity when auto-start is set to n	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and the *Auto-start service?* parameter is set to n. The default is 18 (yellow event indicator).
Severity for an unexpected KS error	Set the level between 1 and 40, to indicate the importance of an event in which the RSVPServiceDown job fails unexpectedly. The default is 35 (magenta event indicator).

5.36 SMTPEventLog

Use this Knowledge Script to periodically scan the Windows System log for SMTP-related events matching the criteria you specify.

Each time this script runs, it checks the System log for entries matching your selection criteria and raises an event if matching entries are found. The event detail message returns the text of the log entries found. When this script is set to collect data, it returns the number of log entries found, and the data point detail message returns the text of the log entries.

In the first interval, the value you specify for the *Start with events in past N hours* parameter determines how far back in the log to check for matching entries. As the script continues to run at subsequent intervals, it checks for any new entries created since the last time the log was checked.

You can further restrict the types of log entries that generate an event in two ways:

- Use the *Monitor for events of type [...]* parameters to search only certain types of events, such as Warning events.
- Use the *Filter the [...] field for* parameters to search only for specific information, such as events associated with a specific user or computer name.

Resource Object

SMTP folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if log entries match your search criteria. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of new event log entries. The default is n.
Start with events in past N hours	Set this parameter to determine which events are searched for the *first* time this script is run. Subsequent searches begin where the last search finished. The following entries are valid: ◆ Enter -1 to search all current and previous System Log events during the first interval. ◆ Enter 0 to search only for current events; previous events are not searched. ◆ Enter the number of hours to go back in the System Log to scan for matching events. For example, enter 8 to scan the last 8 hours of the System Log for matching entries. The default is 0.
Monitor for events of type:	Set to **y** for each type of event you want to monitor: ◆ Error ◆ Warning ◆ Information ◆ Success Audit ◆ Failure Audit If you enable data collection or events, and set any of these parameters to n, this script does not raise an event or collect data for that type of log entry. The default is y.

Parameter	How to Set It
Filter the [...] field for	To limit the types of entries that raise events and the type of data that is collected, enter a search string that filters the following fields in the event log: ◆ **Category**. Specify text strings to look for in the Category field. Separate multiple strings with commas. ◆ **Event ID**. Specify a single event ID or a range of event IDs. Separate multiple entries by commas. For example: `414,1028-1400,4015`. ◆ **User**. Specify a search string to look for events associated with a particular user, for example, `<domain name>\<user name>`. Separate multiple strings with commas. For example: `USA\Tom,USA\Chris,EUROPE\Alex`. ◆ **Computer**. Specify computer names to look for. Separate multiple entries by commas. For example: `SHASTA,MARS`. ◆ **Event Description**. Specify a detail description or keywords in the description. The string can contain spaces, underscores, and periods. Separate multiple entries with commas. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you are specifying only include criteria, the colon is not necessary.
Maximum number of entries per event message	Specify the maximum number of entries to be recorded in each event's detail message. If, during any interval when it scans the log, this script finds more entries in the log than can be put into a single event message, it raises multiple events to return all the log entries. The default is 30 entries.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which log entries match your search criteria. You can adjust the severity based on the types of events you are checking. The default is 8 (red event indicator).

5.37 SMTPQueues

Use this Knowledge Script to monitor the length of the following SMTP queues:

- Categorizer queue
- Local queue
- Local Retry queue
- Remote queue
- Remote Retry queue

This script raises an event if any queue length exceeds the threshold you set.

Resource Object

SMTP folder

Default Schedule

The default interval for this script is **Every 10 minutes.**

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if a queue length exceeds the threshold you set. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the queue length for each SMTP queue. The default is n.
SMTP Categorizer queue length	Specify the maximum number of processes that can be in the SMTP Categorizer queue before an event is raised. The default is 10.
SMTP Local queue length	Specify the maximum number of processes that can be in the SMTP Local queue before an event is raised. The default is 10.
SMTP Local Retry queue length	Specify the maximum number of processes that can be in the SMTP Local Retry queue before an event is raised. The default is 5.
SMTP Remote queue length	Specify the maximum number of processes that can be in the SMTP Remote queue before an event is raised. The default is 10.
SMTP Remote Retry queue length	Specify the maximum number of processes that can be in the SMTP Remote Retry queue before an event is raised. The default is 5.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a queue length exceeds the threshold you set. The default is 8 (red event indicator).

5.38 SMTPServiceDown

Use this Knowledge Script to monitor the up and down status of the SMTP service. You can set this script to automatically attempt to restart the service when it is not running. This script raises an event when auto-start fails, succeeds, or is disabled.

Resource Object

SMTP Service object

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event when auto-start fails, succeeds, or is disabled. The default is y.

Parameter	How to Set It
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the SMTP service is up, or ◆ **0** -- the service is down. These values are used to report the percentage of time the service is up in any given period. The default is n.
Auto-start service?	Set to **y** to automatically restart the SMTP service when it is down. The default is y.
Event severity when auto-start fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which service is down and AppManager cannot restart it. The default is 5 (red event indicator).
Event severity when auto-start succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service was down and AppManager successfully restarted it. The default is 25 (blue event indicator).
Event severity when auto-start is set to n	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and the *Auto-start service?* parameter is set to n. The default is 18 (yellow event indicator).
Severity for an unexpected KS error	Set the level between 1 and 40, to indicate the importance of an event in which the SMTPServiceDown job fails unexpectedly. The default is 35 (magenta event indicator).

6 WIN2003 Knowledge Scripts

The WIN2003 category provides Knowledge Scripts for monitoring computers running Microsoft Windows Server 2003 or later.

From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. Or in the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
ActivationGracePeriod	Monitors the number of days that remain before you are required to activate the Windows operating system.
AUDownLoaded	Monitors the number of critical Windows updates that have been downloaded but not yet installed
AUOptionChange	Monitors the Automatic Updates options.
AUServiceDown	Monitors the status of the Automatic Updates service
AUVerifyHotFix	Verifies whether specified hotfixes have been installed.
BITSJobProgress	Monitors the progress of Background Intelligent Transfer Service jobs by measuring the total bytes transferred and the total files transferred.
BITSJobsActive	Monitors the number of active Background Intelligent Transfer Service jobs.
BITSJobsError	Monitors the total number of Background Intelligent Transfer Service jobs that are in an error state.
BITSJobState	Indicates whether a Background Intelligent Transfer Service job is in error state or not in error state.
BITSJobStats	Monitors the number of times a Background Intelligent Transfer Service job is interrupted by network failure or server unavailability.
BITSServiceDown	Monitors the status of the Background Intelligent Transfer Service.
CLRConnectionPools	Monitors SQL connection pools in managed .NET applications.
CLRContention	Monitors the thread contention rate and thread queue length in managed .NET applications.
CLRExceptions	Monitors exceptions that managed .NET applications raise.
CLRHeap	Monitors heap memory use in managed .NET applications.
CLRJit	Monitors JIT (just-in-time) compilation in managed .NET applications.
CLRMemProfile	Monitors total garbage collection in managed .NET applications.
CLRContention	Monitors network activity in managed .NET applications.
CLRRemoting	Monitors remote procedure call (RPC) activity in managed .NET applications.
CLRThreads	Monitors thread use in managed .NET applications.

Knowledge Script	What It Does
CLRNetworking	Monitors the distributed COM (DCOM) application list.
FaxActivity	Monitors the number of faxes, fax pages, and fax bytes sent and received.
FaxEventLog	Scans the Windows Application event log for entries created by the Microsoft Fax service that match the criteria you specify.
FaxServiceDown	Monitors the status of the Microsoft Fax service.
FaxTotalFailed	Monitors the total number of failed faxes, failed outgoing connections, and failed receptions.
FaxTotalTime	Monitors the number of minutes the Microsoft Fax service spends receiving and sending faxes.
OpenSystemSlots	Monitors the number of available system (PCI) slots.
PNPDeviceChange	Monitors the plug-and-play device list for any device that has been added or removed since the script was last run.
PNPDeviceErrors	Monitors the number of plug-and-play devices that have a status of “error.”
PrinterStuckJobs	Monitors jobs that are stuck in the printer queue.
SRDiskPercent	Monitors the percentage of space on a disk available for the System Restore service.
SREventLog	Scans the Windows Application event log for entries created by the System Restore service that match the criteria you specify.
SRLifeInterval	Monitors the number of days the System Restore service preserves System Restore points
SRPoints	Monitors the number of System Restore points that are being preserved by the System Restore service.
SRScheduledInterval	Monitors the interval at which scheduled System Restore points are created during both current and global sessions.
SRServiceDown	Monitors the status of the System Restore service.

6.1 ActivationGracePeriod

Use this Knowledge Script to monitor the number of days that remain before you are required to activate the system. After you install Windows, you must activate your installation by contacting Microsoft. If you do not activate within a certain number of days, you can no longer log on.

This script raises an event if the number of days remaining falls below the threshold you set.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every 24 hours**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if number of days in grace period falls below threshold?	Set to **y** to raise an event when the number of days remaining before you must activate the system falls below the threshold you set. The default is y.
Collect data for days remaining in grace period?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of days remaining before you must activate your system. The default is n.
Threshold - Minimum days remaining in grace period	Specify the minimum number of days that must remain before activation of the system is required to prevent an event from being raised. The default is 1 day.
Event severity when number of days falls below threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of days remaining before system activation is required falls below the threshold. The default is 5 (red event indicator).

6.2 AUDownLoaded

Use this Knowledge Script to monitor the total number of critical updates from the Windows update Web site that have been downloaded but not yet installed. If you configure Automatic Updates to prompt you before installing downloaded updates, then you may accumulate a large number of downloaded updates that have not yet been installed. This script raises an event if the total number of downloaded, but uninstalled, updates exceeds the threshold you set.

Resource Object

Automatic Updates folder

Default Schedule

The default interval for this script is **Every 24 hours**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if threshold is exceeded?	Set to **y** to raise an event if the total number of updates downloaded exceeds the threshold you set. The default is y.
Collect data for updates downloaded but not installed?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of updates that have been downloaded but not yet installed. The default is n.
Threshold - Maximum total downloaded updates	Specify the maximum number of updates that can be downloaded but not installed before an event is raised. The default is 10.

Parameter	How to Set It
Event severity when downloaded updates exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of updates that can be downloaded but not installed exceeds the threshold. The default is 8 (red event indicator).

6.3 AUOptionChange

Use this Knowledge Script to monitor the Automatic Updates options. You can configure several Automatic Updates options:

- Disabled
- Notify user before download and install
- Download automatically and notify before install
- Automatically download and install on schedule

This script raises an event is raised if an option setting is changed.

Resource Object

Automatic Updates folder

Default Schedule

The default interval for this script is **Every hour**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if Automatic Updates option changed?	Set to **y** to raise an event when an option setting is changed. The default is y.
Collect data for Automatic Updates options changed?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the current selections for the Automatic Updates options. The default is n.
Event severity when option changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which Automatic Updates options have changed. The default is 8 (red event indicator).

6.4 AUServiceDown

Use this Knowledge Script to monitor the status of the Automatic Updates service. This script raises an event if the service is down, and can, optionally, attempt to restart the service when it is not running.

Resource Object

Automatic Updates Service object

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if Automatic Updates service is down?	Set to **y** to raise an event when the Automatic Updates service is down. The default is y.
Collect data for service status?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ 100 -- service is running, or ◆ 0 -- service is not running. The default is n.
Auto-start service if down?	Set to **y** to automatically restart the service when it is down. The default is y.
Event severity when auto-start fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and AppManager cannot restart it. The default is 5 (red event indicator).
Event severity when auto-start succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service was down and AppManager successfully restarted it. The default is 25 (blue event indicator).
Event severity when service down and auto-start disabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and AppManager has been set to not restart the service. The default is 18 (yellow event indicator).

6.5 AUVerifyHotFix

Use this Knowledge Script to verify whether the hotfixes you specify have been installed. This script raises an event when specified hotfixes have not been installed and when specified hotfixes are installed.

Resource Object

Automatic Updates folder

Default Schedule

The default interval for this script is once.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Hotfix articles to monitor	Specify the ID for each hotfix you want to verify. You can enter multiple IDs separated by commas. **NOTE:** The ID for each hotfix is case-sensitive. For example, Q123456 is *not* a match for q123456. If your entry does not exactly match the hotfix ID, an event is raised indicating the hotfix in question has not been installed even if it has been installed.
Raise event when hotfixes are installed?	Set to **y** to raise an event when the hotfixes you specify have been installed. The default is y.
Event severity when hotfixes are installed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which specified hotfixes have been installed. The default is 15 (yellow event indicator).
Raise event if hotfixes have not been installed?	Set to **y** to raise an event when the hotfixes you specify have not yet been installed. The default is y.
Event severity when hotfixes have not been installed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the hotfixes you specified have not been installed. The default is 8 (red event indicator).
Collect data for installed hotfixes?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of installed hotfixes. The default is n.

6.6 BITSJobProgress

Use this Knowledge Script to monitor the progress of Background Intelligent Transfer Service (BITS) jobs by measuring the total bytes transferred and the total files transferred. This script raises an event if the total number of bytes or files transferred exceeds the thresholds you set.

Prerequisites

For Windows Sever 2008 and Windows 2008 R2 or later, you can run this script on both 32-bit and 64-bit platforms. For older versions of Windows, you can only run this script on 32-bit platforms.

This script requires version 2.0 or later of Windows .NET Framework on the computer you want to monitor.

Resource Objects

BITS Jobs object

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if threshold is exceeded?	Set to **y** to raise an event if the total bytes transferred or the total files transferred exceed the threshold you set. The default is y.
Collect data for total bytes and files transferred by BITS jobs?	Set to **y** to collect data for charts and reports. If enabled, data collection returns total bytes transferred and total files transferred for a BITS job. The default is n.
Threshold - Maximum number of bytes transferred by BITS jobs	Specify the maximum number of bytes that can be transferred before an event is raised. The default is 200 bytes.
Threshold - Maximum number of files transferred by BITS jobs	Specify the maximum number of files that can be transferred before an event is raised. The default is 200 files.
Event severity when threshold exceeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of transferred bytes or jobs exceeds the threshold you set. The default is 8 (red event indicator).

6.7 BITSJobsActive

Use this Knowledge Script to monitor the number of active BITS jobs. This script raises an event when the total number of active jobs exceeds the threshold.

Prerequisites

For Windows Sever 2008 and Windows 2008 R2 or later, you can run this script on both 32-bit and 64-bit platforms. For older versions of Windows, you can only run this script on 32-bit platforms.

This script requires version 2.0 or later of Windows .NET Framework on the computer you want to monitor.

Resource Object

BITS folder

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if number of active BITS jobs exceeds threshold?	Set to **y** to raise an event when the total number of active BITS jobs exceeds the threshold you set. The default is y.
Collect data for number of active BITS jobs?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of jobs submitted and the number of incomplete jobs during the interval you specify. The default is n.
Event severity when number of active BITS jobs exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of active BITS jobs exceeds the threshold. The default is 8 (red event indicator).
Threshold - Maximum number of active BITS jobs	Specify the maximum number of BITS jobs that can be active before an event is raised. The default is 200 jobs.

6.8 BITSJobsError

Use this Knowledge Script to monitor the total number of BITS jobs that are in an error state. This script raises an event if the number of BITS jobs with the status of error exceeds the threshold.

Prerequisites

For Windows Sever 2008 and Windows 2008 R2 or later, you can run this script on both 32-bit and 64-bit platforms. For older versions of Windows, you can only run this script on 32-bit platforms.

This script requires version 2.0 or later of Windows .NET Framework on the computer you want to monitor.

Resource Object

BITS folder

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if number of BITS jobs in error state exceeds threshold?	Set to **y** to raise an event when the number of BITS jobs with the status of error exceeds the threshold you set. The default is y.

Parameter	How to Set It
Collect data for BITS jobs in error state?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of BITS jobs that have the status of error. The default is n.
Event severity when BITS jobs in error state exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of BITS job with a status of error exceeds the threshold. The default is 8 (red event indicator).
Threshold - Threshold - Maximum number of BITS jobs in error state	Specify the maximum number of BITS jobs that can have a status of error before an event is raised. The default is 10.

6.9 BITSJobState

Use this Knowledge Script to monitor the state of a BITS job. This script raises an event if a BITS job is in error state and when it is not in error state.

Prerequisites

For Windows Sever 2008 and Windows 2008 R2 or later, you can run this script on both 32-bit and 64-bit platforms. For older versions of Windows, you can only run this script on 32-bit platforms.

This script requires version 2.0 or later of Windows .NET Framework on the computer you want to monitor.

Resource Object

BITS Job object

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if BITS job is in error state?	Set to **y** to raise an event when the BITS job is in the error state. The default is y.
Raise event if BITS job is not in error state?	Set to**y** to raise an event when the BITS job is not in the error state. The default is y.
Collect data for status of BITS job?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the state of the BITS job: canceled, executing, completed, or error. The default is n.
Event severity when BITS job not in error state	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the BITS job is not in an error state. The default is 25 (blue event indicator).

6.10 BITSJobStats

Use this Knowledge Script to monitor the number of times the BITS job is interrupted by network failure or server unavailability. This script raises an event is raised if the total number of times the BITS job is interrupted exceeds the threshold.

Prerequisites

For Windows Sever 2008 and Windows 2008 R2 or later, you can run this script on both 32-bit and 64-bit platforms. For older versions of Windows, you can only run this script on 32-bit platforms.

This script requires version 2.0 or later of Windows .NET Framework on the computer you want to monitor.

Resource Object

BITS Jobs object

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if interruptions exceed threshold?	Set to **y** to raise an event when the number of times the BITS job is interrupted exceeds the threshold you set. The default is y.
Collect data for number of BITS job interruptions?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of times the BITS job is interrupted. The default is n.
Event severity when interruptions exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of times the BITS job is interrupted exceeds the threshold. The default is 8 (red event indicator).
Threshold - Threshold - Maximum number of BITS job interruptions	Specify the maximum number of times the BITS job can be interrupted before an event is raised. The default is 10 times.

6.11 BITSServiceDown

Use this Knowledge Script to monitor the status of the BITS service. This script raises an event if the service is down, and can restart the service when it is not running.

Resource Objects

BITS service object

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if BITS service is down?	Set to **y** to raise an event when the BITS service is down. The default is y.
Collect data for BITS service status?	Set to **y** to collect data for charts and reports. If enabled, data collection returns a value of 100 if the service is running, and a value of 0 if the service is not running. The default is n.
Auto-start service?	Set to **y** to automatically restart the service when it is down. The default is y.
Event severity when auto-start fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and AppManager cannot restart it. The default is 5 (red event indicator).
Event severity when auto-start succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service was down and AppManager successfully restarted it. The default is 25 (blue event indicator).
Event severity when service down and auto-start disabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and AppManager has been set to not restart the service. The default is 18 (yellow event indicator).

6.12 CLRConnectionPools

Use this Knowledge Script to monitor SQL connection pools in managed .NET applications. Connection pools are caches of stored database connections that are reused, eliminating the need to create new connections each time a new request is received.

This script raises an event if either of the following conditions exists:

- The highest number of pooled connections in a session exceeds a specified threshold.
- The total number of connection attempts that failed exceeds a specified threshold.

NOTE: The monitored .NET applications must be running at the time of discovery so their resource objects can be discovered. The applications must also be running for this script to collect data.

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Resource Objects

SQL Client Managed Applications folder

SQL Client Managed Applications object

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Raise event if the managed application is not running?	Set to **Yes** to raise an event if the managed application is not running. The default is Yes.
Event severity when application is not running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the managed application is not running. The default is 20 (yellow event indicator).
Monitor Peak Pooled Connection	
Event Notification	
Raise event if peak pooled connections exceed threshold?	Set to **Yes** to raise an event if the number of pooled connections exceeds the threshold you set. The default is Yes.
Threshold - Threshold - Maximum peak pooled connections	Specify the maximum number of pooled connections that can occur before an event is raised. The default is 64 connections.
Event severity when peak pooled connections exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of pooled connections exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for peak pool connection?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the maximum number of pool connections that occurred during the monitoring interval. The default is unselected.
Monitor Failed Connection	
Event Notification	
Raise event if failed connections exceed threshold?	Set to **Yes** to raise an event if the number of failed connections exceeds the threshold you set. The default is Yes.
Threshold - Maximum failed connections	Specify the maximum of connections that can fail before an event is raised. The default is 4 connections.
Event severity when failed connections exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of failed connections exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for failed connections?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the number of connections that failed during the monitoring interval. The default is unselected.
Monitor General Connection Pool	
Data Collection	
Collect data for current connection pool?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the number of current connection pools. The default is unselected.

Parameter	How to Set It
Collect data for pooled connections?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the number of pooled connections. The default is unselected.

6.13 CLRContention

Use this Knowledge Script to monitor the thread contention rate and thread queue length in managed .NET applications.

This script raises an event if one of the following conditions exists:

- The contention rate (in seconds) exceeds a specified threshold. Contention occurs when numerous threads compete unsuccessfully to acquire managed locks at run time.
- The thread queue length exceeds a specified threshold. Thread queue length is a measurement of all threads that are waiting to acquire a managed lock on an application.

NOTE: The monitored .NET applications must be running at the time of discovery so their resource objects can discovered. The applications must also be running for this script to collect data.

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Resource Objects

Managed Applications folder

Managed Applications object

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Raise event if application not running?	Set to **Yes** to raise an event if the managed application is not running. The default is Yes.
Event severity when application not running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the managed application is not running. The default is 20 (yellow event indicator).
Monitor Contention Rate	
Event Notification	
Raise event if contention rate exceeds threshold?	Set to **Yes** to raise an event if the contention rate exceeds the threshold you set. The default is Yes.

Parameter	How to Set It
Threshold - Maximum contention rate	Specify the maximum number of contentions that can occur per second before an event is raised. The default is one contention per second.
Event severity when contention rate exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the contention rate exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for contention rate?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the contention rate for the monitoring interval. The default is unselected.
Monitor Thread Queue Length	
Event Notification	
Raise event if thread queue length exceeds threshold?	Set to **Yes** to raise an event if the thread queue length exceeds the threshold you set. The default is Yes.
Threshold - Maximum thread queue length	Specify the maximum number of threads that can be in the queue before an event is raised. The default is 10 threads.
Event severity when thread queue length exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of threads in the queue exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for thread queue length?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the number of threads in queue during the monitoring interval. The default is unselected.

6.14 CLRExceptions

Use this Knowledge Script to monitor exceptions that managed .NET applications raise. Exceptions are errors in a program that cause it to branch to a new routine.

This script raises an event if one of the following conditions occurs:

- The total number of .NET exceptions (or converted exceptions) that occur since the application started exceeds a specified threshold.
- The number of .NET exceptions (or converted exceptions) that occur per second exceeds a specified threshold.

NOTE: The monitored .NET applications must be running at the time of discovery so their resource objects can discovered. The applications must also be running for this script to collect data.

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Resource Objects

Managed Applications folder

Managed Applications object

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Raise event if application not running?	Set to **Yes** to raise an event if the managed application is not running. The default is Yes.
Event severity when application not running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the managed application is not running. The default is 20 (yellow event indicator).
Monitor Exception Count	
Event Notification	
Raise event if exceptions exceed threshold?	Set to **Yes** to raise an event if the number of exceptions exceeds the threshold you set. The default is Yes.
Threshold - Maximum number of exceptions	Specify the maximum number of exceptions that can occur before an event is raised. The default is 5 exceptions.
Event severity when exceptions exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of exceptions exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for number of exceptions?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the number of exceptions that occurred during the monitoring period. The default is unselected.
Monitor Exception Rate	
Event Notification	
Raise event if exception rate exceeds threshold?	Set to **Yes** to raise an event if the exception rate exceeds the threshold you set. The default is Yes.
Threshold - Maximum exception rate	Specify the maximum number exceptions allowed per second before an event is raised. The default is 10 exceptions per second.
Event severity when exception rate exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the exception rate exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for exception rate?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the exception rate during the monitoring interval. The default is unselected.

6.15 CLRHeap

Use this Knowledge Script to monitor heap memory use in managed .NET applications.

This script raises an event if one of the following occurs:

- Full garbage collection levels have changed. Total garbage collection frees memory slots in sections of unused memory and is composed of partial garbage collection (where memory freeing processes can be interrupted) and full garbage collection (where memory freeing processes cannot be interrupted). In partial garbage collection, only the most recently allocated objects (Gen 0 objects) are counted. In full garbage collection, older objects (Gen 1 and up) are counted.
- Special heap memory use is greater than a specified percentage of the total garbage collection heap. In some cases, garbage collection software allocates large objects (>20 KB) directly to an area in memory known as the special heap, bypassing generation object promotion.

NOTE: The monitored .NET applications must be running at the time of discovery so their resource objects can discovered. The applications must also be running for this script to collect data.

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Resource Objects

Managed Applications folder

Managed Applications object

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Raise event if application not running?	Set to **Yes** to raise an event if the managed application is not running. The default is Yes.
Event severity when application not running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the managed application is not running. The default is 20 (yellow event indicator).
Monitor heap memory usage	
Event Notification	
Raise event if special heap size exceeds threshold?	Set to **Yes** to raise an event if the special heap size exceeds the threshold you set. The default is Yes.
Threshold - Maximum size of special heap (as % of garbage heap)	Specify the maximum size of the special heap (as a percentage of the total heap) that can occur before an event is raised. The default is 80%. Total heap is the number of bytes in all heaps.

Parameter	How to Set It
Event severity when size of special heap exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the size of the special heap exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for total heap size?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the size of the total heap during the monitoring interval. The default is unselected.
Monitor Full Garbage Collection	
Event Notification	
Raise event if frequency of full garbage collection changes?	Set to **Yes** to raise an event if the number of full garbage collections has changed since the last time the script ran. The default is Yes.
Event severity when full garbage collection frequency changes	Set the event an event in which the number of full garbage collections has changed. The default is 15 (yellow event indicator).

6.16 CLRJit

Use this Knowledge Script to monitor JIT (just-in-time) compilation in managed .NET applications.

This script raises an event if one of the following conditions exists:

- The number of bytes per second that undergo JIT compilation (JIT byte rate) is less than a specified threshold. The JIT byte rate is the difference between the last two samples divided by the number of seconds in the interval.
- The percent of time spent in JIT compilation since the application started exceeds a specified threshold.

NOTE: The monitored .NET applications must be running at the time of discovery so their resource objects can discovered. The applications must also be running for this script to collect data.

Default Schedule

The default schedule for this script is **Every 15 minutes**.

Resource Objects

Managed Applications folder

Managed Applications object

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	

Parameter	How to Set It
Raise event if application not running?	Set to **Yes** to raise an event if the managed application is not running. The default is Yes.
Event severity when application not running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the managed application is not running. The default is 20 (yellow event indicator).
Monitor JIT Byte Rate	
Event Notification	
Raise event if JIT byte rate falls below threshold?	Set to **Yes** to raise an event if the JIT byte rate falls below the threshold you set. The default is Yes.
Threshold - Minimum JIT byte rate	Specify the minimum number of JIT bytes that must occur per second to prevent an event from being raised. The default is 512 bytes per second.
Event severity when JIT byte rate falls below threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the JIT byte rate falls below the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for JIT byte rate?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the JIT byte rate for the monitoring interval. The default is unselected.
Monitor Percent of Time in JIT	
Event Notification	
Raise event if percent of time spent in JIT compilation exceeds threshold?	Set to **Yes** to raise an event if the amount of time the application spends in JIT compilation exceeds the threshold you set. The default is Yes.
Threshold - Maximum percent of time spent in JIT compilation	Specify the maximum percentage of time the application can spend in JIT compilation before an event is raised. The default is 60%.
Event severity when time spent in JIT compilation exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the amount of time the application spends in JIT compilation exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for percent of time in JIT compilation?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the amount of time the application spends in JIT compilation during the monitoring interval. The default is unselected.

6.17 CLRMemProfile

Use this Knowledge Script to monitor total garbage collection in managed .NET applications.

Total garbage collection frees memory slots in sections of unused memory and is composed of partial garbage collection (where memory freeing processes can be interrupted) and full garbage collection (where memory freeing processes cannot be interrupted).

In partial garbage collection, only the most recently allocated objects (Gen 0 objects) are counted.

In full garbage collection, older objects (Gen 1 and up) are counted.

This script raises an event if the total garbage collection time since the previous collection exceeds a specified percentage.

NOTE: The monitored .NET applications must be running at the time of discovery so their resource objects can discovered. The applications must also be running for this script to collect data.

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Resource Objects

Managed Applications folder

Managed Applications object

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Raise event if application not running?	Set to **Yes** to raise an event if the managed application is not running. The default is Yes.
Event severity when application not running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the managed application is not running. The default is 20 (yellow event indicator).
Monitor Garbage Collection Time	
Event Notification	
Raise event if percentage of time spent in garbage collection exceeds threshold?	Set to **Yes** to raise an event if the percent of time spent in garbage collection exceeds the threshold you set. The default is Yes.
Threshold - Maximum percentage of time spent in garbage collection	Specify the maximum amount of time the application should spend in garbage collection before an event is raised. The default is 6%.
Event severity when garbage collection time exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the percent of time spent in garbage collection exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for memory profile?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the amount of time spent in garbage collection during the monitoring interval. The default unselected.

6.18 CLRNetworking

Use this Knowledge Script to monitor network activity in managed .NET applications.

This script raises an event if one of the following conditions exists:

- The number of bytes sent during a process, through all socket connections, exceeds a specified threshold. The number of bytes includes data as well as non-TCP/IP protocol information.
- The number of bytes received during a process, through all socket connections, exceeds a specified threshold. The number of bytes includes data as well as non-TCP/IP protocol information.

NOTE: The monitored .NET applications must be running at the time of discovery so their resource objects can discovered. The applications must also be running for this script to collect data.

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Resource Objects

Networking Managed Applications folder

Networking Managed Applications object

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Raise event if application not running?	Set to **Yes** to raise an event if the managed application is not running. The default is Yes.
Event severity when application not running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the managed application is not running. The default is 20 (yellow event indicator).
Monitor Network Traffic	
Event Notification	
Raise event if network bytes sent exceed threshold?	Set to **Yes** to raise an event if the number of bytes sent during a process exceeds the threshold you set. The default is Yes.
Threshold - Maximum number of network bytes sent	Specify the maximum number of bytes that can be sent during a process before an event is raised. The default is 1000000 bytes.
Event severity when network bytes sent exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of bytes sent during a process exceeds the threshold. The default is 15 (yellow event indicator).
Raise event if network bytes received exceed threshold?	Set to **Yes** to raise an event if the number of bytes received during a process exceeds a specified threshold. The default is Yes.

Parameter	How to Set It
Threshold - Maximum number of network bytes received	Specify the maximum number of bytes that can be received during a process before an event is raised. The default is 1000000 bytes.
Event severity when network bytes received exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of bytes received during a process exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for network byte sent?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the number of bytes sent during the monitoring interval. The default is unselected.
Collect data for network byte received?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the number of bytes received during the monitoring interval. The default is unselected.

6.19 CLRRemoting

Use this Knowledge Script to monitor remote procedure call (RPC) activity in managed .NET applications. An RPC is a type of protocol that enables a software program to execute on a remote server.

This script raises an event if the number of RPC calls per second (the RPC call rate) exceeds the threshold you set. The RPC call rate is the difference between the last two samples divided by the number of seconds in the interval.

NOTE: The monitored .NET applications must be running at the time of discovery so their resource objects can discovered. The applications must also be running for this script to collect data.

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Resource Objects

Managed Applications folder

Managed Applications object

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Raise event if application not running?	Set to **Yes** to raise an event if the managed application is not running. The default is Yes.

Parameter	How to Set It
Event severity when application not running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the managed application is not running. The default is 20 (yellow event indicator).
Monitor Remote Procedure Call Rate	
Event Notification	
Raise event if RPC rate exceeds threshold?	Set to **Yes** to raise an event if the number of RPCs per second exceeds the threshold you set. The default is Yes.
Threshold - Maximum RPC rate	Specify the maximum number of RPCs that can occur per second before an event is raised. The default is 32 calls per second.
Event severity when RPC rate exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of RPC calls per second exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for RPC rate?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the RPC rate for the monitoring interval. The default is unselected.

6.20 CLRThreads

Use this Knowledge Script to monitor thread use in managed .NET applications. This script raises an event if one of the following conditions exists:

- The total number of threads (total recognized threads) that have run at least once since the application started exceeds a specified threshold.
- The rate at which processors are assigned to alternate threads (context switch rate) exceeds a specified threshold. For example, the kernel can reassign an operation when a thread with higher priority becomes available.

NOTE: The monitored .NET applications must be running at the time of discovery so their resource objects can discovered. The applications must also be running for this script to collect data.

Default Schedule

The default schedule for this script is **Every 5 minutes**.

Resource Objects

Managed Applications folder

Managed Applications object

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Raise event if application not running?	Set to **Yes** to raise an event if the managed application is not running. The default is Yes.
Event severity when application not running	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the managed application is not running. The default is 20 (yellow event indicator).
Monitor Thread Count	
Event Notification	
Raise event if thread count exceeds threshold?	Set to **Yes** to raise an event if the number threads that have run at least once exceeds the threshold you set. The default is Yes.
Threshold - Maximum thread count	Specify the maximum number of threads that can run before an event is raised. The default is 32 threads.
Event severity when thread count exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of threads exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for thread count?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the number of threads that ran at least once during the monitoring interval. The default is unselected.
Monitor Context Switch Rate	
Event Notification	
Raise event if context switch rate exceeds threshold?	Set to **Yes** to raise an event if the switch rate exceeds the threshold you set. The default is Yes.
Threshold - Maximum context switch rate	Specify the maximum rate at which processors can be assigned to alternate thread before an event is raised. The default is 4096 switches per second.
Event severity when context switch rate exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the switch rate exceeds the threshold. The default is 15 (yellow event indicator).
Data Collection	
Collect data for context switch rate?	Set to **Yes** to collect data for charts and reports. When enabled, data collection returns the context switch rate for the monitoring interval. The default is unselected.

6.21 DCOMAppChange

Use this Knowledge Script to monitor the distributed COM (DCOM) application list. This script raises an event if an application has been added or removed (registered or unregistered) from the application list on the target computer since the last time the script was run.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every 24 hours**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if DCOM applications added or removed?	Set to **y** to raise an event when one or more applications have been added to or removed from the DCOM application list. The default is y.
Collect data for DCOM applications added or removed?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of DCOM applications currently registered. The default is n.
Event severity when DCOM applications added or removed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which applications have been added to or removed from the DCOM application list. The default is 5 (red event indicator).

6.22 FaxActivity

Use this Knowledge Script to monitor the total number of faxes, fax pages, and fax bytes sent and received. This script raises events if the number of faxes, fax pages, or fax bytes sent and received exceeds the thresholds you set.

Resource Object

Fax folder

Default Schedule

The default interval for this script is **Every 15 minutes.**

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if number of bytes received exceeds threshold?	Set to **y** to raise an event when the number of received fax bytes exceeds the threshold you set. The default is y.

Parameter	How to Set It
Raise event if number of bytes sent exceeds threshold?	Set to **y** to raise an event when the number of sent fax bytes exceeds the threshold you set. The default is y.
Raise event if number of faxes received exceeds threshold?	Set to **y** to raise an event when the number of received faxes exceeds the threshold you set. The default is y.
Raise event if number of faxes sent exceeds threshold?	Set to **y** to raise an event when the number of sent faxes exceeds the threshold you set. The default is y.
Raise event if number of pages received exceeds threshold?	Set to **y** to raise an event when the number of received fax pages exceeds the threshold you set. The default is y.
Raise event if number of pages sent exceeds threshold?	Set to **y** to raise an event when the number of sent fax pages exceeds the threshold you set. The default is y.
Raise event if total number of bytes sent and received exceeds threshold?	Set to **y** to raise an event when the number of total number of sent and received fax bytes exceeds the threshold you set. The default is y.
Raise event if total number of faxes sent and received exceeds threshold?	Set to **y** to raise an event when the number of total number of sent and received faxes exceeds the threshold you set. The default is y.
Raise event if total number of pages sent and received exceeds threshold?	Set to **y** to raise an event when the number of total number of sent and received fax pages exceeds the threshold you set. The default is y.
Threshold - Maximum number of bytes received	Specify the maximum number of fax bytes that can be received before an event is raised. The default is 20000 bytes.
Threshold - Maximum number of bytes sent	Specify the maximum number of fax bytes that can be sent before an event is raised. The default is 20000 bytes.
Threshold - Maximum number of faxes received	Specify the maximum number of fax bytes that can be received before an event is raised. The default is 20000 bytes.
Threshold - Maximum number of faxes sent	Specify the maximum number of faxes that can be sent before an event is raised. The default is 20000 faxes.
Threshold - Maximum number of fax pages received	Specify the maximum number of fax pages that can be received before an event is raised. The default is 20000 pages.
Threshold - Maximum number of fax pages sent	Specify the maximum number of fax pages that can be sent before an event is raised. The default is 20000 pages.
Threshold - Maximum total fax bytes sent and received	Specify the maximum total number of fax bytes that can be sent and received before an event is raised. The default is 20000 bytes.
Threshold - Maximum total faxes sent and received	Specify the maximum total number of faxes that can be sent and received before an event is raised. The default is 20000 faxes.
Threshold - Maximum total fax pages sent and received	Specify the maximum total number of fax pages that can be sent and received before an event is raised. The default is 20000 pages.
Collect data for number of bytes received?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of fax bytes received during the monitoring interval. The default is n.

Parameter	How to Set It
Collect data for number of bytes sent?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of fax bytes sent during the monitoring interval. The default is n.
Collect data for number of faxes received?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of faxes received during the monitoring interval. The default is n.
Collect data for number of faxes sent?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of faxes sent during the monitoring interval. The default is n.
Collect data for number of pages received?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of fax pages received during the monitoring interval. The default is n.
Collect data for number of pages sent?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of fax pages sent during the monitoring interval. The default is n.
Collect data for total number of bytes?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the total number of fax bytes sent and received during the monitoring interval. The default is n.
Collect data for total number of faxes?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the total number of faxes sent and received during the monitoring interval. The default is n.
Collect data for total number of pages?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the total number of fax pages sent and received during the monitoring interval. The default is n.
Event severity when any threshold exceeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default is 8 (red event indicator).

6.23 FaxEventLog

Use this Knowledge Script to periodically scan the Windows Application event log for entries created by the Microsoft Fax service that match the criteria you specify. If any events are found, AppManager raises an event, and the event detail message provides more information about the event.

In the first interval, the value you specify for the *Start with events in past N hours* parameter determines how far back in the log to check for matching entries. As the job continues to run at subsequent intervals, it checks for any new entries created since the last time the log was checked.

You can further restrict the types of log entries that generate an event in two ways:

- Use the *Monitor for events of type [...]* parameters to search only certain types of events, such as Warning events.
- Use the *Filter the [...] field for* parameters to search only for specific information, such as events associated with a specific user or computer name.

Resource Object

Fax folder

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if Fax service event log entries found?	Set to **y** to raise an event if the event log contains entries that match your search criteria. The default is y.
Collect data for Fax service entries?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the number of log entries found, and the data point detail message returns the text of the log entries. The default is n.
Start with events in past N hours	Use this parameter to determine which part of the event log is searched the *first* time you run the job. Subsequent searches begin where the previous one finished. The following entries are valid: ◆ **-1** to search all existing log entries during the first interval ◆ **n** to search entries for the past *n* hours (8 for the past 8 hours, 50 for the past 50 hours, for example.) ◆ **0** to search no previous entries (search from the current time forward) The default is 0.
Monitor for events of type:	Set to **y** for each type of event you want to monitor: ◆ Error ◆ Warning ◆ Information ◆ Success Audit ◆ Failure Audit If you disable any of these event types, that type of log entry does not raise an event, is not returned in an event detail message, and is not collected as data if you enabled *Collect data for Fax service entries?* The default is y.

Parameter	How to Set It
Filter the [...] field for	To limit the types of entries that raise events and the type of data that is collected, enter a search string that filters the following fields in the event log: ◆ **Category**. Specify one or more text strings to look for in the Category field. Separate multiple strings with commas. ◆ **Event ID**. Specify single or multiple event IDs. Separate multiple entries with commas. To specify a range of event IDs, use a hyphen. For example: `414,1028-1400,4015`. ◆ **User**. Specify a single or multiple user names to look for. Separate multiple entries by commas. For example: `Pat,Chris,Alex`. ◆ **Computer**. Specify a single or multiple computer names or IP addresses to look for. Separate multiple entries by commas. For example: `SHASTA,MARS`. ◆ **Event Description**. Specify a description or keywords in the description. The string can contain spaces, underscores, and periods. Separate multiple entries with commas. For example: `no domain,critical error from the Active Directory`. The search string can contain criteria used to include entries, exclude entries, or both. ◆ Separate the include and exclude criteria with a colon (:). For example, `zones,caching:primary or secondary`. ◆ Separate multiple include or exclude entries with commas. For example, `finance,sales:corp00,HQ`. ◆ If you are specifying only include criteria, the colon is not necessary. For example, `primary DNS domain`. ◆ If you are specifying only exclude criteria, start the search string with a colon. For example, `:online help`.
Maximum number of entries per event	Specify the maximum number of log entries to be included in each event's detail message. If this script finds more entries in the log than the specified maximum, the script will return multiple events to report the number of entries you have specified. The default is 30 entries.
Event severity when matching Fax service log entries found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the event log contains entries that match your search criteria. The default is 8 (red event indicator).

6.24 FaxServiceDown

Use this Knowledge Script to monitor the status of the Microsoft Fax service. This script raises an event if the service is down. You can set this script to automatically attempt to restart the service when it is not running.

Resource Object

Fax service object

Default Schedule

The default interval for this script is **Every 30 minutes.**

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if Fax service is down?	Set to **y** to raise an event when the Microsoft fax service is down. The default is y.
Collect data for Fax service status?	Set to **y** to collect data for charts and reports. If enabled, data collection returns a value of 100 if the service is running, and a value of 0 if the service is not running. The default is n.
Automatically restart service if down?	Set to **y** to automatically restart the service when it is down. The default is y.
Event severity when auto-start fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and AppManager cannot restart it. The default is 5 (red event indicator).
Event severity when auto-start succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in the service was down and AppManager successfully restarted it. The default is 25 (blue event indicator).
Event severity when service down and auto-start disabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and AppManager has been set to not restart the service. The default is 18 (yellow event indicator).

6.25 FaxTotalFailed

Use this Knowledge Script to monitor the total number of failed faxes, failed outgoing connections, and failed receptions. This script raises an event if the number of failed faxes, failed outgoing connections, or failed receptions exceeds the threshold you set.

Resource Object

Fax folder

Default Schedule

The default interval for this script is **Every 15 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if number of failed faxes exceeds threshold?	Set to **y** to raise an event if the number of failed faxes exceeds the threshold you set. The default is y.
Raise event if number of failed receptions exceeds threshold?	Set to **y** to raise an event if the number of failed fax receptions exceeds the threshold you set. The default is y.

Parameter	How to Set It
Raise event if number of failed outgoing connections exceeds threshold?	Set to **y** to raise an event if the number of failed outgoing fax connections exceeds the threshold you set. The default is y.
Threshold - Maximum number of failed faxes	Specify the maximum number of faxes that can fail to connect or be received before an event is raised. The default is 200 faxes.
Threshold - Maximum number of failed receptions	Specify the maximum number of faxes that can fail to be received before an event is raised. The default is 20 faxes.
Threshold - Maximum number of failed outgoing connections	Specify the maximum number of faxes that can fail to make an outgoing connection before an event is raised. The default is 20 faxes.
Collect data for number of failed faxes?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the number of faxes that failed to connect or be received during the monitoring interval. The default is n.
Collect data for number of failed receptions?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the number of faxes that failed to be received during the monitoring interval. The default is n.
Collect data for number of failed outgoing connections?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the number of faxes that failed to make an outgoing connection during the monitoring interval. The default is n.
Event severity when threshold exceeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default is 8 (red event indicator).

6.26 FaxTotalTime

Use this Knowledge Script to monitor the number of minutes the Microsoft Fax service spends receiving faxes, the number of minutes the service spends sending faxes, and the total number of minutes the service spends receiving and sending faxes. This script raises an event if any of these values exceeds the threshold you set.

Resource Object

Fax folder

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if minutes spent receiving faxes exceeds threshold?	Set to **y** to raise an event when the number of minutes the fax service spent receiving faxes exceeds the threshold you set. The default is y.

Parameter	How to Set It
Raise event if minutes spent sending faxes exceeds threshold?	Set to **y** to raise an event when the number of minutes the fax service spent sending faxes exceeds the threshold you set. The default is y.
Raise event if total minutes spent sending and receiving faxes exceeds threshold?	Set to **y** to raise an event when the total number of minutes the fax service spent sending and receiving faxes exceeds the threshold you set. The default is y.
Threshold - Maximum minutes spent receiving faxes	Specify the maximum number of minutes that the fax service can spend receiving faxes before an event is raised. The default is 20000 minutes.
Threshold - Maximum minutes spent sending faxes	Specify the maximum number of minutes that the fax service can spend sending faxes before an event is raised. The default is 20000 minutes.
Threshold - Maximum total minutes spent receiving and sending faxes	Specify the maximum number of minutes that the fax service can spend sending and receiving faxes before an event is raised. The default is 20000 minutes.
Collect data for minutes spent receiving faxes?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of minutes the fax service spent receiving faxes during the monitoring period. The default is n.
Collect data for minutes spent sending faxes?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of minutes the fax service spent sending faxes during the monitoring period. The default is n.
Collect data for total minutes spent receiving and sending faxes?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of minutes the fax service spent sending and receiving faxes during the monitoring period. The default is n.
Event severity when any threshold exceeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default is 8 (red event indicator).

6.27 OpenSystemSlots

Use this Knowledge Script to monitor the number of available system (PCI) slots. This script raises an event if the number of available system slots falls below the threshold you set.

PCI (Peripheral Component Interconnect) slots allow different types of expansion cards to be connected inside a computer to extend the computer’s functionality. Examples of PCI expansion cards are network cards, graphics cards, and sound cards.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every 24 hours**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if available system slots fall below threshold?	Set to **y** to raise an event when the number of available system slots falls below the threshold you set. The default is y.
Collect data for number of available system slots?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of system slots that were available during the monitoring period. The default is n.
Threshold - Minimum number of available system slots	Specify minimum number of system slots that must be available to prevent an event from being raised. The default is 1 slot.
Event severity when available system slots fall below threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of available system slots falls below the threshold. The default is 5 (red event indicator).

6.28 PNPDeviceChange

Use this Knowledge Script to monitor the plug-and-play device list for any device that has been added or removed since the script was last run. This script raises an event if plug-and-play devices (for example, an external modem) are added or removed.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every 24 hours**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if plug and play devices added or removed?	Set to **y** to raise an event when a plug-and-play device is added to or removed from the device list. The default is y.
Collect data for devices added or removed?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the contents of the device list. The default is n.
Event severity when devices added or removed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a plug-and-play device is added or removed from the device list. The default is 5 (red event indicator).

6.29 PNPDeviceErrors

Use this Knowledge Script to monitor the number of plug-and-play devices that have a status of “error.” This script raises an event if the number of plug-and-play devices with error status exceeds the threshold.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every 24 hours**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to set it
Raise event if plug and play devices with error status exceed threshold?	Set to **y** to raise an event when the number of devices with a status of “error” exceeds the threshold you set. The default is y.
Collect data for number of devices with error status?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of devices that have the status of “error.” The default is n.
Threshold - Maximum number of plug and play devices with error status	Specify the maximum number of devices that can have an “error” status before an event is raised. The default is 10 devices.
Event severity when devices with error status exceed threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of devices with “error” status exceeds the threshold. The default is 5 (red event indicator).

6.30 PrinterStuckJobs

Use this Knowledge Script to monitor jobs that are stuck in the printer queue. This script raises an event if the number of minutes a job has remained in the printer queue exceeds the threshold you set.

Resource Objects

Printer folder

Printer object

Default Schedule

The default interval for this script is **Every hour**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if time in printer queue exceeds threshold?	Set to **y** to raise an event when the number of minutes a job spends in the printer queue exceeds the threshold you set. The default is y.
Collect data for time spent in printer queue?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of minutes a job spent in the printer queue. The default is n.
Threshold - Maximum time spent in printer queue	Specify the maximum number of minutes a job can spend in the printer queue before an event is raised. The default is 5 minutes.
Event severity when time in printer queue exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of minutes a job spends in the printer queue exceeds the threshold. The default is 1 (red event indicator).

6.31 SRDiskPercent

Use this Knowledge Script to monitor the percentage of space on a disk available for the System Restore service. The System Restore service configuration for the percentage of space available (by default this value is 12%) applies to all the computer’s drives. If you enter a lower value for the threshold, this script raises an event, because the percentage of disk space available for the System Restore Service, as configured, exceeds the threshold you set.

If the amount of free disk space on a system drive falls below 200 MB, or if the amount of free disk space on a non-system drive falls below 80 MB, it is advisable to configure System Restore to turn off automatically on that drive. This script also raises an event if the space available for System Restore on any system or non-system drive falls below the threshold you set.

Resource Object

System Restore folder

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if percentage of disk space exceeds threshold?	Set to **y** to raise an event when the percentage of disk space configured for System Restore exceeds the threshold. The event detail message will include the disk space available for System Restore for each individual drive. The default is y.
Raise event if space available on a drive falls below threshold?	Set to **y** to raise an event when the space available for System Restore on any system or non-system drive falls below the threshold. The default is n.

Parameter	How to Set It
Collect data for percentage of disk space being used?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the percentage of disk space that is configured for System Restore. The default is n.
Collect data for space available on individual drive	Set to **y** to collect data for charts and reports. If enabled, data collection returns the space available for System Restore on any system or non-system drive. The default is n.
Threshold - Maximum disk space configured for System Restore	Specify the maximum percentage of disk space that can be configured for System Restore before an event is raised. The default is 12%.
Threshold - Minimum disk space available on system drive	Specify the minimum amount of disk space (in MB) that should be available for System Restore on a system drive to prevent an event from being raised. The default is 200 MB.
Threshold - Minimum disk space available on non-system drive	Specify the minimum amount of disk space (in MB) that should be available for System Restore on a non-system drive to prevent an event from being raised. The default is 80 MB.
Event severity when disk space available cannot be retrieved	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the amount of available disk space cannot be determined. The default is 8.
Event severity when disk space configured for System Restore exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the amount of configured disk space exceeds the threshold. The default is 8.
Event severity when space available on individual drive falls below threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the amount of available disk space for an individual drive falls below the threshold. The default is 5.

6.32 SREventLog

Use this Knowledge Script to periodically scan the Windows Application event log for entries created by the System Restore service that match the criteria you specify. This script raises an event if an entry matches criteria you specify. The event detail message provides more information about the event.

In the first interval, the value you specify for the *Start with events in past N hours* parameter determines how far back in the log to check for matching entries. As the script continues to run at subsequent intervals, it checks for any new entries created since the last time the log was checked.

You can further restrict the types of log entries that generate an event in two ways:

- Use the *Monitor for events of type [...]*

  parameters to search only certain types of events, such as Warning events.
- Use the *Filter the [...] field for* parameters to search only for specific information, such as events associated with a specific user or computer name.

Each time this script runs, it checks the Windows Application event log for entries matching your selection criteria and raises an event if matching entries are found. The event detail message returns the text of the log entries found. When this script is set to collect data, it returns the number of log entries found, and the data point detail message returns the text of the log entries.

Resource Object

System Restore folder

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if matching log entries found?	Set to **y** to raise an event when the log contains entries that match your search criteria. The default is y.
Collect data for matching log entries found?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of log entries found. The data point detail message returns the text of the log entries. The default is n.
Start with events in past N hours	Set this parameter to determine which part of the log to search the first time the job runs. Subsequent searches begin where the previous one finished. The following entries are valid: ◆ **-1** to search all existing log entries during the first interval ◆ **n** to search entries for the past *n* hours (8 for the past 8 hours, 50 for the past 50 hours, for example.) ◆ **0** to search no previous entries (search from the current time forward) The default is 0.
Monitor for events of type:	Set to **y** for each type of event you want to monitor: ◆ Error ◆ Warning ◆ Information ◆ Success Audit ◆ Failure Audit If you disable any of these event types, that type of log entry does not raise an event, is not returned in an event detail message, and is not collected as data if you enabled *Collect data for matching log entries found?* The default is y.

Parameter	How to Set It
Filter the [...] field for	To limit the types of entries that raise events and the type of data that is collected, enter a search string that filters the following fields in the event log: ◆ **Category**. Specify one or more text strings to look for in the Category field. Separate multiple strings with commas. ◆ **Event ID**. Specify single or multiple event IDs. Separate multiple entries with commas. To specify a range of event IDs, use a hyphen. For example: `414,1028-1400,4015`. ◆ **User**. Specify a single or multiple user names to look for. Separate multiple entries by commas. For example: `Pat,Chris,Alex`. ◆ **Computer**. Specify a single or multiple computer names or IP addresses to look for. Separate multiple entries by commas. For example: `SHASTA,MARS`. ◆ **Event Description**. Specify a detail description or keywords in the description. The string can contain spaces, underscores, and periods. Separate multiple entries with commas. For example: `no domain,critical error from the Active Directory`. The search string can contain criteria used to include entries, exclude entries, or both. ◆ Separate the include and exclude criteria with a colon (:). For example, `zones,caching:primary or secondary`. ◆ Separate multiple include or exclude entries with commas. For example, `finance,sales:corp00,HQ`. ◆ If you are specifying only include criteria, the colon is not necessary. For example, `primary DNS domain`. ◆ If you are specifying only exclude criteria, start the search string with a colon. For example, `:online help`.
Maximum number of entries per event message	Specify the maximum number of log entries to be included in each event's detail message. If this script finds more entries in the log than the specified maximum, it will return multiple events to report the number of entries you have specified. The default is 30 entries.
Event severity when matching entries found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the log contains entries that match your search criteria. The default is 8 (red event indicator).

6.33 SRLifeInterval

Use this Knowledge Script to monitor the number of days the System Restore service preserves System Restore points.

A *restore point* is a snapshot of the system provided by the System Restore service. For example, when you install an application on your computer, a restore point is created and stored in the database. If you later want to return to the registry as it was configured before you installed the new application, select the restore point in the System Restore utility that represents your system configuration prior to installation. Your system is restored to its state before the new application was installed.

This script raises an event if the number of days that restore points have been preserved exceeds the threshold.

Resource Object

System Restore folder

Default Schedule

The default interval for this script is **Every 24 hours**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if restore point lifetime exceeds threshold?	Set to **y** to raise an event when the number of days a restore point has been preserved exceeds the threshold you set. The default is y.
Collect data for length of restore point lifetime?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of days a restore point has been preserved. The default is n.
Maximum number of restore points to include in event	Specify the number of restore points to display in the event detail message. A full list of restore points can be viewed at `$(INSTALLPATH)\LifeSrPtsLog`. Restore points are displayed in order of creation date, the most recent first. The default is 20 restore points. Set to 0 to display all restore points in the event detail message.
Threshold - Maximum restore point lifetime	Specify the maximum number of days restore points can be preserved before an event is raised. The default is 4 days.
Event severity when restore point lifetime exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of days restore points are preserved exceeds the threshold. The default is 8 (red event indicator).

6.34 SRPoints

Use this Knowledge Script to monitor the number of System Restore points that are being preserved by the System Restore service. This script raises an event if the number of System Restore points exceeds the threshold. If the System Restore system preserves too many restore points, including many old restore points, the life interval of the restore points may be too long.

Resource Object

System Restore folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if number of restore points exceeds threshold?	Set to **y** to raise an event when the number of restore points exceeds the threshold. The default is y.
Collect data for number of restore points?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of restore points being preserved by the System Restore service. The default is n.
Threshold - Maximum total system restore points	Specify the maximum number of restore points that can be preserved before an event is raised. The default is 4 restore points.
Maximum number of restore points to include in event	Specify the number of restore points to display in the event detail message. A full list of restore points can be viewed at `$(INSTALLPATH)\ShortSrPtslog`. Restore points are displayed in order of creation date, the most recent first. The default is 20 restore points. Set to 0 to display all restore points in the event detail message.
Event severity when total number of restore points exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of preserved restore points exceeds the threshold. The default is 8 (red event indicator).

6.35 SRScheduledInterval

Use this Knowledge Script to monitor the interval, in hours, at which scheduled System Restore points are created during both current and global sessions. This script raises an event if the interval exceeds the threshold.

Exceeding the threshold means that System Restore points are being created less often than the threshold, not more often. For example, there may be five hours between System Restore points, rather than four (the default maximum threshold).

Resource Object

System Restore folder

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if global time interval exceeds threshold?	Set to **y** to raise an event when the interval at which restore points are created exceeds the time interval for global sessions. The default is y.
Raise event if current session time interval exceeds threshold?	Set to **y** to raise an event when the interval at which restore points are created exceeds the time interval for current sessions. The default is y.
Collect data for global time interval?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the restore point creation interval for global sessions. The default is n.
Collect data for current session time interval?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the restore point creation interval for current sessions. The default is n.
Threshold - Maximum global time interval	Specify the maximum interval at which restore points can be created for a global session before an event is raised. The default is every 4 hours.
Threshold - Maximum current session time interval	Specify the maximum interval at which restore points can be created for a current session before an event is raised. The default is every 4 hours.
Maximum number of restore points to include in event	Specify the number of restore points to display in the detail message.These restore points will have the description "System Check Point." A full list of restore points can be viewed at `$(INSTALLPATH)]\SysChkLog`. Restore points are displayed in order of creation date, the most recent first. The default is 20 restore points. Enter 0 to display all restore points in the event detail message.
Event severity when global time interval exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the restore point creation interval for global sessions exceeds the threshold. The default is 8 (red event indicator).
Event severity when current session time interval exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the restore point creation interval for current sessions exceeds the threshold. The default is 8 (red event indicator).

6.36 SRServiceDown

Use this Knowledge Script to monitor the status of the System Restore service. This script raises an event if the service is down. You can set this script to automatically attempt to restart the service when it is not running.

Resource Object

System Restore Service object

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if System Restore service is down?	Set to **y** to raise an event when the System Restore service is down. The default is y.
Collect data for System Restore service status?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ 100 -- service is running, or ◆ 0 -- service is not running. The default is n.
Auto-start service if down?	Set to **y** to automatically restart the service when it is down. The default is y.
Event severity when auto-start fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and AppManager cannot restart it. The default is 5 (red event indicator).
Event severity when auto-start succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service was down and AppManager successfully restarted it. The default is 25 (blue event indicator).
Event severity when service down and auto-start disabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service is down and AppManager has been set to not restart the service. The default is 18 (yellow event indicator).

7 PowerShell Knowledge Scripts

AppManager for Microsoft Windows provides the following Knowledge Scripts for monitoring the PowerShell scripting and command environment. PowerShell is made up of hundreds of executable objects called *cmdlets*, pronounced command-lets.

From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. Or in the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
RunCommand	Runs a specified PowerShell command.

7.1 RunCommand

Use this Knowledge Script to run the Microsoft Windows PowerShell cmdlet, PowerShell script (`.PS1` file), or code blocks you specify. This script raises an event with the command results and generates a datastream with the value returned by the command.

You can also use this script to run any command that can be run from a Windows PowerShell command prompt, such as `dir c:\temp`. PowerShell accepts commands in cmdlet, `.PS1`, and Windows `cmd.exe` formats.

The PowerShell_RunCommand script makes a number of callback and helper functions available to the PowerShell commands or scripts being run. For more information, see Appendix A, “Using PowerShell Callback and Helper Functions” in the management guide.

Prerequisites

- Microsoft Windows PowerShell version 1.0 or later
- Microsoft .NET Framework version 3.0
- AppManager for Windows version 7.6 or later

Resource Object

PowerShell folder

Default Schedule

By default, this script runs once.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Job Failure Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the RunCommand job fails. The default is 5.
PowerShell Command	

PowerShell to run	Provide the PowerShell scripts, cmdlets, or code blocks you want to run. You can string multiple commands together. For example:
	`Select-String -Path C:\Temp\*.log -Pattern 'Error:.*CPU' | foreach {$_.ToString()}`
	This command returns all lines in all log files in the `C:\Temp` directory that contain the text strings *Error:* followed by *CPU*, with any number of characters (zero or more) between the two strings.
	Ensure your command contains no syntax errors. The RunCommand job will fail if the command contains syntax errors.
	Note Double quotation marks within a command are automatically doubled up, unless the only double-quotes in the command are already doubled up because they represent empty strings.In this situation, you can work around this issue by adding a final statement to the command. For example:
	`$foo -eq ""; [void] "x"`
	where the `;` separates this final statement from the rest of the statements, and the `[void] "x"` has no actual effect on the command execution, but it enables the script to recognize that all double-quotes in the command need to be doubled up.
	Restrictions
	◆ You can run scripts that have pathnames with spaces, but you need to use the *call operator* (&), such as: `& 'C:\Program Files\My Files\Agent.ps1'` The quotes (either single or double-quotes) around the full pathname are required if the path contains spaces.
	◆ PowerShell scripts (`.PS1` files) must be located on the computer on which you run the RunCommand script. The RunCommand script cannot run remote PowerShell scripts.

Event Notification	
Raise event with result of command?	Select **Yes** to raise an event when the command you run returns text or numeric results. The default is Yes.
Format results as	Select whether to format command results in a **Table** or a **List** or to apply no formatting. Select **Unformatted** if the command returns results that are already formatted. The default is Unformatted.
Event title	Provide text to use as the title of the event.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a command returns text or numeric results. The default is 25.

Parameter	How to Set It
Raise event only if result contains specified pattern?	Select **Yes** to raise an event if the command returns text that matches the expression you provide in the *Pattern to find in the results* parameter. The default is unselected. This parameter is valid only if the *Raise event with result of command* parameter is enabled.
Pattern to find in the results	Provide the text you want to compare to the command results. The following wildcards are acceptable: ◆ * - matches zero or more instances of a character. ◆ **?** - matches exactly one instance of a character. ◆ **[ ]** - matches exactly one instance of any character between the square brackets, including ranges. Examples: ◆ `[abc][def]` matches “ad,” “bad,” and “ace,” but not “bleary.” ◆ `[a-z][a-z]` matches text that contains two adjacent alphabetic characters. ◆ `foo? Bar` matches “food bar” and “This is a food bar!” but not “foobar” or “foo bar.” ◆ `*maximum mailbox*` matches “user smith has reached maximum mailbox size.”
Raise event only if numeric result crosses threshold?	Select **Yes** to raise an event if the command returns a numeric value that exceeds or falls below the threshold you set in the *Threshold value parameter*. The default is unselected.
Operator to compare numeric result to threshold	Select the operator with which to compare the command results to the threshold value. Choose from one of the following: ◆ Greater than ◆ Less than ◆ Greater than or equal to ◆ Less than or equal to An event is raised if the command results do not match the threshold value based on the operator you choose. The default is Greater than.
Threshold value	Provide the numeric value to compare with the command results. An event is raised if the command results do not match the threshold value based on the option you choose in the *Select operator to compare numeric result to threshold* parameter. The default is 0.
Metric name to include in event title	Provide the name of the metric for which the command returns numeric results. For example, specify the name of a Performance Monitor counter. The name of the metric will be part of the title of the event raised when the numeric result crosses the threshold.

Parameter	How to Set It
Raise event if command returns no results?	Select **Yes** if the command you run returns no results. The default is unselected. **Hint** You can use this parameter to raise an event when a command that *should* return text or numeric results does not return any results. Enable this parameter and disable the *Raise event with result of command?* parameter.
Event severity when command returns no results	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a command does not return text or numeric results. The default is 5.
Data Collection	
Collect data for numeric command result?	Select **Yes** to collect data for charts and reports. This parameter is valid only if the *Raise event only if numeric result crosses threshold?* parameter is enabled. The default is unselected. Use the following parameters to format the wording and units of datastream legends.
Name of monitored metric	Provide the name of the metric for which the command returns numeric results. For example, specify the name of a Performance Monitor counter. The name of the metric will be part of the datastream legend.
Name of monitored resource	Provide the name of the device associated with the metric for which the command returns numeric results. For example, specify the hostname of a computer. The name of the device will be part of the datastream legend.
Datastream units	Identify the unit of measure associated with the metric for which the command returns numeric results. For example, specify `MB` or `Mbytes`. The unit of measure will be part of the datastream legend.

8 General Knowledge Scripts

The General category provides Knowledge Scripts for generalized monitoring tasks that can be applied to almost any application.

From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. In the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
ADAuthentication	Monitors login time and the time required to read a property value of an object on an Active Directory server.
AsciiLog	Monitors ASCII text files for specific strings and messages logged since the last monitoring interval.
AsciiLogRX	Monitors an ASCII text file for specific strings and messages, as defined by regular expressions, logged since the last monitoring interval.
ConfigMachineDown	Loads computer-specific parameters to a local monitored computer so the MachineDownLR script, running on a group, can get the parameters required for each computer where it runs.
Counter	Monitors any System Monitor performance counter.
CounterCorrelate	Monitors thresholds for any pair of System Monitor performance counters.
EventLog	Monitors and filters information in the Windows event logs based on criteria you define.
EventLogRX	Monitors the Windows event logs for new entries matching the filter criteria you define using regular expressions.
MachineDown	Checks whether the target machine you run the script on can communicate with one or more specified Windows computers.
MachineDownLR	Using parameters planted locally by the ConfigMachineDown Knowledge Script, runs on a group of computers to check whether each computer can communicate with one or more specified Windows computers.
MissingEvent	Determines whether a Windows event log does not contain an entry matching your search criteria.
PingMachine	Checks the availability of any computers that reply to ICMP Echo requests (ping command).
Report_MachineAvailability	Generates a report about the availability of computers.
Report_PingMachine	Generates a report about the availability of computers or other machines that reply to ICMP Echo requests.
Report_ServiceChange	Generates a report about changes to the status and start-type of discovered services.
Report_ServiceDown	Generates a report about the up/down status of discovered services.

Knowledge Script	What It Does
Report_ServiceHung	Generates a report about discovered services in the Start-Pending, Stop-Pending, Continue-Pending, or Pause-Pending state.
ServiceChange	Detects any changes to the status and start type of a discovered service.
ServiceDown	Determines whether a discovered service is running.
ServiceHung	Determines whether a discovered service is hung.
ShortEventLog	Monitors and filters information in the Windows event logs based on criteria you specify.
SNMPGet	Monitors SNMP activity and allows you to check SNMP MIB variable values.
WMICounter	Monitors any WMI object property.

8.1 Creating Filters with Regular Expressions for General_AsciiLogRX

Some Knowledge Scripts enable you to use regular expressions to define include and exclude filters for pattern-matching against the text being evaluated. Depending on the Knowledge Script you are working with, you may be able to use regular expression include and exclude filters when you are setting job properties or you may be able to maintain your search criteria independent of the Knowledge Script parameters in a separate filter file. You may also be able to use regular expression modifiers to further refine your filtering.

For example, if your **include filter** is `replic.*` and you specify the modifier `i` to make the search case-insensitive, the regular expression contains the wildcard (`.`) and repeat (`*`) special characters, indicating you want to find strings that start with `replic` followed by any string of characters. Messages containing either `replication` or `replicated` are captured.

The format is the same for the exclude filter. For example, to find log entries that do not start with the string `success`, the exclude filter might look like this:

```
^success.*
```

If you are only searching for included strings, you can leave the exclude filter blank. If you want to retrieve all messages in the log in a given interval, you can specify `.*` for the include filter and leave the exclude filter blank.

Using Special Characters

The following special characters can be used in regular expressions:

Use This Character	For This Purpose
.	Wildcard for any one character
*	Repeat zero or more occurrences
^	Beginning of the line
\$	End of the line
\	Escape the next meta-character

Use This Character	For This Purpose
|	Alternate matches
[ ]	Any character in the class set. You can specify individual characters or ranges.
( )	Grouping characters. For example, you can specify (a|b|c) to indicate a match with a, or b, or c.
+	Quantifier indicating one or more occurrences
?	Quantifier indicating zero or one occurrence
{*n*}	Quantifier indicating exactly *n* occurrence
\w	A word character (alphanumeric plus _)
\s	A white-space character
\d	A digit character

Using Regular Expression Modifiers

In addition to the special characters you can use in creating the regular expression, there are a number of modifiers that can be used to modify how pattern-matching is handled. Valid modifiers include:

Modifier	Description
c	Complements the search list
g	Matches globally as many times as possible
i	Makes the search case-insensitive
m	Treats the string as multiple lines
o	Interpolates variables only once
s	Treats the regular expression string as a single long line
x	Allows for regular expression extensions

For additional information about writing regular expressions, see your Perl documentation or other regular expression resources.

8.2 Creating Filters with Regular Expressions for General_EventLogRx

Some Knowledge Scripts enable you to use regular expressions to define include and exclude filters for pattern-matching against the text being evaluated. Depending on the Knowledge Script you are working with, you may be able to use regular expression include and exclude filters when you are setting job properties or you may be able to maintain your search criteria independent of the Knowledge Script parameters in a separate filter file. You may also be able to use regular expression modifiers to further refine your filtering.

For example, if your **include filter** is `replic.*` and you specify the modifier `i` to make the search case-insensitive, the regular expression contains the wildcard (`.`) and repeat (`*`) special characters, indicating you want to find strings that start with `replic` followed by any string of characters. Messages containing either `replication` or `replicated` are captured.

The format is the same for the exclude filter. For example, to find log entries that do not start with the string `success`, the exclude filter might look like this:

```
^success.*
```

If you are only searching for included strings, you can leave the exclude filter blank. If you want to retrieve all messages in the log in a given interval, you can specify `.*` for the include filter and leave the exclude filter blank.

Using Special Characters

The following special characters can be used in regular expressions:

Use This Character	For This Purpose
.	Wildcard for any one character, except the newline character `\n`
*	Repeat zero or more occurrences
^	Beginning of the string
\$	End of the string
\	Escape the next meta-character
|	Alternate matches
[ ]	Any character in the class set. You can specify individual characters or ranges.
( )	Grouping characters. For example, you can specify (a|b|c) to indicate a match with a, or b, or c.
+	Quantifier indicating one or more occurrences
?	Quantifier indicating zero or one occurrence
{*n*}	Quantifier indicating exactly *n* occurrence
\w	A word character (alphanumeric plus _)
\s	A white-space character
\d	A digit character

Using Regular Expression Modifiers

In addition to the special characters you can use in creating the regular expression, there are a number of modifiers that can be used to modify how pattern matching is handled. Valid modifiers include:

Modifier	Description
g	Matches globally as many times as possible
i	Makes the search case-insensitive
m	Specifies mulitline mode
s	Treats the regular expression string as a single long line
x	Allows for regular expression extensions

For additional information about writing regular expressions for EventLogRX, see regular expression options in Microsoft's .Net Regular Expression Character Classes at https://msdn.microsoft.com/en-us/library/20bw873z(v=vs.85).aspx.

8.3 ADAuthentication

Use this Knowledge Script to monitor how long it takes AppManager to log in to an Active Directory domain. You can also use this script to monitor how long it takes (response time) to read a property value of an object on the Domain Controller. This script raises an event if the login time or response read time exceeds the threshold you specify.

You can specify the Domain Controller to which you want to log in. If you do not specify a Domain Controller, then the script uses the nearest one. You must specify the account name and password used to connect to the Domain Controller.

To monitor response time for read operations, specify the LDAP path and the property name of an Active Directory object.

Resource Object

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Once every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if login or read response time exceeds threshold?	Select **Yes** to raise an event if the authentication time or response time exceeds the threshold you specify. The default is Yes.

Parameter	How to Set It
Collect data for login or read response time?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the authentication time (in ms) and the response time (in ms). The default is unselected.
Authenticate against domain controller	Specify the name of the Domain Controller for which you want to authenticate the login. If you do not specify a name, the script uses the nearest Domain Controller. The default is `server.netiq.com`.
User name	Specify the domain and user name for the account you are using to log in. Use the following format for this parameter: *<domain>**<username>*
Account password	Specify the password for the account you are using to log in. The password is stored in an encrypted format. **NOTE:** Maximum allowed password length is 32 characters.
Threshold - Maximum login time	Specify the maximum amount of time it can take to log in to the Domain Controller before an event is raised. The default is 1000 ms.
Monitor read-response time?	Select **Yes** to monitor the time (in ms) required to read the property value of an Active Directory object from a client. The default is unselected.
LDAP path to an object on the target AD server	Specify the LDAP path to the Active Directory object for which you want to measure response time. The default is `LDAP://server.netiq.com/RootDSE`.
Specify a property of the AD object	Specify a property of the Active Directory object for which you want to measure response time. The default is `serverName`.
Threshold - Maximum read time	Specify the maximum amount of time it can take to read the specified property before an event is raised. The default is 1000 ms.
Event severity when login or response time exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default is 8 (red event indicator).
Event severity level when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the ADAuthentication job fails. The default is 35 (magenta event indicator).

8.4 AsciiLog

Use this Knowledge Script to monitor one or more ASCII text files for specific strings and messages logged since the last monitoring interval. Also, use this Knowledge Script to specify a pattern or search string to look for in specified ASCII files, and report the matching entries found in the monitoring period. The script checks for changes to the text files that match the string you enter; it does not re-scan the entire file at each interval. The script gathers up to 2 MB worth of result matches for each iteration of the job.

In the first interval, the script reads the file and inserts a marker at the end of the file. The script does *not* search for a specified search string during the first interval. In subsequent intervals, the script checks the file for changes that match the search string you specified. The script raises an event if the number of lines matching your search criteria exceeds the threshold you set.

NOTE: The script reports the number of matched lines in each iteration and the detail message contains the text data. If the detail message is larger than 32KB, the data is saved in a file on the managed computer (for example, `C:\program files\netiq\appmanager\bin\log`) and the detail message contains the truncated data. If you generate these log files, periodically remove the files when you are done with them. This script supports files up to 12 GB in size.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Once every hour**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if matches are found?	Select **Yes** to raise events if text strings or messages that match your search criteria are found. The default is Yes.
Event severity when matches are found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which matches to your search criteria are found. The default is 15 (yellow event indicator).
Raise event if no files are found?	Select **Yes** to raise an event if no ASCII files matching your search criteria are found. The default is unselected.
Event severity when no files found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which no ASCII files matching your search criteria are found. The default is 10 (red event indicator).
Raise event if no matches are found?	Select **Yes** to raise an event if no text strings or messages that match your search criteria are found in the specified files. The default is unselected.
Event severity when no matches found	Set the event severity level, from 1 to 40, to indicate the importance of an event if no matches to your search criteria are found in the specified files. The default is 20 (yellow event indicator).
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the AsciiLog job fails. The default is 5 (red event indicator).
Data Collection	

Parameter	How to Set It
Collect data for matches to search criteria?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns one or more datastreams for each of your search criteria. For example, if you search for `logon` and `logoff`, and `logon` is found in `C:\Log01` and `C:\Log02,` but `logoff` is not found, the script will return three datastreams: ◆ Instances of logon in `C:\Log01` ◆ Instances of logon in `C:\Log02` ◆ Instances of logoff Each data point in a datastream contains the number of matches found for that iteration of the script. The default is unselected.
Monitoring	
Directory to monitor	Specify the path to the directory in which you want to begin your search, or click **Browse [...]** to navigate to that directory. UNC paths are also supported, such as `\\ENG\appdev`.
Include sub-directories?	Select **Yes** to have the script search all sub-directories of the directory you specified in *Directory to monitor*. The default is unselected.
File name (can use wildcards *, ? and %)	Specify the name of the ASCII file in which you want to search. You can use wildcards to specify filenames. The default is logfile*.log. Use the `*` wildcard to match any sequence of zero or more characters. For example, `*.log` instructs the script to search all .log files. Use the ? wildcard to match any single character. For example, `Log0?` instructs the script to search for any file whose name begins with `Log0` and includes one other character. **NOTE:** You can use multiple instances of the `*` and ? wildcards to specify filenames; for example: `*log*.log or ??log.log.` Use the % wildcard as a placeholder for the date format specified in *Date selection format*. For example, if you routinely generate a new file of the same name each day and append the filename with a date, you can use this wildcard to tell the script to always search the latest version of the file. Use this wildcard in place of the date added to the filename. For example, if your file is `Log`<*date*>, specify the filename in this parameter as `Log%`.
Date selection format	Select the date format. If you are searching files that contain a date as part of the filename, as specified in *File name (can use wildcards *, ? and %)*, you can use this parameter to select the format.
Search patterns	Specify the string for which you want to search. Separate multiple string entries by commas. **NOTE:** The strings you enter cannot contain commas, because commas are used to separate strings from one another.

Parameter	How to Set It
Threshold - Maximum number of matching lines	Specify the maximum number of matches to your search criteria that can be found before an event is raised. The default is 0.
Enforce case-sensitive match?	Select **Yes** to enforce a case-sensitive match to your search criteria. The default is unselected. For example, if set to Yes, search criteria of `E*.log` would match `Error.log`, but not `error.log`.
Require literal match?	Select **Yes** to enforce a literal match to your search criteria, where the exact string entered in the *Search Patterns* parameter will be sought. The default is unselected. If this parameter is unselected, and multiple words, separated by white space, are entered in the *Search Patterns* parameter, the script will search for each of the words in each line of the monitored file. If this parameter is selected, and multiple words, separated by white space, are entered in the *Search Patterns* parameter, the script will search for the entire string as it is specified.
Scan entire file on first iteration?	Select **Yes** to scan the entire file on the first iteration of the job. If set to No, the default, the first iteration of the job places a marker at the end of a file and scans from that point on during subsequent iterations.

8.5 AsciiLogRX

Use this Knowledge Script to monitor an ASCII text file for specific strings and messages logged since the last monitoring interval. This script allows you to use regular expressions to specify a pattern or search string to search for in an ASCII file. The script reports the matching entries found in the monitoring period. The script checks for changes to the text file that match the string you enter; it does not re-scan the entire file at each interval. The script gathers up to 2 MB worth of result matches for each iteration of the job.

For more information, see Creating Filters with Regular Expressions for General_AsciiLogRX.

In the first interval, the script reads the file and inserts a marker at the end of the file. The script does not search for a specified search string during the first interval. In subsequent intervals, the script checks the file for changes that match the search string you specified. The script raises an event if the number of lines matching your search criteria exceeds the threshold you set.

NOTE: This script reports the number of matched lines in each iteration and the detail message contains the text data. If the detail message is larger than 32 KB, the data is saved in a file on the managed computer (for example, `C:\program files\netiq\appmanager\bin\log`) and the detail message contains the truncated data. If you generate these log files, periodically remove the files when you are done with them.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Once every hour**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if matches are found?	Select **y** to raise events if text strings or messages that match your search criteria are found. The default is n.
Collect data for matches to search criteria?	Select **y** to collect data for charts and reports. If enabled, data collection returns one or more datastreams for each of your search criteria. The default is n. For example, if you search for `logon` and `logoff`, and `logon` is found in `C:\Log01` and `C:\Log02`, but `logoff` is not found, the script will return three datastreams: ◆ Instances of logon in `C:\Log01` ◆ Instances of logon in `C:\Log02` ◆ Instances of logoff Each data point in a datastream contains the number of matches found for that iteration of the script.
File name	Specify the full path to the file you want to monitor. For example `C:\temp\backup.log`. UNC names are also supported, such as `\\ENG\appdev\mylog.txt`. **Tip** You can only specify one filename for any job instance. To monitor multiple logs or files, create separate Knowledge Script jobs.
Enforce case-sensitive match?	Select **Yes** to enforce a case-sensitive match to your search criteria. The default is n. For example, if set to Yes, search criteria of `Error.log` would match `Error.log`, but not `error.log`.
Find pattern	Specify a regular expression to identify the string you want to find in the specified file. The default is a blank string, which instructs the script to find all new strings entered since the last time the script ran.
Threshold - Maximum number of matching lines	Specify the maximum number of matches to your search criteria that can be found before an event is raised. The default is 0.
Event severity when matches are found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which matches to your search criteria are found. The default is 5.

8.6 ConfigMachineDown

Use this Knowledge Script to set parameter values in the local repository of the computer on which you run it. The values are used by the MachineDownLR Knowledge Script when it runs on that computer. Using this pair of scripts, you can set up individual computers in a group so that when MachineDownLR runs on the group, it can run with different parameter values on each computer. This is particularly useful for enforcing monitoring policies.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if parameter values set successfully?	Set to **y** to raise an event if the script successfully sets parameter values in the local repository. The default is y.
List of computers	Specify all the computers you want this computer to monitor when MachineDownLR runs on it.
Full path to file with a list of computers	Specify the path and filename of a text file containing a list of computers for this computer to monitor when MachineDownLR runs on it. The file must be a text file with the name of each computer on a separate line, with no commas or spaces.
Event severity when parameter values set successfully	Set the event severity level, from 1 to 40, to indicate the importance of an event in which this script successfully sets parameter values in the local repository. The default is 25 (blue event indicator).

8.7 Counter

Use this Knowledge Script to monitor Performance Monitor counters. You can run this script on any computer or server to monitor any counter available in Performance Monitor. You can configure the script to raise an event if the value of the counter you select exceeds or falls below the threshold you set. You can also specify a consecutive number of times that a threshold must be exceeded before an event is raised.

Use this Knowledge Script to yield performance information for the counters you want to monitor. When this script collects and graphs data, the results are similar to the results displayed in Performance Monitor. Use the counter data to start corrective actions when thresholds are exceeded, generate more complex and sophisticated graphs, and provide historical information for reporting, trend analysis, and capacity planning.

Prerequisites

Requirements for Windows Server 2012, Windows 8, Windows 7, Windows 2008 R2, and Windows 2008:

The Log On As account under which the AppManager agent runs for these Windows operating systems must be a domain account and belong to the local Administrator group.

Requirements for Windows Server 2003:

- The Log On As account under which the AppManager agent runs on Windows Server 2003 must belong to the Performance Monitor Users policy.
- If the Operator Console or Control Center is installed on Windows Server 2003, the user account under which the console application runs must belong to the Performance Monitor Users policy.

To check the local policy:

1. At a Command Prompt, type `gpedit.msc` and press `Enter`.
2. In the Group Policy snap-in, double-click **Computer Configuration > Windows Settings > Security Settings > Local Policies > User Rights Assignment**.
3. In the **Local Setting** column, ensure the appropriate user account belongs to the **Performance Monitor Users** policy.

If the Operator Console or Control Center is installed on Windows Server 2003, the Remote Registry service on the console computer must be running. If the Remote Registry service is down when you attempt to configure this script by browsing counter information on the remote computer, the console displays an error message that indicates AppManager was unable to connect to the remote computer.

Requirements for Windows Vista:

If the Operator Console or Control Center is installed on Windows Vista, the Remote Registry service on the console computer must be running. If the Remote Registry service is down when you attempt to configure this script by browsing counter information on the remote computer, the console becomes unresponsive.

Resource Object

Windows computer

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the Counter job fails. The default is 5.

Parameter	How to Set It
Event severity when no counter or instance is found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the script cannot find the counter or instance you specified. The default is 15 (yellow event indicator).
Raise event when counter equals a specific value?	Select **Yes** to raise an event if the counter equals a specific value The default is unselected.
Event severity when counter equals a specific value	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the counter equals a specific value. The default is 15.
Value to match	Specify the number you want the counter to match so an event is generated. The default is 100.
Raise event when counter value exceeds threshold?	Select **Yes** to raise an event if the counter value exceeds the threshold *n* consecutive times. Specify the value of *n* in the *Consecutive times threshold can be crossed before event is raised* parameter. The default is Yes.
Event severity when counter value exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a value exceeds the threshold you set. The default is 8 (red event indicator).
Threshold - Maximum counter value	Specify the highest value a counter can attain before an event is raised. The default is 600.
Raise event when counter value falls below threshold?	Select **Yes** to raise an event if the counter value falls below the threshold *n* consecutive times. Specify the value of *n* in the *Consecutive times threshold can be crossed before event is raised* parameter. The default is Yes.
Event severity when counter value falls below threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a value falls below the threshold you set. The default is 8 (red event indicator).
Threshold - Minimum counter value	Specify the lowest value a counter must maintain to prevent an event from being raised. The default is 20.
Monitoring	

Parameter	How to Set It

Path of counter to monitor	Provide the name of the '\\[computer or cluster name]\object\counter\instance you want to monitor, or click **Browse [...]** to select the computer and counter you want to monitor. You can also select a counter to monitor by starting System Monitor and clicking **Add** [**+**] in the toolbar. The default is `Objects\Threads\`.
	If typing the name of the counter to monitor, use the following formats:
	`object\counter name\instance name`
	`object\counter name\<instance name substring>*`
	`object\counter name\*<instance name substring>`
	`object\counter name\instance name(instance_index)`
	`object\counter name\parent instance name ==> child instance name`
	`object\counter name\parent instance name ==> child instance name(instance_index)`
	In addition, a computer name or the name of a Microsoft cluster counters may be installed under may be specified preceding 'object', as in: `\\<computer name or name of cluster>\object\counter name\instance`.
	For more information, see "Examples of Using This Script" on page 275.
	Tips
	◆ If an instance is a parent of multiple instances (for example, if you have a Logical Disk 0 with partitions C: and D:), enter the complete instance name exactly as displayed in System Monitor. For example: `0 ==> C:`
	◆ To monitor multiple instances of the same instance name, use one of the following methods:
	◆ Indicate the instance index in parentheses to monitor specific instances. For example: `Process\% Privileged Time\netiq,netiq(1),netiq(2)`
	◆ Use an asterisk (*) after the instance name to monitor all instances that begin with the string you provide. For example: `Process\% Privileged Time\netiq*`
	◆ Use an asterisk (*) before the instance name to monitor all instances that end with the string you provide. For example: `Process\% Privileged Time\*netiq`

Consecutive times threshold can be crossed before event is raised	Specify the number of consecutive times a counter value can exceed or fall below the threshold before an event is raised. The default is 1 time. **Tip** This parameter provides functionality similar to that of the *Raise event if event condition occurs x times within y job iterations* parameter on the Advanced tab. NetIQ recommends using the *Consecutive times threshold ...* parameter when you run this script. The *Consecutive times threshold ...* parameter is designed to match the event text that is particular to this script, which can vary depending on your entry for the *Name of counter to monitor* parameter. In summary, use the *Consecutive times threshold ...* parameter to raise events. Leave the *Raise event if event condition ...* parameter at the default setting of **1** time within **1** job iteration.
Data Collection	
Collect data for counter value?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns values for counters that exceed the threshold. The default is unselected.

Examples of Using This Script

The following are examples of providing information in the *Path of counter to monitor* parameter.

Simple Counter with No Instance Name

For example, to monitor the Cache Hit Ratio counter and create one datastream, set the *Path of counter to monitor* parameter as follows:

```
SQLServer\Cache Hit Ratio
```

or

```
\\MyMSCluster\SQL Server\Cache Hit Ratio
```

For this type of counter you can simply leave the instance parameters blank. If selecting this counter through the Counter Browser:

1. Click **Browse [...]** and select the target **Computer**.
2. Select **SQLServer** from the Object list.
3. Select **Cache Hit Ratio** from the Counter list and click **OK**.

Counter with Multiple Identical Instance Names

Assume you want to monitor the percentage of processor time used by several `cmd` processes running on a given computer. If you enter `cmd` as the instance name, only the first `cmd` process found is monitored.

To monitor additional `cmd` processes, or to select a specific `cmd` process rather than the “first found,” you need to specify the instance index. The simplest way to select multiple instances is through the Counter Browser.

The script will monitor the processor time for these three `cmd` processes and create three datastreams.

If you do not use the Counter Browser and there are multiple instances with the same name, you need to identify which instance to monitor using an instance index, with 0 indicating the first instance, 1 the second, and so on. If you do not enter an index, the first instance found is monitored. If you are typing the information in the *Name of counter to monitor* parameter, use one of the formats:

```
<object>\<counter>\<instance> (<instance_index>)
```

or

```
\\<cluster or computer name>\<object>\<counter>\<instance> (instance_index)
```

Counter with Parent-Child Instances

In some cases, an instance is a **parent** of multiple **child** instances. For example, if you have a Logical Disk 0 with partitions C: and D:, the logical disk 0 is the parent of the logical disk C: and the logical disk D: or \\MyMSCluster\LogicalDisk\% Free Space\0 ==> C:. In addition, there may be multiple child instances with identical names. For example, two processes called MSDEV may each have threads 0, 1, 2, and 3:

```
Instance    ProcessID  ThreadID Instance Index
----------------------------------------------
MSDEV ==> 0    361        495      0
MSDEV ==> 0    291        426      1
MSDEV ==> 1    361        275      0
MSDEV ==> 1    291        181      1
MSDEV ==> 2    361        471      0
MSDEV ==> 2    291        256      1
MSDEV ==> 3    361        376      0
MSDEV ==> 3    291        500      1
```

The simplest way to select these child instances is through the Counter Browser. If you do not use the Counter Browser and there are child instances, you need to identify which instance to monitor using the format <object>\<counter>\<parent_instance> ==> <child_instance>. For example:

```
LogicalDisk\% Free Space\0 ==> C:
```

If there are multiple child instances with the same name, you need to identify which child instances to monitor using an instance index, with 0 indicating the first instance, 1 the second, and so on. If you do not enter an index, the first instance found is monitored.

If you are typing the Counter to monitor, use the format: <object>\<counter>\<parent> ==> <child> (<index>). For example:

```
Thread\% Processor Time\MSDEV ==> 0 (0),MSDEV ==> 0 (1)
```

Using the example above, this counter would get % Processor Time for threads 495 (MSDEV process 361) and 426 (MSDEV process 291) and create two datastreams.

```
ID  Job KS Name          Legend
-------------------------------------------------------------
7   12  General_Counter  Thread-% Processor Time-MSDEV ==> 0(0)
6   12  General_Counter  Thread-% Processor Time-MSDEV ==> 0(1)
```

Format for Entering Counter Names without Browsing

To type counter names rather than use the Counter Browser, enter the complete instance name exactly as it is displayed in the Performance Monitor, including any spaces or spelling conventions.

To manually set the *Path of counter to monitor* parameter (without browsing), use one of the following formats, and you can prepend to any of these the name of a cluster or computer name in the format "\\MyMSCluster\object ....":

Counter Type	General Format to Use and Example
Single counter instance	<object>\<counter>\<instance_name>Process\% Privileged Time\cmd
Multiple instances	<object>\<counter>\<instance> (<instance_index>)Process\% Privileged Time\cmd (1),cmd (4)

Counter Type	General Format to Use and Example
Child instances	<object>\<counter>\<parent_instance> ==> <child_instance>`LogicalDisk\% Free Space\0 ==> C:`
Multiple child instances	<object>\<counter>\<parent> ==> <child> (instance_index)`Thread\% Processor Time\MSDEV ==> 0 (0),MSDEV ==> 0 (1)`

8.8 CounterCorrelate

Use this Knowledge Script to monitor any pair of System Monitor performance counters. You can run this script on any computer or server, and you can monitor any counters available in the System Monitor. You can observe either a maximum or minimum threshold for each counter you are monitoring. You can set the script to raise an event if the value of either counter exceeds or falls below the threshold you set.

To see a list of available counters, click **Browse [...]** in the *Name of counter to monitor* parameter or start the System Monitor and click **Add [+]** in the toolbar.

Use this Knowledge Script to monitor for conditions when the values for any pair of counters indicate a problem. For example, you can raise events when CPU and memory counters both exceed a high threshold, or when a data file size counter exceeds a high threshold and an available disk space counter falls below a low threshold.

Prerequisites

Requirements for Windows Server 2012, Windows 8, Windows 7, Windows 2008 R2, and Windows 2008:

The Log On As account under which the AppManager agent runs for these Windows operating systems must be a domain account and belong to the Administrator local group.

Requirements for Windows Server 2003:

- The Log On As account under which the AppManager agent runs on Windows Server 2003 must belong to the Performance Monitor Users policy.
- If the Operator Console or Control Center is installed on Windows Server 2003, the user account under which the console application runs must belong to the Performance Monitor Users policy.

To check the local policy:

1. At a Command Prompt, type `gpedit.msc` and press `Enter`.
2. In the Group Policy snap-in, double-click **Computer Configuration > Windows Settings > Security Settings > Local Policies > User Rights Assignment**.
3. In the **Local Setting** column, ensure the appropriate user account belongs to the **Performance Monitor Users** policy.

If the Operator Console or Control Center is installed on Windows Server 2003, the Remote Registry service on the console computer must be running. If the Remote Registry service is down when you attempt to configure this script by browsing counter information on the remote computer, the console displays an error message that indicates AppManager was unable to connect to the remote computer.

Requirements for Windows Vista:

If the Operator Console or Control Center is installed on Windows Vista, the Remote Registry service on the console computer must be running. If the Remote Registry service is down when you attempt to configure this script by browsing counter information on the remote computer, the console becomes unresponsive.

Resource Objects

Windows computer or application server, such as Exchange Server, SQL Server, IIS server

Default Schedule

The default interval for this script is **Every 5 minutes.**

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Collect data for counter value?	Select **yes** to collect data for charts and reports. If enabled, data collection returns values for counters that exceed the threshold. The default is n.
Raise event if counter value exceeds or falls below threshold?	Select **yes** to raise an event when a counter value counter exceeds or falls below the threshold you set. The default is yes.
Counter 1 Settings	
Counter value for threshold	Specify the value for the threshold you want to observe. The default is 600.
Use counter value as maximum threshold?	Select **yes** to use the value from the *Counter value for threshold* parameter as a maximum threshold. The script then raises events when the counter value exceeds the threshold. Deselect the **yes** to use the value from the *Counter value for threshold* parameter as a minimum threshold. The script then raises events when the counter value falls below the threshold. The default is yes.
Name of counter to monitor	Specify the object\counter\instance name or click **Browse [...]** to select the object, counter, and instances to monitor. If typing the name, use the format `<object>\<counter>\<instance>`. You can enter multiple instances, separated by commas. For example: `Process\% Privileged Time\mapisp32,mqsvc` If an instance is a parent of multiple instances (for example, if you have a Logical Disk 0 with partitions C: and D:), enter the complete instance name exactly as displayed in Performance Monitor (for example “`0 ==> C:`”). For more information, see “Examples of Using This Script” on page 275.
Counter 2 Settings	

Parameter	How to Set It
Counter value for threshold	Specify the value for the threshold you want to observe. The default is 600.
Use counter value as maximum threshold?	Select **yes** to use the value from the *Counter value for threshold* parameter as a maximum threshold. The script then raises events when the counter value exceeds the threshold. Deselect the **yes** to use the value from the *Counter value for threshold* parameter as a minimum threshold. The script then raises events when the counter value falls below the threshold. The default is yes.
Name of counter to monitor	Specify the object\counter\instance name or click **Browse [...]** to select the object, counter, and instances to monitor. If typing the name, use the format `<object>\<counter>\<instance>`. You can enter multiple instances, separated by commas. For example: `Process\% Privileged Time\mapisp32,mqsvc` If an instance is a parent of multiple instances (for example, if you have a Logical Disk 0 with partitions C: and D:), enter the complete instance name exactly as displayed in the Performance Monitor (for example “`0 ==> C:`”). See System Monitor for the exact spelling of counter names and details about what each counter represents.
Event severity when counter value exceeds or falls below threshold	Set the severity level, from 1 to 40, to indicate the importance of an event if a counter value exceeds or falls below the threshold you set. The default is 8 (red event indicator).

8.9 EventLog

Use this Knowledge Script to monitor and filter information in custom Windows Event Logs. With this script, you can track Windows event log entries that match filtering criteria. This script works on an incremental basis; it does not fully rescan the entire event log each time it runs. This script returns all event log entries that match the filtering criteria in the event or data point detail message. If you want to monitor the custom Windows event logs on agent computers running Windows Server 2008 or newer, you must have version 3.5 or later of Microsoft's .Net Framework and AppManager for Microsoft Windows 8.0 or later is present on the agent computer, on which you want to run a General_EventLog job.

If you want to monitor a custom event log that appears under **Applications and Services Logs** in the Microsoft Event Viewer, create the following registry key for that log if the key does not exist:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\eventlog\<Event log name>
```

In addition, you must provide a registry key and values for each event source in the custom event log:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\eventlog\<Event log
name>\<Event source>
```

Under the `EventSourceName` registry key, create a new string registry value labeled `EventMessageFile` with the path to the message file used by the custom log. As a result, AppManager loads the full event description text from the message file and displays that text in AppManager events.

NOTE

- Only the most recent batch of events can be viewed in the data point detail message. For example, assume you set this script to scan all previous entries in the event log and list ten matching entries in each event detail message. When the job runs, 30 entries are found that match your filtering criteria. In this case, the job creates three child events for the interval, and each child event contains ten entries: the oldest matching entries in one child event batch, the second oldest in Batch 2, and the most recent in Batch 3. If this same job is collecting data and you view the data detail message for the interval, only the entries from the third child event (Batch 3) are displayed.
- When you use text or numeric strings in the *Event [...] filter* parameters, this script searches event logs and matches the text or numeric string to any part of the event entry. The results are not exact matches. For example, if your filter string is “foo,” results will include “foobar,” “foo,” and “food.”

Resource Objects

Windows computer or application server, such as Exchange Server or SQL Server

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Job Failure Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the EventLog job fails. The default is 5.
Event Log Monitoring	
Event logs to monitor	Provide a comma-separated list of the event logs you want to monitor. For example: `System,Application,Security` The default is `Application`.

Parameter	How to Set It
Number of previous hours to scan logs	Set this parameter to control how the script scans the logs at the first interval, after which scanning begins where the previous scan ended. Enter one of the following values: ◆ **-1** -- to scan all the existing entries ◆ *N* -- to scan entries only for the past *n* hours (8 for the past 8 hours, 50 for the past 50 hours, for example) ◆ **0** -- to not scan previous entries; only search from this moment on. The default is 0.
Maximum number of entries per event report	Specify the maximum number of entries to be recorded in each event's detail message. If this script finds more entries from the log than can be put into one event message, it will return multiple events to report all the outstanding entries in the log. The default is 30 entries. If this script encounters one or more very large events in the Windows Event log, this script may error out and generate an event message “`Out of string space`." If this occurs, you can usually work around the problem by adjusting this parameter to a smaller value.
Ignore event log matches occurring during agent maintenance mode?	Select **Yes** for the Knowledge Script to ignore event log matches that occur while the agent is in maintenance mode. No events will be raised or data collected for matches that are written to the event logs during this time. The default is unselected.
Event Log Filters	
Event Types	
Monitor critical events?	Select **Yes** to monitor error event entries. The default is Yes.
Monitor error events?	Select **Yes** to monitor error event entries. The default is Yes.
Monitor warning events?	Select **Yes** to monitor warning event entries. The default is Yes.
Monitor information events?	Select **Yes** to monitor information event entries. The default is Yes.
Monitor success audits?	Select **Yes** to monitor success audit event entries. Success audits are successful security access attempts that are audited. The default is Yes. **NOTE:** This option only applies to computers with WiN2003.
Monitor failure audits?	Select **Yes** to monitor failure audit event entries. Failure audits are failed security access attempts that are audited. The default is Yes. **NOTE:** This option only applies to computers with WiN2003.

Parameter	How to Set It
Event source filter	Use this parameter to filter for events generated by a particular source, which can be the name of a program, a system component, or a component of a large program. For example, SQLExecutive, SNMP, or the Service Control Manager. Provide a search string. This script will look for matching entries in the Event Log **Source** field. Separate multiple strings with commas. The search string can contain criteria to include entries, exclude entries, or both. Separate include and exclude criteria with a colon (:) using the following format: `include:exclude`. For example, to include all SQL sources and to exclude all SNMP sources, enter the following: `SQL:SNMP` If you specify only include criteria, the colon is not necessary.
Event category filter	Use this parameter to filter for events in a particular category, such as Server or Logon. Provide a search string. This script will look for matching entries in the Event Log **Category** field. Separate multiple strings with commas. The search string can contain criteria to include entries, exclude entries, or both. Separate include and exclude criteria with a colon (:) using the following format: `include:exclude`. For example, to include the Server category and to exclude the Logon category, enter the following: `Server:Logon` If you specify only include criteria, the colon is not necessary.

Event ID filter	Use this parameter to filter for particular event IDs.
	Provide a search string or ID range, for example 100-2000). This script will look for matching entries in the Event Log **Event** field. Separate multiple IDs and ranges with commas. For example:
	`1,2,10-15,202`
	The search string can contain criteria to include entries, exclude entries, or both. Separate include and exclude criteria with a colon (:) using the following format: `include:exclude`. For example, to include event IDs 10 through 15 and to exclude event ID 202, enter the following:
	`10-15:202`
	If you specify only include criteria, the colon is not necessary.

Parameter	How to Set It
Event user filter	Use this parameter to filter for events associated with a particular user. Provide a search string, for example, `<domain name>\<user name>`. This script will look for matching entries in the Event Log **User** field. Separate multiple strings with commas. The search string can contain criteria to include entries, exclude entries, or both. Separate include and exclude criteria with a colon (:) using the following format: `include:exclude`. For example, to include events for user Joe and exclude events for user Sam, both of whom are in the `RALQE` domain, enter the following: `RALQE\Joe:RALQE\Sam` If you specify only include criteria, the colon is not necessary.
Event computer filter	Use this parameter to filter for events generated by a particular computer. Provide a search string. This script will look for matching entries in the Event Log **Computer** field. Separate multiple strings with commas. The search string can contain criteria to include entries, exclude entries, or both. Separate include and exclude criteria with a colon (:) using the following format: `include:exclude`. For example, to include all computers with `SFO` in the hostname and to exclude all computers with `RDU` in the hostname, enter the following: `*SFO*:*RDU*` If you specify only include criteria, the colon is not necessary.
Event keywords filter	Use this parameter to keyword for events generated by a particular computer. Provide a search string. This script will look for matching entries in the Event Log **Computer** field. Separate multiple strings with commas. The search string can contain criteria to include entries, exclude entries, or both. Separate include and exclude criteria with a colon (:) using the following format: `include:exclude`. For example, to include all computers with `SFO` in the hostname and to exclude all computers with `RDU` in the hostname, enter the following: `*SFO*:*RDU*` If you specify only include criteria, the colon is not necessary.

Parameter	How to Set It
Event description filter	Use this parameter to filter for events with a particular detail description or containing keywords in the description. Provide a search string. This script will look for matching entries in the Event Log **Description** field. Separate multiple strings with commas. The search string can contain criteria used to include entries, exclude entries, or both. Separate include and exclude criteria with a colon (:) using the following format: `include:exclude`. For example, to include the keyword `error` and to exclude the keyword `RSVP`, enter the following: `error:RSVP` If you specify only include criteria, the colon is not necessary.
Event Notification	
Use XML format for event message	Select **Yes** for event detail created by this Knowledge Script to be composed of XML. The default is unselected. **NOTE:** This parameter is only applicable when the agent computer is running version 8.0 or later of AppManager for Microsoft Windows.
Raise event if log entries matching criteria are found?	Select **Yes** to raise an event when log entries match your filtering criteria. The default is Yes.
Raise event grouped by EventID	Select **Yes** to raise an event classified based on each event ID. The default is unselected.
Raise event only when event log threshold is crossed?	Select **Yes** to raise an event when the threshold is crossed. The default is Yes.
Threshold value per event log	Specify the maximum number of matches to your search criteria that can be found before an event is raised. The default is 1.
Event severity when log entries match criteria	Set the event severity level, from 1 to 40, to indicate the importance of an event in which log entries match your search criteria. The default is 15 (red event indicator). **Tip** You can adjust the severity based on which log or type of event you are checking for.
Raise event if log cannot be accessed?	Select **Yes** to raise an event when the log file cannot be read or reached. The default is Yes.
Event severity when a log is inaccessible	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the log file cannot be read or reached. The default is 10.
Data Collection	
Collect data for log entries that match criteria?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns detail about log entries that match your filtering criteria. The default is unselected.

Parameter	How to Set It
Separate data by log file?	Select **Yes** to separate event entries from different log files into different datastreams. If unselected, all event entries matching your filtering criteria are placed in the same datastream and the data detail message may include event entries from multiple log sources. For example, if you are monitoring both the System and Application logs, you can enable this parameter so that events in the System log are tracked separately from events in the Application log. The default is unselected.

Examples of How this Script Is Used

You can customize this script in many ways based on your requirements. For example, for general system events, you can set the following options when detecting security failures:

Properties and Parameters	How You Might Set Them
Schedule interval	10 minutes
Raise event if log entries match criteria?	Yes
Log files to filter	Security
Monitor failure audits?	Yes
Event severity when event log entries match criteria	2
Action	MapiMail

With this scenario, on the Schedule tab in the Knowledge Script Properties dialog box, set the interval to **Run every 10 minutes** because you want a short window for checking for this type of problem.

On the Values tab, enable the *Raise event if log entries match criteria?* parameter, indicate you will monitor failure audits in the Security log, and set the event severity to 2, indicating this is a very serious event that should be highly visible. Leave the other filtering options blank.

On the Action tab, indicate that you want an email sent when an event is raised. With these settings, AppManager will regularly check for security failures and will notify you, or whoever you designate, through email if any security failure events are detected.

Another example of how to use this script to detect all problems with your SQL Server could involve setting up the script job like this:

Properties and Parameters	How You Might Set Them
Schedule interval	30 minutes
Raise event if log entries match criteria?	Yes
Log files to filter	Application
Monitor error events?	Yes
Event source filter	MSSQLServer
Event severity when event log entries match criteria	8
Action	MapiMail

Another way you can use this script is to collect data and graph a trend chart from your System event log:

Properties and Parameters	How You Might Set Them
Schedule interval	1 hour
Collect data for log entries that match criteria?	Yes
Log files to filter	System
All other filters	not set
Action	Null

If you choose to collect data, the script returns the number of matched entries as the primary data point to be graphed. The first batch of filtered results can be viewed in the detail data message when you double-click a data point. Additional matching entries may be included in the graph. The peaks and valleys in the graph indicate a large number of events or low event activity.

8.10 EventLogRX

Use this Knowledge Script to scan the Windows logs you specify for entries that match the criteria you specify. You can filter the event log entries by event type and by specifying a combination of include and exclude strings for each event field using regular expressions. This script raises an event if a log entry matches all the filter criteria you specify. All event log entries that match the filtering criteria are returned in the event detail message.

Use the *Filter the [...] field with the regular expression* parameters to control which fields to filter and the filtering criteria to use to find specific information, such as events associated with a specific user or computer name. With this script, you specify the filtering criteria for each field you are interested in using a regular expression or you can specify the name of a file that contains all your filtering criteria.

For more information, see Creating Filters with Regular Expressions for General_EventLogRx.

You can use the *Events in past N hours* parameter to determine the number of previously recorded event entries, if any, to scan for matches. For example, if you want to check whether any event entries recorded in the last two hours, on the first job iteration, match your filtering criteria, you would set this parameter to 2. To scan the entire log for any previously reported events, set the *Events in* *past N hours* parameter to -1. After the Knowledge Script job completes its first iteration, only new entries written to the event log that match your criteria are reported. When the *Events in past N hours* parameter is set to 0, the script does not scan the log for any previously reported events.

Prerequisite

This script requires the Async managed object to be installed and the Microsoft EventLog service to be running on the computer you want to monitor.

Resource Objects

Windows computer or application server, such as Exchange Server or SQL Server

Default Schedule

The default interval for this script is **Every 10 minutes.**

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Job Failure Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the EventLogRX job fails. The default is 5.
Event Log Monitoring	
Event log files to monitor	Specify the event log you want to monitor. You can specify multiple event logs, separated by commas. For example: `System,Application,Security`. The default is `Application`. If you do not specify an event log, AppManager monitors all logs. **Notes** ◆ If, in addition to these event logs, you specify a filter file in the *Full path to a file containing filtering criteria* parameter, AppManager ignores the *Filter the [...] field with the regular expression* parameters, but continues to scan the log file you specified. ◆ If the event log you specify does not exist on the target computer, the Application log is automatically monitored.

Parameter	How to Set It
Number of previous hours to scan logs	Set this parameter to control how the script scans the logs at the first interval, after which scanning begins where the previous scan ended. Enter one of the following values: ◆ **-1** to scan all the existing entries ◆ *N* to scan entries only for the past *n* hours (8 for the past 8 hours, 50 for the past 50 hours, for example) ◆ **0** to not scan previous entries; only search from this moment on. The default is 0.
Enforce case-sensitive filters?	Select **Yes to** make all filter statements for this script case-sensitive. The default is unselected.
Maximum number of entries per event report	Specify the maximum number of entries to be recorded in each event's detail message. If this script finds more entries from the log than can be put into one event message, it will return multiple events to report all the outstanding entries in the log. The default is 30 entries. If this script encounters one or more very large events in the Windows Event log, this script may error out and generate an event message "`Out of string space`." If this occurs, you can usually work around the problem by adjusting this parameter to a smaller value.
Ignore event log matches occurring during the agent maintenance mode?	Select **Yes** for the Knowledge Script to ignore event log matches that occur while the agent is in maintenance mode. No events will be raised or data collected for matches that are written to the event logs during this time. The default is unselected.
Full path to a file containing filtering criteria	Type the full path to a file containing the filtering criteria you want to match if you want to specify matching expressions in an external file. For example: `C:\TEMP\MyFilters.txt`. **NOTE:** If you specify a filter file, AppManager ignores the *Filter the [...] field with the regular expression* parameters, but continues to scan the log file specified in the *Log files to filter (Application, Security, System)* parameter. However, if AppManager cannot process the filter file, the script raises an event (for example, `fail to process filter file C:\general.xml`) and continues to scan the log file using the filtering criteria you specified in the *Filter the [...] field with the regular expression* parameters.
Event Log Filters	

Parameter	How to Set It
Filter the [...] field with a regular expression	Use a regular expression to specify the criteria to look for in each event log field you want to monitor: ◆ **Type**. To filter information based on the type of event (for example, Error, Warning, Information, Audit_Success, Audit_Failure), use a regular expression to identify the type of event entries to include. ◆ **Source**. To filter the entries generated by a particular source (for example `SQLExecutive`, `SNMP`, or `Service Control Manager`), use a regular expression to identify the source of event entries to include. ◆ **Category**. To filter information based on a particular category (for example Server or Logon), use a regular expression to identify the category of event entries to include. ◆ **Event ID**. To filter information based on the event ID, use a regular expression to identify the event IDs to include. ◆ **User**. To filter information based on the user name, use a regular expression to identify the user names to include. ◆ **Computer**. To filter information based on the computer name, use a regular expression to identify the computers to include. ◆ **Keywords**. To filter information based on the keywords of an event, use a regular expression to identify the keywords to include. ◆ **Description**. To filter information based on the event description, use a regular expression to indicate the description to include. **NOTE:** If you specify a filter file in the *Full path to a file containing filtering criteria* parameter, AppManager ignores the *Filter the [...] field with the regular expression* parameters, but continues to scan the log file specified in the *Log files to filter (Application, Security, System)* parameter.
Event Notification	
Use XML format for event message	Select **Yes** for event detail created by this Knowledge Script to be composed of XML. The default is unselected. **NOTE:** This parameter is only applicable when the agent computer is running version 8.0 or later of AppManager for Microsoft Windows.
Raise event if log entries matching criteria are found?	Select **Yes** raise an event when log entries match your filtering criteria. The default is Yes.
Event severity when log entries match criteria	Set the event severity level, from 1 to 40, to indicate the importance of an event in which log entries match your search criteria. The default is 15 (red event indicator). **Tip** You can adjust the severity based on which log or type of event you are checking for.
Raise event if log cannot be accessed?	Select **Yes** to raise an event when the log file cannot be read or reached. The default is Yes.
Event severity when a log is inaccessible	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the log file cannot be read or reached.The default is 10.
Data Collection	

Parameter	How to Set It
Collect data for log entries that match criteria?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns detail about log entries that match your filtering criteria. The default is unselected.
Separate data by log file?	Select **Yes** to separate event entries from different log files into different datastreams. If unselected, all event entries matching your filtering criteria are placed in the same datastream and the data detail message can include event entries from multiple log sources. The default is unselected. For example, if you are monitoring both the System and Application logs, you can enable this parameter to track events in the System log separately from events in the Application log.

Examples of How this Script Is Used

Using this script you can specify regular expressions for each event log field as Knowledge Script properties or maintain your search criteria independent of the script parameters in a separate filter file.

In many cases, specifying an external filter file provides greater flexibility and makes modifying your search criteria more straightforward because you can add almost any number of expressions and you do not need to modify the Knowledge Script properties to pick up your changes.

If you want to use a filter file:

- Identify the strings that you want to find a match for (that is, the entries you want to include in your results).
- Create a text file with one regular expression string per line to locate matching strings. Each line in the file consists of a parameter keyword followed by a colon (:), a tab or blank space, and the regular expression. Or the filter file can be written using XML.
- Make sure the file exists on the target computer.
- Type the absolute path to the file on the local computer in the *Full path to a file containing filtering criteria* parameter and start the job.

Formatting the Filter File

There are two valid formats for the filter file: a simple table format to define the strings to include and an XML format that allows you to define more complex include and exclude filtering. For both formats, the parameter name keywords are required, but the field values can be left blank if no filtering is needed.

Select a file format appropriate for the complexity of the filtering you need to do.

Table Format

The table format provides a simple way to create the filter file. Each filtering section in the file begins with `EventStart` and ends with `EventEnd`. If an entry in the event log matches all the criteria you have specified within a filtering section, it is considered a match and an AppManager event is raised. If you have more than one filtering section, an entry matching either section raises an event.

For example, the following table format provides two filter sections:

```
EventStart
CaseSensitive:n
Log:System
Type:Error|Warning|Information
Source:^SQL*
Category:*
EventID:1[0-9][0-9][0-9]
User:Sam|Joe|Chris
Computer:SFO*
Description:($Error.*)|(.*error.*occurred.$)
EventEnd
EventStart
CaseSensitive:n
Log:Application
Type:Error|Warning|Information
Source:^SQL*
Category:*
EventID:1[0-9][0-9][0-9]
User:Sam|Joe|Chris
Computer:SFO*
Description:($Error.*)|(.*error.*occurred.$)
EventEnd
```

NOTE: If you create only one filter section, you do not need to include the `EventStart` and `EventEnd` lines in the file. These lines are only required if you have more than one filtering section.

XML Format

The XML format is somewhat more sophisticated and more flexible than the table format. The XML format allows you to set both include and exclude filters using the `<Include>` and `<Exclude>` tags and to combine these filter sets to define the search criteria. Each filtering section in the file begins with the `<Events>` tag. A log entry must match all the criteria you specified within a filtering section for it to be considered a match.

For example:

```
<?xml version = "1.0" standalone = "yes"?>
<EventLogConfig Name = "Event Filter" Type = "EVENT_FILTER_CUSTOM" ID = "76">
<Include>
   <Events>
      <Log>Application</Log>
      <Type>Information|Warning|Error</Type>
      <Source><Net*]></Source>
      <Category>*</Category>
      <EventID>2*</EventID>
      <User>*</User>
      <Computer>*</Computer>
      <Description><![CDATA[Event.]]></Description>
      <CaseSensitive>y</CaseSensitive>
   </Events></Include>
</EventLogConfig>

<Exclude>
   <Events>
      <Log>Application</Log>
      <Type>Warning</Type>
      <Source>RSVP</Source>
      <Category>*</Category>
      <EventID>2468</EventID>
      <User>*</User>
      <Computer>SHASTA</Computer>
      <Description>RSVP*</Description>
      <CaseSensitive>y</CaseSensitive>
   </Events>
</Exclude>
</EventLogConfig>
```

NOTE: If a field contains a regular expression that conflicts with XML syntax or includes special characters, you can use `![CDATA[regular_expression]]` to enclose the expression and prevent parsing problems.

8.11 MachineDown

Use this Knowledge Script to detect whether the computer on which you run the script can communicate with one or more specified Windows computers.

This script does **not** require the AppManager agent to be installed on the remote computers you want to monitor.

To run this script on a Windows Vista computer, the Remote Registry service on the agent computer must be running to connect to the Windows registry on the remote computers you want to monitor. If the Remote Registry Service is down when this script runs, an event is raised to indicate the remote computer was unresponsive and the connection to the Windows registry failed.

You can select computers by browsing the AppManager repository, specifying a list of computers using the *Computers to monitor* parameter, or naming a file that contains a list of computer names or addresses. Browse the AppManager repository to select the remote computers you want and prevent event information from appearing in AppManager while the computer is in maintenance mode.

If you specify a list of computers, instead of browsing the repository for the computers you want, this script displays event information in AppManager even if the remote computer is in maintenance mode.

When typing a list of Windows computers, you can specify computers that are not currently in the Navigation pane or the TreeView pane.

When you run this script on a computer, the script tries to communicate with each of the computers you specified in the *Computers to monitor* parameter.

This script attempts to communicate by:

- Checking name-to-IP-address resolution
- Executing an Internet Control Message Protocol (ICMP) ping
- Connecting to the Windows registry

This script raises an event if any of these attempts fail.

You can also instruct the script to ping specific router IP addresses before attempting to communicate with any of the specified computers. This provides an additional test of the network connection between the computer on which the script is running and the monitored computers. If this test is successful, it eliminates one reason for a lack of communication between computers.

This script does not monitor the computer where the script itself is running. For example, if you run this script on a server named SERVER01 and use the *Select computers from the Repository* parameter to select the server SERVER01 (either explicitly or as a member of a group or view), the script automatically excludes SERVER01 at run time because it does not make sense to monitor the local computer’s availability. If the script is running, the computer must be available. If the script is not running, either the local computer is down or the script or agent has been stopped.

To monitor the local computer, create a second MachineDown job running on a different computer that monitors the local computer in question. In this case, you could have a server SERVER02 running the script and monitoring SERVER01 and server SERVER01 monitoring server SERVER02. If both jobs are collecting data, be careful that the two scripts are not monitoring the same computers, for example, SERVER01 and SERVER02 should not both monitor SERVERA. This would result in two datastreams collecting uptime information for the same server (SERVERA), which can cause the ComputerAvailability report to miscalculate the uptime for SERVERA.

In some cases, this script may not be able to communicate with one or more remote computers because AppManager does not have sufficient privileges to access those remote machines. To avoid this problem, grant Admin privileges to the AppManager agent’s user account or use the PingMachine Knowledge Script to check connectivity.

If you select target computers by browsing the AppManager repository, the logon account for the agent on which the job is running must have sufficient privileges to query the AppManager repository.

If you select to include computers from the AppManager repository by View or Server Group, AppManager automatically includes the new computers on the next iteration. If you select to include computers from the AppManager repository by Computer, AppManager only monitors the computers that were selected. However, if you delete a monitored computer from the AppManager repository, AppManager does not monitor that computer unless you add it back into the AppManager repository.

AppManager also reads the server list file on every script iteration. If you remove a computer name from the server list file, starting with the next script iteration, AppManager no longer monitors the computer.

This script can check connections to computers that are across a firewall from the AppManager repository so long as the script is running on a computer on the same side of the firewall as the computers to which it is checking connections. Keep in mind, however, that under these circumstances you cannot select computers by browsing the AppManager repository unless the SQL Server communication ports are open in the firewall and the agent can query the AppManager repository. If you are using an agent across a firewall from the AppManager repository, you are advised to use the *Computers to monitor* or *Filename for computer list* parameter to specify computers.

If the computer that is down has been discovered and is displayed in the Navigation pane or the TreeView, that computer's icon blinks in the Navigation pane or the TreeView. If the computer that is down is not displayed, the computer where you ran the Knowledge Script blinks instead.

For computers running AppManager agents version .x and later where you want to use a monitoring policy, consider using the ConfigMachineDown and MachineDownLR Knowledge Scripts.

When configuring an action for this Knowledge Script, configure the Location to initiate the action on the MS (to run on the management server) or on a Proxy (to run on a particular managed client).

If you instead configure an action to run on the managed client (MC), when a remotely monitored computer is placed into machine maintenance mode (from AppManager) or scheduled maintenance mode (using the AMAdmin_SchedMaint Knowledge Script), any event conditions detected on the remote computer are ignored, but the action is not disabled. In this case, an action runs, but no event information appears on the **Events** tab.

Use the ReportAM_GeneralMachineDown Knowledge Script to generate a report about computers that were detected as down during a specified period.

If you are using the Web Console, the *Select computers from the repository* parameter is not supported. Instead, use the *Computers to monitor* parameter to specify the computers you want to monitor.

Using this Script to Monitor a Subnet

Run this script on a computer in the same subnet as the management server. When completing the *Computers to monitor* parameter, specify a limited number of computers that represent different subnets in your network.

You can then run additional MachineDown jobs on each of the computers specified in the first job to monitor the computers in each of their own subnets. This gives you coverage without stressing network bandwidth. It also ensures that, if a router or subnet is down, you receive only one event for the server being monitored from the agent on the management server's subnet. The other servers in that subnet will not post duplicate "Computer Down" events.

As an example, assume:

- The AppManager management server is installed on the computer `TARZAN` in subnet 1. Other servers in subnet 1 include `TITO` and `BLUE`.
- Subnet 2 includes the servers `PAOLO`, `BONN`, and `KENO`.
- Subnet 3 includes the servers `TRISTE`, `VOILA`, and `TONTO`.

You create a Knowledge Script job that runs on `TITO` (subnet 1, same as the management server) and set the *Computers to monitor* parameter to `PAOLO` (subnet 2) and `TRISTE` (subnet 3).

You then create a job (J-2) on PAOLO with the *Machine list* parameter set to BONN and KENO, and a job (J-3) on TRISTE with the Machine List set to VOILA and TONTO. Also create a job that runs on the management server (for example, TARZAN) that does a reciprocal check with the server in its own subnet (for example, TITO) in its *Computers to monitor* parameter.

TIP: If you want this Knowledge Script to raise an action when a connection is down, enable the *Managed Client Action* parameter in the Knowledge Script Properties dialog box for the job that monitors your subnets.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every 5 minutes**.

Be sure to schedule this job so that you allow enough time for the job to complete during the interval. As a general guideline, allow 20 to 30 seconds for each computer being monitored. This allows enough time for the connection to the registry on each computer.

You can use the following formula to calculate how many minutes are required for the job to complete:

```
(number of computers x 30 seconds)/60 = minutes for job to complete
```

For example:

```
(10 computers x 30 seconds)/60 = 5 minutes
```

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if a computer is down?	Select **Yes** to raise an event if a connection cannot be established to the target computer. The default is Yes. **NOTE:** For AppManager agents version 6.x and later, events raised for computers in maintenance mode are suppressed.
Require Windows Registry connection?	Select **Yes** to require the script to attempt a connection to the registry after it has attempted an ICMP ping. The default is Yes. This test is recommended because Windows can respond to ICMP ping requests even though the computer is in a blue screen state. A connection to the registry is further validation that the target computer is up. If you are using this script to check the status of UNIX machines, you must disable this option. **NOTE:** The account under which the AppManager agent is running must have sufficient privileges to connect to the registry.

Parameter	How to Set It
Event severity when computer is down	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a connection cannot be established to the target computer. The default is 5 (red event indicator).
Raise single event for all computers that are down?	Select **Yes** to raise only one event regardless of the number of computers that are down. The default is unselected. If you choose to raise only a single event, the information about specific computers is contained in the event detail message. The same rules for the suppression of events that apply to the *Raise event if a computer is down* parameter also apply here.
Raise event if specified router is down?	Select **Yes** to raise an event if a router specified in the *Router IP addresses* parameter is down. The default is Yes.
Severity - Router down	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a specified router is down. The default is 5 (red event indicator).
Raise event if default gateway is down?	Select **Yes** to raise an event when the default gateway is down. The default is Yes. If the default gateway is down, the script might not be able to connect to any of the computers you identified, and false events can be raised,
Event severity when default gateway is down	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the default gateway is down. The default is 5 (red event indicator).
Raise event if the computer list file is missing?	Select **Yes** to raise an event if the file containing the list of monitored computers cannot be found. The default is Yes.
Event severity when computer list file is missing	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the list of monitored computers cannot be found. The default is 15 (yellow event indicator).
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the MachineDown job fails. The default is 5 (red event indicator).
Data Collection	
Collect data for log server?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the availability, or status, of a specific computer you are monitoring. The default is unselected.
Collect single data point for number of servers down?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the number of unavailable, or down, computers for the monitored machines. The default is unselected.
Collect data for default gateway availability?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the availability, or status, of the default gateway of the computer that is running the job. The default is unselected.
Collect data for router availability?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns the availability, or status, of one or more routers that you configured in the *Router IP addresses* parameter. By default, data is not collected.
Monitoring	

Parameter	How to Set It
Select computers from the repository	Click **Browse [...]** to search the AppManager repository for the computers you want to monitor. You can select computers by view (for example, Master or NT), by server group, or individually. You can use this parameter as the sole selection method, or you can use it in conjunction with the *Computers to monitor* and *Filename for computer list* parameters. **NOTE:** Once you specify a list of computers with this parameter, the script always monitors a list of computers generated by this parameter. You can modify the list, but you cannot delete it. If you want to subsequently specify monitored computers without using this parameter, you need to run a new monitoring job with this script and leave this parameter blank. If you choose to select computers by server group, the server groups must actually contain agent computers. If you have a hierarchy of server groups where you want to choose a parent server group that contains child server groups, you must select the child server groups that have actual agent computers.
Computers to monitor	Specify a list of computers to monitor. Separate multiple names with commas and no spaces. For example, to check whether the Sales1 server can communicate with the computers `JOE`, `SAM`, and `PAT`, run this script on the `Sales1` computer and enter `JOE,SAM,PAT` in this field. You can use this parameter as the sole selection method, or you can use it in conjunction with the *Select computers from the repository* and *Filename for computer list* parameters.

Filename for computer list	Specify the path to the file that contains a list of computers you want to monitor, or click **Browse [...]** and navigate to the file.
	Use the local path to the file rather than the UNC path. For example, use `D:\<path to file>` rather than `\\<server>\D$\<path to file>`.
	The file should contain the hostname or IP address for each computer in one or more lines. Each line can have multiple computer names, separated by commas and with no spaces.
	For example:
	`NYC01,NYC02`
	`SALES01,10.15.221.5,SFO01`
	`LABMACH,QATEST`
	You can use this parameter as the sole selection method, or you can use it in conjunction with the *Select computers from the repository and Computers to monitor* parameters.

Router IP addresses	Specify the IP addresses of the routers through which the computer running the script should communicate with the target computers. **NOTE:** If one of the listed routers is down, none of the target computers will be monitored.
Number of seconds to wait for ping response	Set the maximum number of seconds to wait for a response from a target computer. The default is 3 seconds.

8.12 MachineDownLR

Use this Knowledge Script to detect whether the computer on which you run the script can communicate with one or more remote Windows computers. This script requires that you first use the ConfigMachineDown Knowledge Script to store a list of remote computers in the local repository on the managed client computer where this script runs.

This script does **not** require the AppManager agent to be installed on the remote computers you want to monitor.

To run this script on a Windows Vista computer, the Remote Registry service on the agent computer must be running to connect to the Windows registry on the remote computers you want to monitor. If the Remote Registry Service is down when this script runs, an event is raised to indicate the remote computer was unresponsive and the connection to the Windows registry failed.

Once you have run ConfigMachineDown on each computer in a group, you can use MachineDownLR in a monitoring policy for the group. On each computer, the script knows what to monitor because ConfigMachineDown previously stored that information in the local repository. The use of MachineDownLR is the same as for MachineDown.

Note that this script displays event information in AppManager even if the remote computer is in maintenance mode.

Example of How this Script Is Used

If you want each computer in your environment to be able to check whether other selected computers are down, run ConfigMachineDown on each computer and specify the particular machine list you want that computer to monitor.

You can then put the MachineDownLR jobs in a monitoring policy that covers all those computers. As the job runs on each computer, it picks up the machine list from the local repository where ConfigMachineDown set it.

In this way, each instance of MachineDownLR can check a different list of computers from each computer where it runs.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Collect data for computer availability?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the following: - **100** -- target computer is up, - **0** -- the target computer is down, or - **50** -- communication failed (for example, because a computer's IP address is not found). The default is n.
Event severity when computer is down	Set the severity level, from 1 to 40, to indicate the importance of an event in which the target computer is down. The default is 5 (red event indicator).
Ping router?	Set to **y** to routinely ping the default gateway router. If enabled and the ping fails, the script stops and raises an event. The default is n. When this script runs, it first pings the default gateway router if the Ping Router parameter is enabled. If the ping fails, an event is raised. If the ping is successful or if no ping is requested, this script checks the registry of the destination computer. If that check fails, an event is raised. It also traces the route to the destination if the Trace the route parameter is enabled. If the registry check succeeds, communication with that computer is verified.
Number of seconds to wait for ping response	Specify the maximum number of seconds to wait for the ping to return a positive result. If the *Ping router* parameter is set to n, this threshold parameter is ignored. The default is 3 seconds.
Trace the route to a destination computer	Set to **y** to trace the route to a computer that is down. A traceroute can help you determine where the problem lies. The default is n.
Raise single event for all computers that are down?	Set to **y** to raise a single event regardless of the number of computers that are down. Set to **n** to raise a separate event for each computer that is down. The default is n.

8.13 MissingEvent

Use this Knowledge Script to determine whether a Windows event log does not contain an expected entry. This script raises an event if the Application, Security, or System event log does not contain an entry that matches your filtering criteria. You can use regular expressions or text/numeric strings to specify filtering criteria.

For more information, see Section 8.1, "Creating Filters with Regular Expressions for General_AsciiLogRX," on page 262.

For example, the SQL backup process normally adds an entry to the event log to indicate databases were successfully backed up. This script can search for log entries that match your filtering criteria and raise an event if the event log does *not* contain an entry for a successful SQL backup.

To determine whether a Windows event log *does* contain an entry matching your filtering criteria, use the EventLog Knowledge Script.

NOTE: If you use text/numeric strings in the *Event [...] filter* parameters, this script searches event logs and matches the filter string to any part of the event entry. The results are not exact matches. For example, if your filter string is “foo,” results will include “foobar,” “foo,” and “food.”

Results for regular expression filters are exact matches.

Resource Objects

Windows computer or application server, such as Exchange Server or SQL Server

Default Schedule

By default, this script runs every hour.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Job Failure Notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the MissingEvent job fails. The default is 25.
Event Log Monitoring	
Event logs to monitor	Indicate the name of the event log you want to monitor, separating multiple names with a comma. For example: `System,Application,Security` The default is `Application`. **NOTE:** If the event log you specify does not exist on the target computer, the Application log is automatically monitored.
Number of hours to scan logs	Set this parameter to control how the script scans the logs in the first interval, after which scanning begins where the previous scan ended. Enter one of the following values: ◆ **-1** - to scan all the existing entries ◆ *N* - to scan entries only for the past *n* hours (8 for the past 8 hours, 50 for the past 50 hours, for example) ◆ **0** - to not scan previous entries; only search from this moment on The default is 0.
Event Log Filters	

Parameter	How to Set It
Use regular expressions for filter criteria?	Select **Yes** to use a regular expression to specify the criteria to look for in each event log you want to monitor. The default is unselected. You can use regular expressions in the *Event [...] filter* parameters, or you can create an external `.txt` file that contains the regular expressions you want to use as filtering criteria.
Enforce case sensitivity in regular expressions?	Select **Yes** to enforce case-sensitivity in all regular expressions used in the filter parameters or in the external `.txt` file. The default is unselected.
Event type for regular expression filtering	Use a regular expression to identify the type of event entries to search for in the logs: Error, Warning, Information, Audit_Success, Audit_Failure, Unclassified.
Path to file containing regular expression filters	Provide the full path to a file containing the regular expression filtering criteria you want to find in each monitored event log. For example: `C:\TEMP\MyFilters.txt`. Format the contents of the `.txt` file as described in "Using an External Filter File" on page 303. **NOTE:** If you specify a filter file, AppManager ignores the *Event [...] filter* parameters, even if the filter file is inaccessible for any reason, but continues to scan the log file specified in the *Event logs to monitor* parameter.
Event Types	
Error	Select **Yes** to search for log entries with an event type of Error. The default is Yes.
Warning	Select **Yes** to search for log entries with an event type of Warning. The default is Yes.
Information	Select **Yes** to search for log entries with an event type of Informational. The default is Yes.
Success Audit	Select **Yes** to search for log entries with an event type of Success Audit. A Success Audit event is an audited security access attempt that succeeds. The default is Yes.
Failure Audit	Select **Yes** to search for log entries with an event type of Failure Audit. A Failure Audit event is an audited security access attempt that fails.The default is Yes.
Unclassified	Some events written to Windows event logs do not have event levels or severities set to event types recognized by Windows Server 2008 and later. This Knowledge Script identifies these entries as unclassified. These entries will not be found by the error, warning, informational, success audit, or failure audit filter criteria. Set to **Yes** to monitor log entries that are unclassified. The default is Yes.
Event source filter	To search for log entries generated by a particular source (such as SQLExecutive, SNMP, or the Service Control Manager), enter a search string or a regular expression. This script will look for matching entries in the event log's Source field. Separate multiple strings with commas.

Parameter	How to Set It
Event category filter	To search for log entries in a particular category (such as Server or Logon), enter a search string or regular expression. This script will look for matching entries in the event log’s Category field. Separate multiple strings with commas.
Event ID filter	To search for log entries with particular event IDs, enter a search string, regular expression, or ID range (for example 100-2000). This script will look for matching entries in the event log’s Event field. Separate multiple IDs and ranges with commas. For example: `1,2,10-15,202`.
Event user filter	To search for log entries associated with a particular user, enter a regular expression or search string, for example, <*domainname*>\<*username*>. This script will look for matching entries in the Event log’s User field. Separate multiple strings with commas.
Event computer filter	To search for log entries generated by a particular computer, enter a search string or regular expression. This script will look for matching entries in the event log’s Computer field. Separate multiple strings with commas.
Event description filter	To search for log entries with a particular detail description or containing keywords in the description, enter a search string or regular expression. This script will look for matching entries in the event log’s Description field. Separate multiple strings with commas.
Event Notification	
Raise event if entries are missing from event logs?	Select **Yes** to raise an event if the selected event logs *do not* contain entries matching your filtering criteria. The default is Yes.
Event severity when entries are missing from event logs	Set the severity level, from 1 to 40, to indicate the importance of an event in which the selected logs do not contain entries matching your filtering criteria. The default is 5.
Data Collection	
Collect data for missing log entries?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the number of log entries found that match the filtering criteria. If no entries match the criteria, data collection returns 0 (zero). The default is unselected.
Separate data by log file type	Select **Yes** to separate event entries from different log files into different datastreams. If disabled, all event entries matching your filtering criteria are placed in the same datastream and the data detail message may include event entries from multiple log sources. For example, if you are monitoring both the System and Application logs, you can enable this parameter so that events in the System log are tracked separately from events in the Application log. The default is Yes.

Using an External Filter File

With the MissingEvent script, you can specify regular expressions for each *Event [...] filter* parameter or maintain your search criteria independent of the Knowledge Script parameters in a separate filter file.

In many cases, specifying an external filter file provides greater flexibility and makes modifying your search criteria more straightforward because you can add almost any number of expressions and you do not need to modify the Knowledge Script properties to pick up your changes.

NOTE: If you specify a filter file, AppManager ignores the *Event [...] filter* parameters, even if the filter file is inaccessible for any reason.

If you want to use a filter file:

- Identify the strings that you want to find a match for, that is, the entries you want to include in your results.
- Create a text file with one regular expression string per line to locate matching strings. Each line in the file consists of a parameter keyword followed by a colon (:), a tab or blank space, and the regular expression. Or the filter file can be written using XML.
- Make sure the file exists on the target computer.
- Type the absolute path to the file on the local computer in the *Path to file containing regular expression filters* parameter and start the job.

Two formats are valid for the filter file: a simple table format to define the strings to include and an XML format that allows you to define more complex include and exclude filtering. For both formats, the parameter name keywords are required, but the field values can be left blank if no filtering is needed.

Select a format appropriate for the complexity of your filtering needs.

Table Format

The table format provides a simple way to create the filter file. Each filtering section in the file begins with `EventStart` and ends with `EventEnd`. If an entry in the event log matches all the criteria you have specified within a filtering section, it is considered a match; no AppManager event is raised. If you have more than one filtering section, a log entry can match either section. Remember, for the MissingEvent Knowledge Scripts, events are raised only when no log entry matches your criteria.

For example, the following table format file provides two filter sections:

```
EventStart
CaseSensitive:n
Log:System
Type:Error|Warning|Information
Source:^SQL*
Category:*
EventID:1[0-9][0-9][0-9]
User:Sam|Joe|Chris
Computer:SFO*
Description:($Error.*)|(.*error.*occurred.$)
EventEnd
EventStart
CaseSensitive:n
Log:Application
Type:Error|Warning|Information
Source:^SQL*
Category:*
EventID:1[0-9][0-9][0-9]
User:Sam|Joe|Chris
Computer:SFO*
Description:($Error.*)|(.*error.*occurred.$)
EventEnd
```

NOTE: If you are only specifying one filter section, do not include the `EventStart` and `EventEnd` lines in the file.

XML Format

The XML format is more sophisticated and more flexible than the table format. The XML format allows you to set both include and exclude filters using the `<Include>` and `<Exclude>` tags and to combine these filter sets to define the search criteria. Each filtering section in the file begins with the `<Events>` tag. A log entry must match all the criteria you specified within a filtering section for it to be considered a match.

For example:

```
<?xml version = "1.0" standalone = "yes"?>
<EventLogConfig Name = "Event Filter" Type = "EVENT_FILTER_CUSTOM" ID = "76">
<Include>
   <Events>
      <Log>Application</Log>
      <Type>INFORMATION|WARNING|ERROR</Type>
      <Source><Net*]></Source>
      <Category>*</Category>
      <EVENTID>2*</EVENTID>
      <User>*</User>
      <Computer>*</Computer>
      <Description><![CDATA[Event.]]></Description>
      <CaseSensitive>y</CaseSensitive>
   </Events>
   <Events>
      <Log>System</Log>
      <Type>Warning</Type>
      <Source>RSVP</Source>
      <Category>*</Category>
      <EVENTID>*</EVENTID>
      <User>*</User>
      <Computer>SHASTA</Computer>
      <Description>RSVP*</Description>
      <CaseSensitive>y</CaseSensitive>
   </Events>
</Include>
</EventLogConfig>
```

NOTE: If a field contains a regular expression that conflicts with XML syntax or includes special characters, you can use `![CDATA[regular_expression]]` to enclose the expression and prevent parsing problems.

8.14 PingMachine

Use this Knowledge Script to check the availability of computers or other machines that reply to ICMP Echo requests. With this script, you can use your managed client Windows computer to check the up/down status of UNIX computers, other Windows computers, and other equipment, such as TCP/IP-based printers. The ICMP Echo request is commonly used by the ping command on UNIX and Windows computers.

This script automatically raises an event if a computer does not respond to the ping command from the computer where this script is running. Note that this script returns event information even if the remote computer is in maintenance mode. In addition, you can choose to raise an event if the ping attempt fails for any other reason.

When configuring an action for this script, configure the Location to initiate the action on the MS (management server) or on a Proxy (to run on a particular managed client computer).

If you configure an action to run on the managed client (MC), when a remotely monitored computer is placed into machine maintenance mode or scheduled maintenance mode, any event conditions detected on the remote computer are ignored but the action is not disabled; in this case, an action is run but there will be no event information on the **Events** tab.

NOTE: This script does not require the AppManager agent to be installed on the remote computers you want to monitor.

Resource Objects

Any Windows server that recognizes the ping command

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
List of computers to check	Specify a list of the computer names or hostnames, separated by commas, to which you want to test communication. For example, to check connectivity to the NetIQ Corporation Web site, type: `www.netiq.com`. You can specify computers that are not currently in the Navigation pane or the TreeView pane.

Full path to file with list of computers	Provide the full path to the file containing a list of the computers to check. The file should contain the hostname or IP address for each computer in one or more lines. Each line can have multiple computer names, separated by commas and with no spaces. Do not include tabs or any other characters other than commas or computer names in this file.
	For example:
	`NYC01,NYC02`
	`SALES01,10.15.221.5,SF001`
	`LABMACH,QATEST`

Number of seconds to wait for ping response	Specify the maximum number of seconds to wait for a response before timing out. The default is 3 seconds.
Number of echo requests to send	Specify the maximum number of times to send the ping request before raising an event. The default is 2 times.
Threshold - Maximum number of consecutive timeouts	Specify the maximum number of consecutive timeouts to allow before raising an event. The default is 1 timeout.
Raise event for any errors during ping?	Select **Yes** to raise an event if an error other than timing out occurs during the ping attempt. The default is Yes.

Parameter	How to Set It
Event severity for ping errors	Set the event severity level, from 1 to 40, to indicate the importance of an event in which an error other than timing out occurs during the ping attempt. The default is 10 (red event indicator).
Data Collection	
Collect data for computer availability?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the target computer responded to the ping ◆ **0** -- the target computer did not respond ◆ **50** -- either the ping failed or the ping returned no output to the results file The default is Yes.
Collect data for response time?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the average response time for the computers to which a ping request has been sent. If a computer is unavailable or a ping error occurs, response time data collection returns 0. Ping response times of less than 1 ms are returned as 1 ms. The default is unselected.

8.15 Report_MachineAvailability

Use this Knowledge Script to generate a report about the availability of computers. This report uses data collected by the MachineDown Knowledge Script, which tests the connection from a single computer to one or more other computers.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select time range	Set a specific or sliding time range for data included in your report. The default is a sliding time of 1 day.

Parameter	How to Set It
Select peak weekday(s)	Select the days of the week to include in your report. The default is seven days: Sunday through Saturday
Data settings	
Hours or percentage on chart	Select whether to illustrate availability by hours or by percentage. The default is Percentage.
Select sorting/display option	Select whether data is sorted, or the method of display: ◆ **No sort**: Data is not sorted. This is the default option. ◆ **Sort**: Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right) ◆ **Top %**: Chart only the top N % of selected data (sorted by default) ◆ **Top N**: Chart only the top N of selected data (sorted by default) ◆ **Bottom %**: Chart only the bottom N % of data (sorted by default) ◆ **Bottom N**: Chart only the bottom N of selected data (sorted by default)
Percentage/count for top/bottom	Enter a number for either the percentage or count defined in the previous parameter (for example, Top 10%, or Top 10). The default is 25.
Truncate top/bottom?	If set to **yes**, the data table shows only the top or bottom N or % (for example, only the top 10%). Otherwise, the table shows all data. The default is yes.
Report settings	
Include parameter help card?	Select **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Select an option to include datastream values in the report: ◆ **Table: Select this option to include a table of datastream values in the report.** ◆ **Chart:** Select this option to include a chart of datastream values in the report. ◆ **Both:** Select this option to include both table and chart of datastream values in the report.
Select chart style	Define the graphic properties of the charts in your report. The default chart style is Pie.
Select output folder	Set parameters for the output folder. The default folder name is General_MachineAvailability.
Add job ID to output folder name?	Select **yes** to add the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is no.
Select properties	Set report properties. The default report name is General Machine Availability.

Parameter	How to Set It
Add time stamp to title	Select **yes** to add a time stamp to the title of the report, making each title unique. The time stamp shows the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is no.
Event notification	
Event for report success?	Select **yes** to raise an event when the report is successfully generated. The default is yes.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the report is successfully generated. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the generated report contains no data. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the report is not generated. The default is 5 (red level indicator).

8.16 Report_PingMachine

Use this Knowledge Script to generate a report about the availability of computers or other machines that reply to ICMP Echo requests. This report uses data collected by the PingMachine Knowledge Script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select time range	Set a specific or sliding time range for data included in your report. The default is 1 day sliding time ending in 24 hours.
Select peak weekday(s)	Select the days of the week to include in your report. The default is seven days: Sunday through Saturday.

Parameter	How to Set It
Data settings	
Hours or percentage on chart	Select whether to illustrate availability by hours or by percentage. The default is Percentage.
Select sorting/display option	Select whether data is sorted, or the method of display: ◆ **No sort**: Data is not sorted ◆ **Sort**: Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right) ◆ **Top %**: Chart only the top N % of selected data (sorted by default) ◆ **Top N**: Chart only the top N of selected data (sorted by default) ◆ **Bottom %**: Chart only the bottom N % of data (sorted by default) ◆ **Bottom N**: Chart only the bottom N of selected data (sorted by default) The default is No Sort.
Percentage/count for top/bottom	Specify a value for the percentage or count defined in the *Select sorting/display option* parameter (for example, Top 10%, or Top 10). The default is 25.
Truncate top/bottom?	If set to **yes**, then the data table shows only the top or bottom N or % (for example, only the top 10%). Otherwise, the table shows all data. The default is yes.
Report settings	
Include parameter help card?	Select **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Select an option to include datastream values in the report: ◆ **Table: Select this option to include a table of datastream values in the report.** ◆ **Chart:** Select this option to include a chart of datastream values in the report. ◆ **Both:** Select this option to include both table and chart of datastream values in the report.
Select chart style	Define the graphic properties of the charts in your report. The default chart style is Pie.
Select output folder	Set parameters for the output folder. The default folder prefix is General_PingMachine.
Add job ID to output folder name?	Select **yes** to add the job ID to the name of the output folder. By default, the job ID is not included. A job ID helps you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Set report properties. The default title for your report is General Ping Machine Availability.

Parameter	How to Set It
Add time stamp to title	Select **yes** to add a time stamp to the title of the report, making each title unique. The time stamp shows the date and time the report was generated. The default is no. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output.
Event notification	
Event for report success?	Select **yes** to raise an event when the report is successfully generated. The default is yes.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the report is successfully generated. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the generated report contains no data. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the report is not generated. The default is 5 (red level indicator).

8.17 Report_ServiceChange

Use this Knowledge Script to generate a report about changes to the status and start-type of discovered services. This report uses data collected by the ServiceChange Knowledge Script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select time range	Set a specific or sliding time range for data included in your report. The default is 1 day sliding time ending in 24 hours.
Select peak weekday(s)	Select the days of the week to include in your report. The default is seven days: Sunday through Saturday.
Data settings	

Parameter	How to Set It
Hours or percentage on chart	Select whether to illustrate availability by hours or by percentage. The default is Percentage.
Select sorting/display option	Select whether data is sorted, or the method of display: ◆ **No sort**: Data is not sorted ◆ **Sort**: Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right) ◆ **Top %**: Chart only the top N % of selected data (sorted by default) ◆ **Top N**: Chart only the top N of selected data (sorted by default) ◆ **Bottom %**: Chart only the bottom N % of data (sorted by default) ◆ **Bottom N**: Chart only the bottom N of selected data (sorted by default) The default is No Sort.
Percentage/count for top/bottom	Specify a value for the percentage or count defined in the previous parameter (for example, Top 10%, or Top 10). The default is 25.
Truncate top/bottom?	If set to **yes**, the data table shows only the top or bottom N or % (for example, only the top 10%). Otherwise, the table shows all data. The default is no.
Report settings	
Include parameter help card?	Select **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Select an option to include datastream values in the report: ◆ **Table: Select this option to include a table of datastream values in the report.** ◆ **Chart:** Select this option to include a chart of datastream values in the report. ◆ **Both:** Select this option to include both table and chart of datastream values in the report. The default is Table.
Select chart style	Define the graphic properties of the charts in your report. The default chart style is Pie.
Select output folder	Set parameters for the output folder. The default report prefix is General_ServiceChange.
Add job ID to output folder name?	Select **yes** to add the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is no.
Select properties	Set report properties. The default report title is General Service Change.

Parameter	How to Set It
Add time stamp to title	Select **yes** to add a time stamp to the title of the report, making each title unique. The time stamp shows the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is no.
Event notification	
Event for report success?	Select **yes** to raise an event when the report is successfully generated. The default is yes.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the report is successfully generated. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the generated report contains no data. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the report is not generated. The default is 5 (red level indicator).

8.18 Report_ServiceDown

Use this Knowledge Script to generate a report about the up/down status of discovered services. This report uses data collected by the ServiceDown Knowledge Script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run Once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select time range	Set a specific or sliding time range for data included in your report. The default is 1 day sliding time ending in 24 hours.
Select peak weekday(s)	Select the days of the week to include in your report. The default is seven days: Sunday through Saturday.
Data settings	

Parameter	How to Set It
Hours or percentage on chart	Select whether to illustrate availability by **Hours** or by **Percentage**. The default is Percentage.
Select sorting/display option	Select whether data is sorted, and the method of display: ◆ **No sort**: Data is not sorted ◆ **Sort**: Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right) ◆ **Top %**: Chart only the top N % of selected data (sorted by default) ◆ **Top N**: Chart only the top N of selected data (sorted by default) ◆ **Bottom %**: Chart only the bottom N % of data (sorted by default) ◆ **Bottom N**: Chart only the bottom N of selected data (sorted by default)
Percentage/count for top/bottom	Enter a number for either the percent or count defined in the previous parameter (for example, Top 10%, or Top 10). The default is 25.
Truncate top/bottom?	If set to **yes**, the data table shows only the top or bottom N or % (for example, only the top 10%). If set to no, the table shows all data. The default is yes.
Report settings	
Include parameter help card?	Select **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Select an option to include datastream values in the report: ◆ **Table: Select this option to include a table of datastream values in the report.** ◆ **Chart:** Select this option to include a chart of datastream values in the report. ◆ **Both:** Select this option to include both table and chart of datastream values in the report.
Select chart style	Define the graphic properties of the charts in your report. The default chart style is Pie.
Select output folder	Set parameters for the output folder. The default report prefix is General_ServiceDown.
Add job ID to output folder name?	Select **yes** to add the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is no.
Select properties	Set report properties. The default report title is General Service Down.
Add time stamp to title	Select **yes** to add a time stamp to the title of the report, making each title unique. The time stamp shows the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is no.
Event notification	

Parameter	How to Set It
Event for report success?	Select **yes** to raise an event when the report is successfully generated. The default is yes.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the report is successfully generated. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the generated report contains no data. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the report is not generated. The default is 5 (red level indicator).

8.19 Report_ServiceHung

Use this Knowledge Script to generate a report about discovered services in the Start-Pending, Stop-Pending, Continue-Pending, or Pause-Pending state. If a service is detected in one of these states for a specified number of intervals, it is considered hung. The number of intervals is specified in the Knowledge Script that collects data for this report.

This report uses data collected by the ServiceHung Knowledge Script.

Resource Object

Report agent

Default Schedule

The default schedule for this script is **Run Once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Data source	
Select computer(s)	Select the computers whose data you want to include in your report.
Select time range	Set a specific or sliding time range for data included in your report. The default is a sliding time of 1 day.
Select peak weekday(s)	Select the days of the week to include in your report. The default is seven days: Sunday through Saturday.
Data settings	
Hours or percentage on chart	Select whether to illustrate availability by **Hours or** by **Percentage**. The default is Percentage.

Parameter	How to Set It
Select sorting/display option	Select whether data is sorted, and the method of display: ◆ **No sort**: Data is not sorted ◆ **Sort**: Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right) ◆ **Top %**: Chart only the top N % of selected data (sorted by default) ◆ **Top N**: Chart only the top N of selected data (sorted by default) ◆ **Bottom %**: Chart only the bottom N % of data (sorted by default) ◆ **Bottom N**: Chart only the bottom N of selected data (sorted by default)
Percentage/count for top/ bottom	Enter a number for either the percent or count defined in the previous parameter (for example, Top 10%, or Top 10). The default is 25.
Truncate top/bottom?	If set to **yes**, the data table shows only the top or bottom N or % (for example, only the top 10%). If set to no, the table shows all data. The default is yes.
Report settings	
Include parameter help card?	Select **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Select an option to include datastream values in the report: ◆ **Table: Select this option to include a table of datastream values in the report.** ◆ **Chart:** Select this option to include a chart of datastream values in the report. ◆ **Both:** Select this option to include both table and chart of datastream values in the report.
Select chart style	Define the graphic properties of the charts in your report. The default chart style is Pie.
Select output folder	Set parameters for the output folder. The default report prefix is General_ServiceHung.
Add job ID to output folder name?	Select **yes** to add the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is no.
Select properties	Set report properties. The default report title is General Service Hung.
Add time stamp to title	Select **yes** to add a time stamp to the title of the report, making each title unique. The time stamp shows the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is no.
Event notification	
Event for report success?	Select **yes** to raise an event when the report is successfully generated. The default is yes.

Parameter	How to Set It
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the report is successfully generated. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the generated report contains no data. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the report is not generated. The default is 5 (red level indicator).

8.20 ServiceChange

Use this Knowledge Script to detect any changes to the status and start type of a discovered service. You can run this script for almost any service, including SQL Services, Exchange Services, and IIS Services. This script raises an event if the status (running, stopped, pending, and so on) or startup type (manual, automatic, disabled) of any service has been changed.

Resource Objects

Windows computer or Windows application service

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Collect data for service changes?	Set to **y** to collect data for charts and reports. If enabled, data collection returns one of the following: ◆ **100** -- the service is unchanged ◆ **0** -- the service is not running or has been changed. The default is n.
Event severity when service start type changes	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service’s start type has changed. The default is 10 (red event indicator).
Severity - service status changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service’s status has changed. The default is 5 (red event indicator).
Event severity when information retrieval fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the script failed to retrieve service information. The default is 18 (yellow event indicator).

8.21 ServiceDown

Use this Knowledge Script to detect whether a discovered service is running. You can run this script for most services, including SQL Server services, Exchange Server services, and IIS services. Use the NT_ServiceDown Knowledge Script to check other services, such as `WinLogon` or `NetIQms`, which are not included in the Navigation pane or the TreeView.

This script raises an event if any monitored service is not running and can automatically restart the down service.

Resource Objects

Windows computer or Windows application service

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if service is stopped?	Select **Yes** to raise an event if the status of a monitored service is “stopped.” The default is Yes.
Event severity when service is stopped	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the status of a monitored service is “stopped.” The default is 18 (yellow event indicator).
Event severity when service cannot be started	Set the event severity level, from 1 to 40, to indicate the importance of an event in which this script is unable to start a stopped service. The default is 5 (red event indicator).
Raise event if service is started?	Select **Yes** to raise an event if this script successfully starts a stopped service. The default is unselected.
Event severity when service is started	Set the event severity level, from 1 to 40, to indicate the importance of an event in which this script successfully starts a stopped service. The default is 25 (blue event indicator).
Raise event if service is missing?	Select **Yes** to raise an event if a service that was found during a previous discover cannot be found. The default is unselected.
Event severity when service is missing	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a discovered service is missing. The default is 8 (red event indicator).
Raise event if service is disabled?	Select **Yes** to raise an event if a service is stopped and is disabled. The default is unselected.
Event severity if service is disabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is stopped and is disabled. The default is 12 (yellow event indicator).

Parameter	How to Set It
Raise event if service is shut down normally?	Select **Yes** to raise an event if a service was stopped as a result of a stop request, such as a stop request issued by a user or as a result of another service being stopped. The default is Yes. This parameter takes effect only when the *Only restart service if shut down normally?* parameter is disabled.
Event severity when service shut down normally	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service was stopped as a result of a stop request. The default is 30 (blue event indicator).
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the ServiceDown job fails. The default is 5 (red event indicator). An event is raised for circumstances under which the script fails to run properly.
Data Collection	
Collect data for service status?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns one datastream for each service you are monitoring, and, for each monitored service, one datastream covering all dependent services. For monitored services, the data detail message includes the service name, the start type, and the status. If a monitored service is up, a value of 100 is returned. If the service is down, a value of 0 is returned. If the service is down and the Knowledge Script successfully restarts the service, a value of 50 is returned. For services depending on the monitored service, the data detail message includes service names, and the start type and status of each service. If all dependent services are up, a value of 100 is returned; if any dependent service is down, a value of 0 is returned. The default is unselected.
Monitoring	
Start stopped services?	Select **Yes** to start a service that is not running. The default is Yes.
Number of seconds to wait for service start	Specify the number of seconds this script should attempt to start a service before timing out. The default is 30 seconds. **NOTE:** If the service fails to start within the timeout period, an event is raised if the *Raise event if service is down?* parameter is set to Yes. The severity of the event is determined by the *Event severity when service cannot be started* parameter.
Start dependent services?	Select **Yes** to start any service that depends on other services started by this script. For example, if this script starts the `MSSQLSERVER` service, it will also start the dependent `SQLSERVERAGENT` service. The default is Yes.
Restart service if shut down normally?	Select **Yes** to restart a service that was stopped as a result of a stop request, such as a stop request issued by a user or as a result of another service being stopped. The default is Yes.

8.22 ServiceHung

Use this Knowledge Script to detect whether a discovered service is hung. You can run this script for most services, including SQL Services, Exchange Services, and IIS Services. A service is considered hung if it is in a Start-Pending, Stop-Pending, Continue-Pending or Pause-Pending state for a specified number of consecutive intervals. This script raises an event if a hung service is detected, and can stop or restart the hung service.

Resource Objects

Any discovered Windows computer or Windows application service

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Number of consecutive iterations before service is hung	Specify the number of consecutive job iterations a service can be in a Start-Pending, Stop-Pending, Continue-Pending or Pause-Pending state before it is considered hung. The default is 2 consecutive iterations.
Collect data for service status?	Set to **y** to collect data for charts and reports. If enabled, data collection returns one of the following: ◆ **100** -- service is up ◆ **0** -- service is hung The default is n.
Stop the hung service?	Set to **y** to automatically stop hung services. The default is y.
Start hung service after it is stopped?	Set to **y** to automatically start the service after stopping it. The default is y.
Event severity when start fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which this script fails to start a hung service that had been stopped. The default is 5 (red event indicator).
Event severity when start succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which this script successfully starts a hung service that had been stopped. The default is 25 (blue event indicator).
Event severity when “start hung service” is disabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the a hung service is detected and the *Start hung service after it is stopped?* parameter is set to n. The default is 18 (yellow event indicator).
Event severity when status retrieval fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which this script cannot determine the status of a service. The default is 10 (red event indicator).

Parameter	How to Set It
Event severity when service stop fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which this script fails to stop a hung service. The default is 8 (red event indicator).

8.23 ShortEventLog

Use this Knowledge Script to track Windows event log entries that match filtering criteria you specify. This script works on an incremental basis (it does not fully rescan the event log each time it runs), and all event log entries that match the filtering criteria are returned in the event or data point detail message.

This script works in the same fashion as the EventLog Knowledge Script, but removes the header information and returns only the description of the event.

NOTE: Only the most recent batch of events can be viewed in the data point detail message. For example, you might set this script to scan all previous entries in the event log and list ten matching entries in each event detail message. When the job runs, 30 entries are found that match your filtering criteria. In this case, the script creates three child events for the interval. Each child event contains ten entries: the oldest matching entries in one child event batch, the second oldest in Batch 2, and the most recent in Batch 3. If this job is collecting data, and you view the data detail message for the interval, only the entries from the third child event (Batch 3) are displayed.

Resource Objects

Windows computer or application server such as Exchange Server or SQL Server

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if log entries match criteria?	Set to **y** to raise an event when log entries match your filtering criteria. The default is y.
Collect data for log entries that match criteria?	Set to **y** to collect data for charts and reports. When enabled, data collection returns detail about log entries that match your filtering criteria. The default is n.

Parameter	How to Set It
Separate data by log file type?	Set to **y** to separate event entries from different log files into different datastreams. If set to n, all event entries matching your filtering criteria are placed in the same datastream and the data detail message may include event entries from multiple log sources. For example, if you are monitoring both the System and Application logs, you can enable this parameter so that events in the System log are tracked separately from events in the Application log. The default is n.
Log files to filter (Application, Security, System)	Specify the event log you want to monitor. You can specify multiple event logs, separated by commas. For example: `System,Application,Security` The default is `Application`.
Log scanning for first interval	Set this parameter to control how the script scans the logs at the first interval, after which scanning begins where the previous scan ended. Enter one of the following values: ◆ **-1** - to scan all the existing entries ◆ *N* - to scan entries only for the past *n* hours (8 for the past 8 hours, 50 for the past 50 hours, for example) ◆ **0** - to not scan previous entries; only search from this moment on. The default is 0.
Monitor error events?	Set to **y** to monitor error event entries. The default is y.
Monitor warning events?	Set to **y** to monitor warning event entries. The default is y.
Monitor information events?	Set to **y** to monitor information event entries. The default is y.
Monitor success audits?	Set to **y** to monitor success audit event entries. Success audits are successful security access attempts that are audited. The default is y. **NOTE:** This parameter applies to WinOS2003 only. With Windows Vista or Windows Server 2008 or higher, you monitor success audits using keywords.
Monitor failure audits?	Set to **y** to monitor failure audit events entries. Failure audits are failed security access attempts that are audited. The default is y. **NOTE:** This parameter applies to WinOS2003 only. With Windows Vista or Windows Server 2008 or higher, you monitor failure audits using keywords.
Event source filter	To filter for events generated by a particular source (such as SQLExecutive, SNMP, or the Service Control Manager), enter a search string. This script will look for matching entries in the Event Log's **Source** field. Separate multiple strings with commas. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you specify only include criteria, the colon is not necessary

Parameter	How to Set It
Event category filter	To filter for events in a particular category (such as Server or Logon), enter a search string. This script will look for matching entries in the Event Log's **Category** field. Separate multiple strings with commas. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you specify only include criteria, the colon is not necessary.
Event ID filter	To filter for particular event IDs, enter a search string or ID range, for example 100-2000. This script will look for matching entries in the Event Log's **Event** field. Separate multiple IDs and ranges with commas. For example: `1,2,10-15,202`. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you specify only include criteria, the colon is not necessary.
Event user filter	To filter for events associated with a particular user, enter a search string, for example, `<domain name>\<user name>`) This script will look for matching entries in the Event Log's **User** field. Separate multiple strings with commas. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you specify only include criteria, the colon is not necessary
Computer filter	To filter for events generated by a particular computer, enter a search string. This script will look for matching entries in the Event Log's **Computer** field. Separate multiple strings with commas. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you specify only include criteria, the colon is not necessary.
Event description filter	To filter for events with a particular detail description or containing keywords in the description, enter a search string. This script will look for matching entries in the Event Log's **Description** field. Separate multiple strings with commas. The search string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you specify only include criteria, the colon is not necessary
Maximum number of entries per event report	Specify the maximum number of entries to be recorded in each event's detail message. If this script finds more entries from the log than can be put into one event message, it will return multiple events to report all the outstanding entries in the log. The default is 30 entries. If this script encounters one or more very large events in the Windows Event log, this Knowledge Script may error out and generate an event message "`Out of string space`." If this occurs, you can usually work around the problem by adjusting this parameter to a smaller value.
Event severity when event log entries match criteria	Set the event severity level, from 1 to 40, to indicate the importance of an event in which log entries matched your search criteria. The default is 8 (red event indicator). **Tip** You can adjust the severity based on which log or type of event you are checking for.

Examples of How this Script Is Used

You can customize this script in many ways based on your requirements. For example, for general system events, you can set the following options when detecting security failures:

Properties and Parameters	How You Might Set Them
Schedule interval	10 minutes
Raise event if log entries match criteria?	y
Log files to filter	Security
Monitor failure audits?	y
Event severity when event log entries match criteria	2
Action	MapiMail

With this scenario, on the **Schedule** tab in the Knowledge Script Properties dialog box, set the interval to *Once very 10 minutes* because you want a short window for checking for this type of problem.

On the Values tab, enable the *Raise event if log entries match criteria?* parameter. Set *Log files to filter* to **Security** and set *Monitor failure audits?* to **y**. Set the *Event severity level* parameter to **2**, indicating this is a very serious event that you want to be highly visible. Leave the other filtering options blank.

On the **Action** tab, indicate that you want an e-mail sent if an event is raised. With these settings, AppManager will regularly check for security failures and will notify you, or whoever you designate, through e-mail if any security failure events are detected.

Another example of how to use this script to detect all problems with your SQL Server involves setting up the Knowledge Script job as follows:

Properties and Parameters	How You Might Set Them
Schedule interval	30 minutes
Raise event if log entries match criteria?	y
Log files to filter	Application
Monitor error events?	Error
Event source filter	MSSQLServer
Event severity when event log entries match criteria	8
Action	MapiMail

Another way you can use this Knowledge Script is to collect data and graph a trend chart from your System event log:

Properties and Parameters	How You Might Set Them
Schedule interval	1 hour
Collect data for log entries that match criteria?	y
Log files to filter	System
All other filters	not set
Action	Null

If you select the data collection option, this script returns the number of matched entries as the primary data point to be graphed. The first batch of filtered results can be viewed in the detail data message when you double-click a data point. Additional matching entries may be included in the graph. The peaks and valleys in the graph indicate a large number of events (something unusual) or low event activity (quiet and all “OK”).

8.24 SNMPGet

Use this Knowledge Script to monitor SNMP activity for the device or computer you specify. This script performs an SNMP v1 `Get` or `GetNext` against the selected SNMP agent, allowing you to check SNMP MIB (management information base) variable values. The value returned can be compared to the thresholds you set or a text string. This script requires the Microsoft SNMP Service to be running.

NOTE: When setting the parameters for this script, choose whether the jobs should perform a threshold comparison, equality check, or string matching. These operations are mutually exclusive operations.

Resource Objects

Any Windows server or CIM server with an SNMP agent installed and running

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if MIB variable matches string or numeric value, or exceeds or falls below threshold?	Set to **y** to raise an event if a MIB value falls below or exceeds the threshold, or if a value matches the parameters you set for *Equality check?* or *String match*. The default is y.

Parameter	How to Set It
Collect data for MIB variable?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the value of the MIB variable for graphing. If the MIB variable you specify is of an octet string type, the value is displayed in the graph data detail message. The default is y.
MIB object identifier	Specify a MIB object identifier in OID notation (for example, `.1.2.3.456.78`) or ODE notation (for example, `system.sysUpTime.0`). The default is `system.sysName.0`. OID notation must include the dot (.) at the beginning of the identifier. ODE notation must be case-sensitive. You can use the ODE if the `mib.bin` file has been compiled on the agent computer in the `%windir%/system32` directory. For information about compiling the `mib.bin`, see the Windows Resource Kit.
SNMP community string	Provide the SNMP community string for the device or computer on which you want to monitor SNMP activity. Leave this parameter blank to use the SNMP community name entered in the AppManager Security Manager. The default is `public`.
SNMP agent	Specify the hostname or IP address of the device or computer on which you want to monitor SNMP activity. If you do not specify an SNMP agent, the local client computer is assumed.
Threshold - Maximum MIB variable value	Specify the maximum value the MIB variable can attain before an event is raised. The default is 600000.
Threshold - Minimum MIB variable value	Specify the minimum value the MIB variable must maintain to prevent an event from being raised. The default is 300000.
Check for equality to numeric MIB variable	◆ Set to **e** to compare the MIB variable's value to a specific value (set in the *Numeric MIB variable value* parameter) and raise an event when the values are equal. ◆ Set to **n** to compare the MIB variable's value to a specific value (set in the *Numeric MIB variable value* parameter) and raise an event when the values are not equal. ◆ Set to **s** to skip testing for equality. This parameter is applicable for numeric MIB variables such as `INTEGER`, `GAUGE`, or `COUNTER`. The default is s.
Numeric MIB variable value	Specify the value that you want to compare with the returned MIB variable value. ◆ If *Check for equality to numeric MIB value?* is set to **e**, an event is raised when the MIB variable equals the value you specify in this parameter. ◆ If *Check for equality to numeric MIB value?* is set to **n**, an event is raised when the values are not equal. The default is 0.
Text string or IP address MIB variable value	Specify the text string or IP address that you want to compare with the returned MIB variable value. This parameter is applicable only when the MIB type is `OCTETSTRING` or `IPADDRESS`. The MIB variable value is compared to this string, and an event is raised if they are equal.
Enforce case-sensitive string match?	Set to **y** to enable this script to match case when checking for a match to the string entered for the *String match* parameter. The default is n.

Parameter	How to Set It
Event severity when MIB variable matches string or numeric value, or exceeds or falls below threshold?	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a MIB variable value exceeds or falls below the threshold, or if a value matches the parameters you set for *Equality check?* or *String match*. The default is 5 (red event indicator).
Select operation: Get or GetNext	Specify whether to perform SNMP `Get` or `GetNext`. The default is `Get`.
Number of times to perform the operation	Specify the number of times this script should try to perform the `Get` operation before returning an error. The default is 3 times.
Number of seconds to wait for operation to complete	Specify the number of seconds this script should wait for the `Get` or `GetNext` operation to complete before timing out and returning an error. The default is 5 seconds.
Event message text	Provide the text to display in the event detail message. If you do not enter a message, a default message consisting of the MIB variable and value is used.

8.25 WMICounter

Use this Knowledge Script to monitor any Windows Management Instrumentation (WMI) object property. You can run this script on any WMI server and monitor any property available for an object. This script raises an event if the value of the property you select exceeds or falls below the threshold you set. You can also specify a consecutive number of times that the threshold must be exceeded before an event is raised.

Use this script to yield performance information for the WMI properties you are monitoring. Use the property data to start corrective actions when thresholds are exceeded, to generate complex and sophisticated graphs, and to provide historical information for reporting, trend analysis, and capacity planning.

NOTE: An event is raised only if the property value exceeds or falls below the thresholds you set. If a counter does not exist on the monitored computer, the job terminates with an error.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Collect data for WMI object property values?	Set to **y** to collect data for charts and reports. When enabled, data collection returns the object property values that exceeded or fell below the threshold you set. The default is n.
Raise event when object property value exceeds threshold?	Set to **y** to raise an event if the object property value exceeds the maximum threshold you set. The default is y.
Threshold - Maximum value for object property	Specify the maximum value the object property can attain before an event is raised. The default is 100.
Raise event when object property value falls below threshold?	Set to **y** to raise an event if the object property value falls below the minimum threshold you set. The default is y.
Threshold - Minimum value for object property	Specify the minimum value the object property must maintain to prevent an event from being raised. The default is 10.
WMI object property to monitor	Specify the namespace, class, and property to monitor. For more information, see "Selecting a Property to Monitor Using the WMI Browser" on page 329 and "Entering Property Names Without Browsing" on page 329. For details about WMI classes and objects, see the WMI Object Browser available with the WMI platform SDK.
Number of consecutive times to exceed or fall below threshold	Specify the number of consecutive times a monitored object property value should exceed or fall below the threshold before an event is raised. The default is 1 time.
Raise event if value cannot be retrieved?	Set to **y** to raise an event if this script cannot retrieve the object property value. The default is y.
Event severity when object property value exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the object property value exceeds the threshold you set. The default is 8 (red event indicator).
Event severity when object property value falls below threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the object property value falls below the threshold you set. The default is 8 (red event indicator).
Event severity when property/ instance not found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the specified property or instance does not exist. The default is 8 (red event indicator).

Selecting a Property to Monitor Using the WMI Browser

To select the property you want to monitor, click **Browse [...]** in the *WMI object property to monitor* parameter to launch the Windows Management Instrumentation Browser dialog box. Specify the target computer, the namespace and class in which the property resides, the instance of the property, such as the name of a service or particular log file, and the name of the specific property you want to monitor. The term “schema” in this dialog box refers to properties.

To select a property to monitor using the WMI browser:

1. Click **Browse [...]** and select the target **Computer**.
2. From the **Classes In** list, specify the namespace that contains the class in which the object and property (schema) are located.
3. Click **Enumerate** to view the classes and objects in the namespace you specified.
4. Select the **Instance** of the class you want to monitor.
5. From the **Schema** list, select the name of the property you want to monitor.
6. Click **OK**.

Entering Property Names Without Browsing

To type property names rather than use the Windows Management Instrumentation Browser, enter the name in the *WMI object’s property to monitor* parameter using the following format:

```
<namespace>:<class.instance="unique identifier">:<property>
```

where `instance` is the instance category (such as name or log file), and `"unique identifier"` is the name of the specific instance you want to monitor.

Using the example from the previous section, to monitor the state of the `NetIQmc` service, type this information in the *WMI object’s property to monitor* parameter as follows:

```
//./root/cimv2:Win32_Service.Name=""NetIQmc"":State
```

where

- `//./root/cimv2` is the namespace
- `Win32_Service` is the class
- `Name` is the instance
- `"NetIQmc"` is the unique identifier that specifies the specific service you want to monitor
- `State` is the property of the instance that you want to monitor

9 NetServices Knowledge Scripts

The NetServices category provides Knowledge Scripts for monitoring network services with AppManager, such as monitoring the availability and use of the Windows Internet Name Service (WINS) server. This Knowledge Script category is added when you discover Windows.

NOTE: You can use the NetServices category to monitor Windows Server 2008 (or later) services.

From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. In the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
DHCPHealthCheck	Checks the availability and SNMP MIB counter of the DHCP server.
DHCPLeases	Monitors the percentage of DHCP address leases that are being used.
DNSHealthCheck	Monitors availability, CPU usage, memory usage, and name resolution of the DNS service.
DNSSync	Checks connectivity between two DNS servers.
RASConnections	Monitors the average number of connections to the remote access server (RAS).
RASErrors	Monitors the total number of remote access server (RAS) errors in an interval.
RASHealthCheck	Checks the availability and CPU usage of the RAS server.
RASStat	Monitors the throughput of the RAS server.
WINSConflict	Monitors conflict activity on the WINS server.
WINSFailure	Reports the number of failures per second on the WINS server.
WINSHealthCheck	Checks availability and CPU usage of the WINS server.
WINSQueries	Monitors query activity on the WINS server.
WINSReplication	Monitors replication activity on the WINS server.
WINSStat	Monitors the total registrations, renewals, and releases on the WINS server.

9.1 DHCPHealthCheck

Use this Knowledge Script to check the status of the Dynamic Host Configuration Protocol (DHCP) service and the SNMP MIB variable value for a DHCP object identifier (OID). If the DHCP service is not running or a valve cannot be retrieved for the DHCP OID, an event is raised. If the DHCP service is not running, it can be automatically restarted.

Prerequisite

This script requires the Microsoft SNMP service to be running.

Resource Object

DHCP service object

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event when DHCP service is down or MIB counter can't be retrieved?	Set to **y** to raise an event when the DHCP service is down or the MIB counter cannot be retrieved. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the DHCP service is running, or ◆ **0** -- the service is not running. The default is n.
Auto-start service?	Set to **y** to automatically restart the DHCP service. The default is y.
Event severity level for service down; restart failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the DHCP service is down and AppManager cannot restart the service. The default is 5 (red event indicator).
Event severity level for service down; restart successful	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the DHCP service is down and AppManager successfully restarted the service. The default is 25 (blue event indicator).
Event severity level for service down; don't restart	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the DHCP service is down and the *Auto-start service?* parameter is set to n. The default is 18 (yellow event indicator).
Event severity level for SNMP failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which an SNMP failure has occurred. The default is 18 (yellow event indicator).
OID of any DHCP MIB counter	Provide the object identifier of any DHCP MIB counter you want checked. The default is .1.3.6.1.4.1.311.1.3.1.2.0.
Community	Specify the SNMP community string. The default is either the community string entered in AppManager Security Manager or *public* if no community string has been entered. The default is *public*.

9.2 DHCPLeases

Use this Knowledge Script to monitor the percentage of DHCP address leases that are being used. This script raises an event if the percentage of addresses used exceeds the threshold you set or the number of available addresses falls below the threshold you set. This script can monitor addresses for each DHCP scope individually or for the entire DHCP server.

The concept of a DHCP address lease is one in which a client computer does not retain a permanent DHCP address. With a DHCP lease, a client computer communicates with a DHCP server on reboot to begin or confirm the lease of an address.

Prerequisite

This script requires the Microsoft SNMP service to be running.

Resource Object

DHCP service object

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event for total address usage?	Set to **y** to raise events when the percentage of addresses used or the number of available addresses exceeds the threshold for the entire DHCP server. The default is y.
Event for scope address usage?	Set to **y** to raise events when the percentage of addresses used or the number of available addresses exceeds the appropriate threshold for any DHCP scope. The default is y.
Collect data for total address usage?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the total number of addresses in use. The default is n.
Collect data for scope address usage?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of addresses in use in a DHCP scope. The default is n.
Maximum threshold for the percentage of total addresses in use	Specify the maximum percentage of addresses that can be in use at one time for the entire DHCP server (total addresses available) before an event is raised. The default is 95%.
Maximum threshold for percentage of addresses in use in a scope	Specify the maximum percentage of addresses that can be in use in a DHCP scope before an event is raised. The default is 80%. A DHCP scope is the range of IP addresses that the DHCP server can assign to clients that are on one subnet.
Minimum threshold for total number of addresses available	Specify the minimum number of addresses that must be available for the DHCP server to prevent an event from being raised. The default is 10.

Parameter	How to Set It
Minimum threshold for number of addresses available in a scope	Specify the minimum number of addresses that must be available in a DHCP scope to prevent an event from being raised. The default is 5.
Event severity level for total usage high or availability low	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the percentage of addresses in use exceeds the threshold or the number of available addresses falls below the threshold. The default is 10 (red event indicator).
Event severity level for scope usage high or availability low	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the percentage of addresses in use in a scope exceeds the threshold or the number of available addresses in a scope falls below the threshold. The default is 15 (yellow event indicator).
Community	Specify the SNMP community string. The default is either the community name entered in AppManager Security Manager or *public* if no community name has been entered. The default is public.

9.3 DNSHealthCheck

Use this Knowledge Script to monitor the availability, CPU usage, memory usage, and name resolution of the Domain Name System (DNS) service. By default, this script attempts to restart the DNS service if the service is not running.

Resource Object

DNS service object

Default Schedule

The default interval for this script is **Every five minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event when DNS service is down?	Set to **y** to raise an event when the DNS service is down. The default is y.
Event when DNS name resolution fails?	Set to **y** to raise an event when name resolution fails. If set to y, the script attempts to resolve the specified hostname to the specified IP address. If the name resolution fails, an event is raised. If set to n, the script does not perform a name resolution lookup, and the *Hostname for name resolution* and *Host IP address that should be returned* parameters are ignored. The default is y.
Event when DNS CPU usage is over threshold?	Set to **y** to raise an event when the percentage of CPU consumed by the DNS service exceeds the threshold. The default is y.

Parameter	How to Set It
Event when DNS memory usage is over threshold?	Set to **y** to raise an event when the amount of memory consumed by the DNS service exceeds the threshold. The default is y.
Collect data for DNS service up/down?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the DNS service is running, or ◆ **0** -- the service is not running. The default is n.
Collect data for DNS CPU usage?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the percentage of CPU consumed by the DNS service. The default is n.
Collect data for DNS memory usage?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the amount of memory, in kilobytes (KB), consumed by the DNS service. The default is n.
Auto-start service?	Set to **y** to automatically restart the DNS service if it is down. The default is y.
Event severity level for service down; restart failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the DNS service is down and the script cannot restart the service. The default is 5 (red event indicator).
Event severity level for service down; restart succeeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the DNS service is down and the script successfully restarted the service. The default is 25 (blue event indicator).
Event severity level for service down; don’t restart	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the DNS service is down and the *Auto-start service?* parameter is set to n. The default is 18 (yellow event indicator).
Event severity level for NS lookup failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the DNS name resolution function fails. The default is 8 (red event indicator).
Event severity level for CPU usage exceeded threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which CPU usage exceeds the threshold. The default is 8 (red event indicator).
Event severity level for memory usage exceeded threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which memory usage exceeds the threshold. The default is 8 (red event indicator).
Event severity level for external command failed to execute	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a system error prevents the execution of the DNS service. The default is 8 (red event indicator).
DNS process % CPU maximum threshold	Specify the maximum percentage of CPU resources that can be consumed by the DNS process before an event is raised. The default is 10%.
DNS process memory usage maximum threshold	Specify the maximum amount of memory resources that can be consumed by the DNS process before an event is raised. The default is 1024 KB.
Hostname for name resolution	Specify the name of the host to look up if you are using this script to test the hostname-to-IP address resolution. Default is localhost.
Host IP address that should be returned	Specify the correct IP address that should be returned by the DNS service if you are using this script to test the hostname-to-IP address resolution. The default is 127.0.0.1.

9.4 DNSSync

Use this Knowledge Script to check connectivity between two DNS servers. This script compares the DNS time stamp serial number for the current site with the time stamp serial number for the DNS site you specify. Both the remote site hostname and DNS domain name are required. This script raises an event if the serial numbers of the DNS servers are out of sync by more than the threshold value.

Resource Object

DNS service object

Default Schedule

The default interval for this script is **Every hour**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the serial numbers of the DNS servers are out of sync by more than the threshold value. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the servers are in sync, or ◆ **0** -- the servers are out of sync by more than the threshold value. In either case, the difference between the local and remote serial numbers is recorded in the detail message. The default is n.
DNS serial number difference maximum threshold	Specify the maximum difference that can occur between the local and remote DNS serial numbers before an event is raised. The default is 10.
Remote DNS host name	Specify the name of the DNS host computer. The default is `eclipse.netiq.com.`
DNS domain name	Specify the name of the DNS domain on the remote server. The default is netiq.com.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the serial numbers of the DNS servers are out of sync by more than the threshold value. The default is 8 (red event indicator).

9.5 RASConnections

Use this Knowledge Script to monitor the average number of connections to the Remote Access Server (RAS). This script raises an event if the total number of connections per minute exceeds the threshold you set.

Resource Object

RAS service object

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the total number of connections to the RAS per minute exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the average number of connections to the RAS per minute during the monitoring interval. The default is n.
Connections per minute maximum threshold	Specify the maximum number of connections to the server that can occur per minute before an event is raised. The default is 50 connections per minute.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of connections per minute exceeds the threshold. The default is 8 (red event indicator).

9.6 RASErrors

Use this Knowledge Script to monitor the total number of Remote Access Server (RAS) errors in an interval. Remote access server errors can include CRC errors, alignment errors, and timeout errors, for example. This script raises an event if the total number of errors exceeds the threshold.

Resource Object

RAS service object

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the total number of errors exceeds the threshold. The default is y.

Parameter	How to Set It
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the difference between the number of RAS errors during the last monitoring interval and the number of RAS errors during the current monitoring interval. The default is n.
Total number of errors maximum threshold	Specify the maximum number of errors that can occur before an event is raised. The default is 50 errors.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of errors exceeds the threshold. The default is 8 (red event indicator).

9.7 RASHealthCheck

Use this Knowledge Script to check the availability of the Remote Access Server (RAS) service. This script raises an event if the RAS service is not running and attempts to restart the service if the service is not running.

Resource Object

RAS service object

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event when RAS service is down?	Set to **y** to raise an event when the RAS service is down. The default is y.
Collect data for RAS service up/down?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the RAS service is running, or ◆ **0** -- the RAS service is not running. The default is n.
Auto-start service?	Set to **y** to automatically restart down services. The default is y.
Check RasMan?	Set to **y** to check whether the Remote Access Connection Manager (`RasMan`) service is running. The default is n.
Event severity level for service down; restart failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the RAS service is down and this script could not restart the service. The default is 5 (red event indicator).

Parameter	How to Set It
Event severity level for service down; restart succeeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the RAS service is down and this script successfully restarted the service. The default is 25 (blue event indicator).
Event severity level for service down; don't restart	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the RAS service is down and the *Auto-start service?* parameter is set to n. The default is 18 (yellow event indicator).

9.8 RASStat

Use this Knowledge Script to monitor the traffic on the Remote Access Server (RAS) server. This script raises an event if the total number of bytes transferred (transmitted and received) exceeds the threshold you set.

Resource Object

RAS service object

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the total number of bytes transferred per second by the RAS server since the script was last run exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of bytes transferred by the RAS server since the script was last run. The default is n.
Total bytes per second maximum threshold	Specify the maximum total number of bytes that can be transferred per second before an event is raised. The default is 500 bytes per second.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the total number of transferred bytes exceeds the threshold. The default is 8 (red event indicator).

9.9 WINSConflict

Use this Knowledge Script to monitor conflict activity on the Windows Internet Name Service (WINS) server. This script raises an event if the number of group and unique conflicts per second exceeds the threshold you set.

Group conflicts per second is the rate at which group registrations received by the WINS server resulted in conflicts with records in the database. *Unique conflicts* per second is the rate at which unique registrations and renewals received by the WINS server resulted in conflicts with records in the database

Resource Object

WINS service object

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event when the number of conflicts per second exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of unique conflicts and group conflicts per second. The default is n.
Total conflicts per second maximum threshold	Specify the maximum number of group and unique conflicts that can occur per second before an event is raised. The default is 4 conflicts per second.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of group and unique conflicts exceeds the threshold. The default is 8 (red event indicator).

9.10 WINSFailure

Use this Knowledge Script to report the number of failures per second on the Windows Internet Name Service (WINS) server. This script raises an event if the total failure rate for queries and releases exceeds the threshold you set.

Resource Object

WINS service object

Default Schedule

The default interval for this script is **Every 10 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the total failure rate for queries and releases on the WINS server exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of failed queries and failed releases per second. The default is n.
Total failures per second maximum threshold	Specify the maximum number of failures that can occur per second before an event is raised. The default is 2 failures per second.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of failures exceeds the threshold. The default is 8 (red event indicator).

9.11 WINSHealthCheck

Use this Knowledge Script to check the availability and CPU usage of the Windows Internet Name Service (WINS) service. This script raises an event if the WINS service is not running or if CPU usage exceeds the threshold you set. In addition, this script attempts to restart the WINS service if the service is not running.

Resource Object

WINS service object

Default Schedule

The default interval for this script is **Every 5 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event when WINS service is down?	Set to **y** to raise an event if the WINS service is not running. The default is y.
Event when WINS CPU usage is over threshold?	Set to **y** to raise an event if the percentage of CPU consumed by the WINS service exceeds the threshold. The default is y.
Collect data for WINS service up/down?	Set to **y** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- the WINS service is running, or ◆ **0** -- the service is not running. The default is n.

Parameter	How to Set It
Collect data for WINS CPU usage?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the percentage of CPU used by the WINS service. The default is n.
Auto-start service?	Set to **y** to automatically restart down services. The default is y.
Event severity level for service down; restart failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the WINS service is down and this script cannot restart it. The default is 5 (red event indicator).
Event severity level for service down; restart succeeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the WINS service is down and this script successfully restarted the service. The default is 25 (blue event indicator).
Event severity level for service down; don't restart	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the WINS service is down and the *Auto-start service?* parameter is set to n. The default is 18 (yellow event indicator).
Event severity level for CPU usage exceeded threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which CPU usage exceeds the threshold you set. The default is 8 (red event indicator).
WINS process %CPU maximum threshold	Specify the maximum percentage of CPU that the WINS process can consume before an event is raised. The default is 60%.

9.12 WINSQueries

Use this Knowledge Script to monitor query activity on the Windows Internet Name Service (WINS) server. This script raises an event when query failure rate or the total number of queries exceeds the threshold you set.

Resource Object

WINS service object

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event when the total number of queries per second or the number of failed queries per second exceeds the threshold. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the number of failed and successful queries per second. The default is n.
Failed query maximum threshold	Specify the maximum number of failed queries that can occur per second before an event is raised. The default is 2 query failures per second.

Parameter	How to Set It
Total query maximum threshold	Specify the maximum number of queries that can occur per second before an event is raised. The default is 50 queries.
Event severity level for failed queries exceeded threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of failed queries exceeds the threshold you set. The default is 8 (red event indicator).
Event severity level for total queries exceeded threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the total number of queries exceeds the threshold you set. The default is 25 (blue event indicator).

9.13 WINSReplication

Use this Knowledge Script to monitor replication activity on the Windows Internet Name Service (WINS) server. This script raises an event if either a planned or network-triggered replication does not occur within the specified period.

Prerequisite

This script requires the Microsoft SNMP service to be running.

Resource Object

WINS service object

Default Schedule

The default interval for this script is **Every 30 minutes.**

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Community	Provide the SNMP community string. The default is either the community name entered in AppManager Security Manager or *public* if no community name has been entered.
Event if planned replication failed?	Set to **y** to raise events when planned replications fail. The default is y.
Event if network-triggered replication failed	Set to **y** to raise events when network-triggered replications fail. The default is y.
Number of minutes without planned replication maximum threshold	Specify the maximum number of minutes to wait for a planned replication. If replication does not occur within the elapsed time, this script assumes that the planned replication failed. The default is 60 minutes.
Number of minutes without network replication maximum threshold	Specify the maximum number of minutes to wait for a network-triggered replication. If replication does not occur within the elapsed time, this script assumes that the network replication failed. The default is 60 minutes.

Parameter	How to Set It
Event severity level for planned replication failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a planned replication fails. The default is 8 (red event indicator).
Event severity level for network-triggered replication failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a network-triggered replication fails. The default is 15 (yellow event indicator).
Event severity level for SNMP failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which an SNMP failure occurs. The default is 18 (yellow event indicator).

9.14 WINSStat

Use this Knowledge Script to monitor the total registrations, renewals, and releases on the Windows Internet Name Service (WINS) server. This script raises an event when the number of registrations, renewals, or releases per second exceeds the threshold you set.

Resource Object

WINS service object

Default Schedule

The default interval for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise events when the total number of registrations, renewals, or releases per second on the WINS server exceeds the thresholds you set. The default is y.
Collect data?	Set to **y** to collect data for charts and reports. If enabled, data collection returns the total numbers of registrations, renewals, and releases per second on the WINS server. The default is n.
Registrations per second maximum threshold	Specify the maximum number of registration requests that can occur per second before an event is raised. The default is 20 requests per second.
Renewals per second maximum threshold	Specify the maximum number of renewal requests that can occur per second before an event is raised. The default is 20 requests per second.
Releases per second maximum threshold	Specify the maximum number of release requests that can occur per second before an event is raised. The default is 20 requests per second.
Event severity level	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a threshold is exceeded. The default is 15 (yellow event indicator).

10 ASYNC Knowledge Scripts

The ASYNC category provides Knowledge Scripts for monitoring Microsoft Windows servers for file changes, event log entries, and SNMP traps as these events occur. The ASYNC Knowledge Scripts run on an asynchronous schedule, which means that they run when a monitored event occurs to provide real-time feedback.

From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. In the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
FilesChanged	Monitors files for changes as they occur.
NTEventLog	Monitors Windows event logs for new entries matching the include and exclude criteria you define.
NTEventLogRX	Monitors Windows event logs for new entries matching the filter criteria you define using regular expressions.
SNMPTrap	Checks for incoming SNMP traps forwarded from NetIQ Trap Receiver.

10.1 Creating Filters with Regular Expressions

Some Knowledge Scripts enable you to use regular expressions to define include and exclude filters for pattern-matching against the text being evaluated. Depending on the Knowledge Script you are working with, you may be able to use regular expression include and exclude filters when you are setting job properties or you may be able to maintain your search criteria independent of the Knowledge Script parameters in a separate filter file. You may also be able to use regular expression modifiers to further refine your filtering.

For example, if your **include filter** is `replic.*` and you specify the modifier `i` to make the search case-insensitive, the regular expression contains the wildcard (`.`) and repeat (`*`) special characters, indicating you want to find strings that start with `replic` followed by any string of characters. Messages containing either `replication` or `replicated` are captured.

The format is the same for the exclude filter. For example, to find log entries that do not start with the string `success`, the exclude filter might look like this:

```
^success.*
```

If you are only searching for included strings, you can leave the exclude filter blank. If you want to retrieve all messages in the log in a given interval, you can specify `.*` for the include filter and leave the exclude filter blank.

Using Special Characters

The following special characters can be used in regular expressions:

Use This Character	For This Purpose
.	Wildcard for any one character
*	Repeat zero or more occurrences
^	Beginning of the line
\$	End of the line
\	Escape the next meta-character
|	Alternate matches
[ ]	Any character in the class set. You can specify individual characters or ranges.
( )	Grouping characters. For example, you can specify (a|b|c) to indicate a match with a, or b, or c.
+	Quantifier indicating one or more occurrences
?	Quantifier indicating zero or one occurrence
{*n*}	Quantifier indicating exactly *n* occurrence
\w	A word character (alphanumeric plus _)
\s	A white-space character
\d	A digit character

Using Regular Expression Modifiers

In addition to the special characters you can use in creating the regular expression, there are a number of modifiers that can be used to modify how pattern-matching is handled. Valid modifiers include:

Modifier	Description
c	Complements the search list
g	Matches globally as many times as possible
i	Makes the search case-insensitive
m	Treats the string as multiple lines
o	Interpolates variables only once
s	Treats the regular expression string as a single long line
x	Allows for regular expression extensions

For additional information about writing regular expressions, see your Perl documentation or other regular expression resources.

10.2 FilesChanged

Use this Knowledge Script to monitor files for changes to the current size, time stamp, and file attributes. This script raises an event if the size, time stamp, or attribute indicates the file has been modified.

Because this script checks the file properties rather than the file content, you can use this script with almost any file type.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Asynchronous**. Regardless of the schedule you select, once you start the Knowledge Script, its job status appears as **Running**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the size, time stamp, or attribute indicates the file has been modified. The default is y.
Collect data?	Set to **y** to collect data about file modifications. The default is n.

File path	Provide a valid directory path and filename to specify the file (or files) you want to monitor.
	To monitor more than one file with a similar name or to monitor the creation and deletion of a file, use pattern-matching characters to specify the filename. Use an asterisk (*) to match all characters and use a question mark (?) to match a single character. For example:
	`C:\NetIQ Corporation\Temp\NetIQ Corporation_Debug\Tomc\m?.*`
	matches the following files:
	`mo.log`
	`mo.log.bkup`
	`ms.log`
	`ms.log.bkup`
	To monitor all files in a directory (including new and deleted files), specify a directory path without a filename, for example, `C:\temp`.

Check file size?	Set to **y** to monitor the file’s current size. The default is y.
Check file last write time?	Set to **y** to monitor the file’s modification time stamp. The default is y.
Check file attributes?	Set to **y** to monitor the file’s attributes. The default is y.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which file modifications are detected. The default is 8 (red event indicator).

10.3 NTEventLog

Use this Knowledge Script to receive notification of specified Windows logs for entries that match the criteria you specify. You can filter event log entries by particular field values and set for each event type. Also, you can monitor the legacy Windows event logs, such as Application or System, and the custom event logs under the `Applications and Services Logs` folder in the Windows Event Viewer. This script raises an event when a log entry matches all your filter criteria. All event log entries that match the filtering criteria are returned in the event detail message.

This script requires the Microsoft **EventLog** service to be running on the managed client computer.

When you run this script, only new entries that are written to the event log after you start the job are reported. This script does not review the entire event log each time it runs.

On computers where the Security log is updated frequently, such as domain controller computers, consider using the NetIQ Security Manager product to securely and quickly consolidate Security logs with low impact to the server.

NOTE: To specify filters using regular expressions, use the NTEventLogRX Knowledge Script.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Asynchronous**. Regardless of the schedule you select, once you start the Knowledge Script, its job status appears as **Running**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Job Failure Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the job fails. The default is 5.
Event Log Monitoring	

Parameter	How to Set It
Event logs to filter	Provide a comma-separated list of the event logs you want to monitor, or enter an asterisk (*) to monitor all event logs on the agent computer. When monitoring all logs, up to 63 event logs can be monitored by one job, so on agents where more than this number exists, some logs may not be monitored when running a job in this manner. The following is an example of specifying multiple event logs: `System,Application,Microsoft-Windows-Bits-Client/Operational` The default is Application. **NOTE:** Monitoring all event logs can have a significant performance impact on the agent computer because every event written will need to be reviewed by the job.
Ignore event log matches occurring during agent maintenance mode?	Select **Yes** for the Knowledge Script to ignore event log matches that occur while the agent is in maintenance mode. No events will be raised or data collected for matches that are written to the event logs during this time. The default is Yes.
Filters	
Use case-sensitivity for specified filter strings?	Select **Yes** to enable case-sensitivity for the specified filter strings. The default is unselected.
Event Types	
Filter critical events?	Select **Yes** to filter critical events. The default is Yes.
Filter error events?	Select **Yes** to filter error events. The default is Yes.
Filter warning events?	Select **Yes** to filter warning events. The default is Yes.
Filter information events?	Select **Yes** to filter information events. The default is Yes.
Filter success audit events?	Select **Yes** to filter success audit events on agent computers running Windows Server 2003 or prior operating systems. The default is Yes. **NOTE:** `Audit Success` is one of the Keywords field in the `Security event` log on Windows Server 2008 and later.
Filter failure audit events?	Select **Yes** to filter failure audit events on agent computers running Windows Server 2003 or prior operating systems. The default is Yes. **NOTE:** `Audit Failure` is one of the Keywords field in the `Security event` log on Windows Server 2008 and later.
Event source filter	To filter for events generated by a particular source, such as `SQLExecutive`, `SNMP`, or `Service Control Manager`, enter an appropriate filter string. This script looks for matching entries in the event log's **Source** field. Multiple strings can be entered separated by commas. There is no default value set, so no filtering by Source takes place by default. You must enter the appropriate source(s) to filter events generated by the source(s). The filter string must contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you specify only include criteria, the colon is not necessary.

Parameter	How to Set It
Event category filter	To filter events in a particular category, such as `Server` or `Logon`, enter an appropriate filter string. The Knowledge Script looks for matching entries in the event log's **Category** field. There is no default value set, so no filtering by Category takes place by default. You must enter the appropriate category(ies) to filter events generated by the category(ies). The filter string must contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you specify only include criteria, the colon is not necessary.

Event ID filter	Use this parameter to filter for particular event IDs.
	Provide an Event ID or ID range, for example 100-2000). This script will look for matching entries in the Event Log **Event ID** field. Separate multiple IDs and ranges with commas. For example:
	`1,2,10-15,202`
	The search string can contain criteria to include entries, exclude entries, or both. Separate include and exclude criteria with a colon (:) using the following format: `include:exclude`. For example, to include event IDs 10 through 15 and to exclude event ID 202, enter the following:
	`10-15:202`
	If you specify only include criteria, the colon is not necessary.
	There is no default value set, so no filtering by Event ID takes place by default. You must enter the appropriate Event ID(s) to filter events by the Event ID(s).

Event user filter	To filter events associated with a particular user, enter a filter string that includes the user's domain name and user name, separated with a backslash "\", or enter a filter string using just the user name you want events in the monitored event log to contain. For example, `NetIQ Corporation\Tom Jones`. This script looks for matching entries as they appear in the **User** field of the event log's Event Detail dialog box (To view the Event Details dialog box, double-click a log entry in the Event Viewer). Separate multiple strings with commas (,). There is no default value set, so no filtering by User takes place by default. You must enter the appropriate user(s) to filter events generated by specific user(s). The filter string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you specify only include criteria, the colon is not necessary.

Parameter	How to Set It
Event computer filter	To filter events generated by a particular computer, enter an appropriate filter string. This script looks for matching entries in the event log's **Computer** field. Multiple strings can be entered separated by commas. There is no default value set, so no filtering by Computer takes place by default. You must enter the appropriate computer(s) to filter events generated by the computer(s). The filter string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you specify only include criteria, the colon is not necessary.
Event keywords filter	To filter by keyword for events generated by a particular computer. Provide a search string. This script will look for matching entries in the Event Log **Keywords** field. Separate multiple strings with commas. The search string can contain criteria to include entries, exclude entries, or both. Separate include and exclude criteria with a colon (:) using the following format: `include:exclude`. If you specify only include criteria, the colon is not necessary. There is no default value set, so no filtering by Keywords takes place by default. You must enter the appropriate Keywords to filter events containing specific strings in the Keywords field.
Event description filter	To filter events with a particular detail description, enter an appropriate filter string. This script looks for matching entries in the event log's **Description** field. Multiple strings can be entered separated by commas. There is no default value set, so no filtering by Description takes place by default. You must enter the appropriate event log's description(s) to filter events generated by the event log's description(s). The filter string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you specify only include criteria, the colon is not necessary.
Event Notification	
Raise event if log entries matching criteria are found?	Select **Yes** to raise an event when log entries match your filtering criteria. The default is Yes.
Severity for critical events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which critical events are detected. The default is 5 (red event indicator)
Severity for error events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which error events are detected. The default is 10 (red event indicator).
Severity for warning events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which warning events are detected. The default is 15 (yellow event indicator).
Severity for information events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which information events are detected. The default is 25 (blue event indicator).

Parameter	How to Set It
Severity for success audit events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which success audit events are detected. The default is 10 (red event indicator).
Severity for failure audit events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which failure audit events are detected. The default is 5 (red event indicator).
Data Collection	
Collect data for log entries that match criteria?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns detail about log entries that match your filtering criteria. The default is unselected.

10.4 NTEventLogRX

Use this Knowledge Script to scan specified Windows logs for entries that match the criteria you specify. You can filter the event log entries by event type and by specifying a combination of include and exclude strings for each event field using regular expressions. This script raises an event when a log entry matches all your filter criteria. All event log entries that match the filtering criteria are returned in the event detail message.

Use the *Filter the [...] field with* parameters to control which fields to filter and the filtering criteria to use to find specific information, such as events associated with a specific user or computer name. With this script, specify the filtering criteria for each field you are interested in using a regular expression, or specify the name of a file that contains all your filtering criteria.

For more information, see Section 10.1, “Creating Filters with Regular Expressions,” on page 345.

Once you start the Knowledge Script job, any new entries written to the event log that match your criteria are reported. This script does not scan the entire log for any previously-reported events.

This scripts requires the Microsoft **EventLog** service to be running on the managed client computer.

On computers where the Security log is updated frequently, such as domain controller computers, consider using the NetIQ Security Manager product to securely and quickly consolidate Security logs with low impact to the server. For more information, visit the NetIQ Web site at http://www.netiq.com/products/sm/default.asp.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Asynchronous**. Regardless of the schedule you select, once you start the Knowledge Script, its job status appears as **Running**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Job Failure Notification	
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the job fails. The default is 5.
Event Log Monitoring	
Filters	
Use case-sensitivity for specified filter strings?	Select **Yes** to enable case-sensitivity for the specified filter strings. The default is unselected.
Event logs to filter	Provide a comma-separated list of the event logs you want to monitor, or enter an asterisk (*) to monitor all event logs on the agent computer. When monitoring all logs, up to 63 event logs can be monitored by one job, so on agents where more than this number exists, some logs may not be monitored when running a job in this manner. The following is an example of specifying multiple event logs: `System,Application,Microsoft-Windows-Bits-Client/Operational` **NOTE:** Monitoring all event logs can have a significant performance impact on the agent computer because every event written will need to be reviewed by the job.
Filter the … field with the regular expression	Use a regular expression to indicate the criteria to look for in each event log field: ◆ **Type**. To filter information based on the type of event (such as `Critical`, `Error`, `Warning`, `Information`), use a regular expression to identify the type of event entries to include ◆ **Source**. To filter the entries generated by a particular source (such as `SQLExecutive`, `SNMP`, or `Service Control Manager`), use a regular expression to identify the source of event entries to include. ◆ **Category**. To filter information based on a particular category (such as `Server` or `Logon`), use a regular expression to identify the category of event entries to include. ◆ **Event ID**. To filter information based on the event ID, use a regular expression to identify the event IDs to include. ◆ **User**. To filter information based on the user name field, use a regular expression to identify the user names to include. ◆ **Computer**. To filter information based on the computer name, use a regular expression to identify the computers to include. ◆ **Keywords**. To filter events based on the keywords field, use a regular expression to identify the keywords include. ◆ **Description**. To filter information based on the event description, use a regular expression to indicate the description or portion of description to include.

Parameter	How to Set It
Full path to a file containing filtering criteria	To specify matching expressions in an external file, type the full path to a file containing the filtering criteria you want to match. For example: `C:\TEMP\MyFilters.txt.` **NOTE:** If you specify a filter file, AppManager ignores the `Filter the [...]` field with parameters. However, if AppManager cannot process the filter file, the script raises an event (for example, fail to process filter file `C:\async.xml`). For additional information, see "Using an External Filter File" on page 354.
Event Notification	
Raise event if log entries matching criteria are found?	Select **Yes** to raise an event when log entries match your filtering criteria. The default is Yes.
Severity for critical events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which critical events are detected. The default is 5 (red event indicator).
Severity for error events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which error events are detected. The default is 10 (red event indicator).
Severity for warning events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which warning events are detected. The default is 15 (yellow event indicator).
Severity for information events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which information events are detected. The default is 25 (blue event indicator).
Severity for success audit events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which success audit events are detected. The default is 10 (red event indicator).
Severity for failure audit events	Set the event severity level, from 1 to 40, to indicate the importance of an event in which failure audit events are detected. The default is 5 (red event indicator).
Data Collection	
Collect data for log entries that match criteria?	Select **Yes** to collect data for charts and reports. When enabled, data collection returns detail about log entries that match your filtering criteria. The default is unselected.

Using an External Filter File

Use this Knowledge Script to specify regular expressions for each event log field as script properties or maintain your search criteria independent of the script parameters in a separate filter file.

In many cases, specifying an external filter file provides greater flexibility and makes modifying your search criteria more straightforward, because you can add almost any number of expressions. You do not need to modify the script properties through the Operator Console or Control Center to pick up your changes.

To use a filter file:

- Identify the strings that you want to find a match for (that is, the entries you want to include in your results).

- Create a text file with one regular expression string per line to locate matching strings. Each line in the file consists of a parameter keyword followed by a colon (:), a tab or blank space, and the regular expression. Or the filter file can be written in XML.
- Ensure the file exists on the target computer.
- Provide the absolute path to the file on the local computer in the *Full path to a file containing filtering criteria* parameter and start the job.

Formatting the Filter File

There are two valid formats for the filter file: a simple table format to define the strings to include, and an XML format that allows you to define more complex include and exclude filtering. For both formats, the parameter name keywords are required, but the field values can be left blank if no filtering is needed.

Select a file format appropriate for the complexity of the filtering you need to do.

Table Format

The table format provides a simple way to create the filter file. Each filtering section in the file begins with `EventStart` and ends with `EventEnd`. If an entry in the event log matches all the criteria you specified within a filtering section, it is considered a match and an event is raised in AppManager. If you have more than one filtering section, an entry matching either section raises an event.

For example, the following table format file provides two filter sections:

```
EventStart
CaseSensitive:  n
Log:  System
Type:  Error|Warning|Information
Source:  ^SQL*
Category:  *
EVENTID:  1[0-9][0-9][0-9]
User:  Sam|Joe|Chris
Computer:  SFO*
Description:  ($Error.*)|(.*error.*occurred.$)
EventEnd
EventStart
CaseSensitive:  n
Log:  Application
Type:  Error|Warning|Information
Source:  ^SQL*
Category:  *
EVENTID:  1[0-9][0-9][0-9]
User:  Sam|Joe|Chris
Computer:  SFO*
Description:  ($Error.*)|(.*error.*occurred.$)
EventEnd
```

NOTE: If you specify only one filter section, do not include the EventStart and EventEnd lines in the file.

XML Format

The XML format is somewhat more sophisticated and more flexible than the table format. The XML format allows you to set both include and exclude filters using the `<Include>` and `<Exclude>` tags and to combine these filter sets to define the search criteria. Each filtering section in the file begins with the `<Events>` tag. An log entry must match all the criteria you specified within a filtering section for it to be considered a match.

For example:

```
<?xml version = "1.0" standalone = "yes"?>
<EventLogConfig Name = "Event Filter" Type = "EVENT_FILTER_CUSTOM" ID = "76">
<Include>
   <Events>
      <Log>Application</Log>
      <Type>INFORMATION|WARNING|ERROR</Type>
      <Source><Net*]></Source>
      <Category>*</Category>
      <EVENTID>2*</EVENTID>
      <User>*</User>
      <Computer>*</Computer>
      <Description><![CDATA[Event.]]></Description>
      <CaseSensitive>y</CaseSensitive>
   </Events>
   <Events>
      <Log>System</Log>
      <Type>Warning</Type>
      <Source>RSVP</Source>
      <Category>*</Category>
      <EVENTID>*</EVENTID>
      <User>*</User>
      <Computer>SHASTA</Computer>
      <Description>RSVP*</Description>
      <CaseSensitive>y</CaseSensitive>
   </Events>
</Include>
</EventLogConfig>
```

NOTE: If a field contains a regular expression that conflicts with XML syntax or includes special characters, you can use `![CDATA[regular_expression]]` to enclose the expression and prevent parsing problems.

10.5 SNMPTrap

Use this Knowledge Script to check for SNMP traps forwarded from NetIQ Trap Receiver (Trap Receiver). This script raises an event when an SNMP trap is received and when Trap Receiver is unavailable or subsequently becomes available. In addition, this script generates datastreams for Trap Receiver availability. For more information, see "Working with NetIQ Trap Receiver" on page 359.

Prerequisites

- Trap Receiver is not installed automatically when you install the AppManager for Microsoft Windows module. You must start Trap Receiver manually by running the following:

  ```
  \AppManager\bin\NetIQTrapReceiver_Setup.exe
  ```

- This script supports SNMP v1, v2, and v3. If you use SNMP v3, configure your SNMP permissions in AppManager Security Manager. For more information, see “Configuring SNMP Permissions” on page 362.
- Trap Receiver filters SNMP traps based on the criteria you provide in the script parameters: IP address, hostname, or object identifier (OID). This script can translate numerical OIDs to their object descriptor (ODE) counterparts. The translation process requires access to the Management Information Base (MIB) files that reference the OIDs and ODEs. For more information, see “Adding MIBs for Use By Trap Receiver” on page 364.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Asynchronous**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Job Error Notification	
Event severity when an error occurs	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the SNMPTrap job fails. The default is 15. The default is 15.
Trap Filters	
Filter by IP source address or hostname	Provide the IP address or hostname of the SNMP source from which to receive traps. For example: `10.10.10.10` Separate multiple addresses or hostnames with a comma (,). **Notes** ◆ For SNMP v1 and v2, leave this parameter blank (the default) to receive traps from any source IP address or hostname. ◆ For SNMP v3, you must provide at least one IP address or hostname from which to receive traps. Trap Receiver can receive traps only from devices that are registered in `net-snmp` with the appropriate profile information: username, security mode, and passwords.

Parameter	How to Set It

Filter by object identifier	Provide the object identifier of the trap messages you want to receive. The object identifier is defined by the SNMP source agent.
	You can use OID or ODE notation to specify the object identifier. To filter for more than one object identifier, separate each notation with a comma (,).
	If you leave this parameter blank, the script does not use the object identifier to filter for events.
	If you are using ODE notation, use a case-sensitive descriptor. For example:
	`system.sysUptime.0`
	If you are using OID notation, include the dot (.) at the beginning of the identifier. For example:
	`.1.2.6.1.4.1.1691`
	NOTE: This script filters for an exact match to the OID you provide. If your OID is `.1.2.6.1.4.1.1691`, the script will not match all OIDs that begin with `.1.2.6`. It matches only the OID you specified.

Filter by MIB sub-tree	Provide the part of the MIB tree (sub-tree) about which you want to receive events. For example: `1.3.6.1.4.1.9` Separate multiple sub-trees with a comma (,). If you leave this parameter blank, the script reports events related to the entire MIB tree. You can use this parameter for any trap; however, this parameter uses SNMP v2 terminology.
Filter by generic trap number	Specify a generic trap number to filter trap messages that use the same OID for more than one trap message. You usually do not need to filter for generic trap message numbers if the OID is unique. The generic value of the OID is defined by the SNMP source agent. If you leave this parameter blank, the script does not use a generic value to filter for events.
Filter by specific trap number	Specify a specific trap number to filter trap messages that use the same OID for more than one trap message. You usually do not need to filter for specific trap message numbers if the OID is unique. The specific value of the OID is defined by the SNMP source agent. If you leave this parameter blank, the script does not use a specific value to filter for events.
Filter by enterprise	Provide the enterprise from which you want to receive events. The enterprise is defined in MIB 1.3.6.1.4.1.9.87.2. Separate multiple enterprises with a comma (,). If you leave this parameter blank, the script reports events related to all enterprises. You can use this parameter for any trap; however, this parameter uses SNMP v1 terminology.

Parameter	How to Set It
Event Notification	
Raise event when SNMP trap received?	Select **Yes** to raise an event when an SNMP trap matching your filter criteria is received. The default is Yes.
Event severity when SNMP trap received	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a trap message matches all filter criteria. The default is 15. You can adjust the severity depending on which type of message you are checking for.
Format trap data according to SNMP version	Select the version of SNMP whose formatting should be used for trap event messages. The data provided by each format is the same; only the layout is different.
Raise event for Trap Receiver availability?	Select **Yes** to raise an event when Trap Receiver becomes unavailable and when Trap Receiver becomes available once again. The default is Yes.
Event severity when Trap Receiver is unavailable	Set the severity level, from 1 to 40, to indicate the importance of an event in which Trap Receiver becomes unavailable. The default is 5.
Event severity when Trap Receiver becomes available	Set the severity level, from 1 to 40, to indicate the importance of an event in which Trap Receiver becomes available. The default is 25.
Data Collection	
Collect data for received traps?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns information about received traps based on your search criteria. The default is unselected.
Collect data for Trap Receiver availability?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns "1" if Trap Receiver is available and "0" if Trap Receiver is unavailable. The default is unselected.
Interval for collecting Trap Receiver availability data	Specify the frequency with which the script collects Trap Receiver availability data. The default is every 5 minutes.

Working with NetIQ Trap Receiver

In general, a trap receiver is an application that receives traps from SNMP agents. Trap Receiver receives and filters SNMP traps, and then forwards the traps to AppManager. Trap Receiver runs as a service, `NetIQTrapReceiver.exe`, and may compete for port usage with any other trap receiver installed on the same computer.

What is NetIQ Trap Receiver?

At its most basic, a trap receiver is an application that receives traps from SNMP agents. Trap Receiver receives, filters, and forwards SNMP traps to AppManager. When you use Trap Receiver with the AppManager for Microsoft Windows module, the SNMPTrap Knowledge Script raises events when SNMP traps are received.

What is an SNMP Trap?

Simple Network Management Protocol (SNMP) is a protocol-based system used to manage devices on TCP/IP-based networks. From devices on which an SNMP agent resides, such as routers and switches, SNMP sends unsolicited notifications, called traps, to network administrators when thresholds for certain conditions are exceeded. These conditions are defined by the vendor in a device's Management Information Base (MIB). The network administrator sets the thresholds.

Traps are composed of Protocol Data Units (PDUs). Each PDU contains the following information, organized in various ways depending on the version of SNMP in use:

- SNMP version number
- Community name of the SNMP agent
- PDU type
- Enterprise OID (object identifier), a unique number that identifies an enterprise and its system objects in the MIB
- IP address of the SNMP agent
- Generic trap type: Cold start, Warm start, Link down, Link up, Authentication failure, Egp Neighbor Loss, and Enterprise
- Specific trap type. When the Generic trap type is set to "Enterprise," a specific trap type is included in the PDU. A specific trap is one that is unique or specific to an enterprise.
- Time the event occurred
- Varbind (variable binding), a sequence of two fields that contain the OID and a value

Understanding Trap Receiver Architecture

Trap Receiver operates on a Client-Server architecture: the *Server*—the stand-alone Trap Receiver application—receives, filters, and forwards SNMP traps to the *Client*—an application that receives traps, such as AppManager. The Server may receive traps from standard UDP port 162 or from any other configured port. The Client and the Server can reside on the same computer or on separate (proxy) computers.

Communication between Client and Server is implemented as XML messages over a TCP connection. Only one Server is allowed per computer, however, several Clients are allowed per computer. Clients that are registered to the same Server share the same TCP connection. The Server TCP port should be known to all potential Clients.

Understanding the Trap Receiver Configuration File

The configuration file for Trap Receiver, `NetIQTrapReceiver.conf`, identifies the UDP and TCP ports used by Trap Receiver: the UDP port is used for receiving traps; the TCP port is used for communicating with the Client, such as AppManager or another supported NetIQ application. The configuration file also identifies the level of logging you want to use and whether port forwarding is enabled.

By default, the configuration file is installed in `[`*installation directory*`]\config`, and has the following format:

```
##############################################################
#
# NetIQTrapReceiver.conf
#
# A configuration file for NetIQ Trap Receiver
#
##############################################################
#########################
# TCP port
# Syntax: tcp_port [port]
# E.g. : tcp_port 2735
#########################
tcp_port 2735
#########################
# UDP port
# Syntax: udp_port [port]
# E.g. : udp_port 162
#########################
udp_port 162
#########################
# Forwarding
# Syntax: forward [address]:[port] [v1]
# E.g. : forward 127.0.0.1:1000 v1
#########################
#########################
# Log level
# Syntax: log_level error|warning|info|debug|xml
# E.g. : log_level info
#########################
log_level debug
```

If the configuration file cannot be found, cannot be parsed, or does not contain one of the required values, Trap Receiver is initialized with the default configuration as shown above.

When changing values in the configuration file, take into account the following:

- If you change the TCP port number, stop all asynchronous Knowledge Script jobs associated with the modules that support Trap Receiver. Run the Discovery_NT Knowledge Script on all monitored devices to enable the devices to recognize the new TCP port number.
- If you change the UDP port number, also change the UDP port number configured on the devices that send traps to Trap Receiver.
- If another service uses port 2735 or port 162, Trap Receiver *will not start*. The Trap Receiver log file will contain different levels of messages, based on the log_level you choose. Either change the port numbers in the configuration file, stop the service that is using the default Trap Receiver port numbers, or forward the traps coming in to UDP port 162.

- To forward incoming traps to another trap receiver, such as Microsoft SNMP Trap Service, set the Forwarding values as follows:
  `forward [`*IP address of other trap receiver*`]:[`*port number of other trap receiver*`] [`*SNMP version*`]`.
  For example: `forward 10.40.40.25:167 v1`. By default, incoming traps are not forwarded. For more information, see "Coexisting with Microsoft SNMP Trap Service" on page 362.
- Restart Trap Receiver after any change to the configuration file. From Control Panel, double-click **Administrative Tools** and then double-click **Services**. Right-click **NetIQ Trap Receiver** and select **Restart**.

Coexisting with Microsoft SNMP Trap Service

Two trap receivers cannot be in use on the same computer while using the same standard UDP port (162). If NetIQ Trap Receiver and another trap receiver such as Microsoft SNMP Trap Service are installed on the same computer and both are receiving traps, then configure Trap Receiver to use the standard UDP port and to forward incoming traps (UDP forwarding) to the other trap receiver. For more information, see "Understanding the Trap Receiver Configuration File" on page 361.

Then, configure the other trap receiver to use a different, non-standard, UDP port that is not in use by another application. The following are instructions for configuring Microsoft SNMP Trap Service.

To configure Microsoft SNMP Trap Service to use another port:

1. Navigate to `\system32\drivers\etc`.
2. Open the **services** file.
3. In the row for `snmptrap`, change the value for **udp** from 162 to another port number that is not in use by any other application. Use the same port number you set as the forwarding port in the Trap Receiver configuration file. For more information, see "Understanding the Trap Receiver Configuration File" on page 361.
4. Save and close the **services** file.
5. Restart Windows SNMP Trap Service. In Control Panel, double-click **Administrative Tools** and then double-click **Services**. Right-click **SNMP Trap Service** and select **Restart**.

TIP: To see which ports are in use, run `netstat.exe` from a command prompt. Then select an available port as the port for the other trap receiver service.

Configuring SNMP Permissions

For each device you want to monitor for SNMP v3 traps, configure Simple Network Management Protocol (SNMP) information in AppManager Security Manager *before* you run the SNMPTrap Knowledge Script. You do not need to configure permissions for SNMP v1 or v2.

By configuring SNMP information, you provide AppManager the permission it needs to access the Management Information Bases (MIBs) on SNMP-enabled devices.

The AppManager for Microsoft Windows module supports the following modes for SNMP v3:

- No authentication; no privacy
- Authentication; no privacy
- Authentication and privacy

In addition, the module supports the following protocols for SNMP v3:

- MD5 (Message-Digest algorithm 5, an authentication protocol)
- SHA (Secure Hash Algorithm, an authentication protocol)
- DES (Data Encryption Standard, encryption protocol)

Your SNMP v3 implementation may support one or more combinations of mode and protocol. That combination dictates the type of information you configure in AppManager Security Manager: user name (or entity), context name, protocol name, and protocol passwords.

Configure SNMP information for each device you want to monitor. On the Custom tab in Security Manager, complete the following fields:

Field	Description
Label	`SNMPTrap`
Sub-label	Indicate whether the community string information will be used for a single device or for all devices: ◆ *For a single device,* type the <*device name*>. ◆ *For all devices*, type `default`.
Value 1	SNMP user name, or entity, configured for the device. All SNMP v3 modes require an entry in the **Value 1** field.
Value 2	Name of the context associated with the user name or entity you entered in the **Value 1** field. A context is a collection of SNMP information that is accessible by an entity. If possible, enter a context that provides access to all MIBS for a device. *If the device does not support context*, type an asterisk (`*`). All SNMP v3 modes require an entry in the **Value 2** field.
Value 3	Combination of protocol and password appropriate for the SNMP v3 mode you have implemented. ◆ For *no authentication/no privacy mode*, leave the **Value 3** field blank. ◆ For *authentication/no privacy mode*, type `md5` or `sha` and the password for the protocol, separating each entry with a comma. For example, type `md5,abcdef` ◆ For *authentication/privacy mode*, type `md5` or `sha` and the associated password, and then type des and the associated password, separating each entry with a comma. For example, type `sha,hijklm,des,nopqrs`

Adding MIBs for Use By Trap Receiver

The SNMPTrap Knowledge Script can translate numerical OIDs to their ODE counterparts. The translation process requires access to the Management Information Base (MIB) files that reference the OIDs you specified as filters in the script parameters.

You must copy the necessary MIB files to the default MIBs directory on the computer on which NetIQ Trap Receiver is installed. After installing the MIBs, reload the MIBs directory so the new MIBs can be compiled for use by Trap Receiver.

To add MIBs to the MIB directory and reload the directory:

1. On the computer on which Trap Receiver is installed, copy all necessary MIB files to the default directory: `\Program Files\NetIQ\AppManager\bin\MIBs`. Ensure you copy MIB files for all your modules, not only the MIB files for the module with the trap definition.
2. On that same computer, restart the AppManager agent services: `NetIQmc` (NetIQ AppManager Client Resource Monitor) and `NetIQccm` (NetIQ AppManager Client Communication Manager). Restarting the services allows Trap Receiver to load the MIB files.

11 Action Knowledge Scripts

The AppManager Action Knowledge Scripts perform corrective or responsive actions when events are raised. From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. In the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
Diagnose	Triggers AppManager Diagnostic Console to run a diagnosis of the target computer.
DiagnoseNortelIPT	Triggers NetIQ Vivinet Diagnostics to run a diagnosis of VoIP quality in a Nortel CS1000 IP Telephony environment.
DiagnoseVoIPQuality	Triggers NetIQ Vivinet Diagnostics to run a diagnosis of voice quality between two phones or two endpoints.
DominoCommand	Issues a Domino command to a Domino server.
DosCommand	Runs a non-interactive DOS command.
DumpTran	Dumps or truncates the SQL Server transaction log.
ExtendedSNMPTrap	Sends an extended SNMP trap message to a specified list of computers.
IISContinueSite	Continues a paused Internet Information Services (IIS) site.
IISPauseSite	Pauses an IIS site.
IISRestartServer	Restarts an IIS server.
IISRestartSite	Restarts an IIS site.
MapiMail	Sends mail to one or more email users.
Messenger	Sends a Messenger service message that contains AppManager event information to a specified computer.
NotesMail	Sends mail to one or more Lotus Domino/Notes email users.
NTEventLog	Writes an event to the Windows Event Log.
Page	Sends a paging call to one or more recipients in response to an event.
RebootSystem	Shuts down and restarts a computer when an event is raised.
RestartServices	Stops and restarts Windows services.
RunDiscoveryNetworkDevice	Used with NetworkDevice_Device_Uptime to rediscover devices that reboot during monitoring.
RunKS	Runs up to three other Knowledge Scripts.
RunPowerShell	Runs a non-interactive Windows PowerShell command.
RunSql	Runs SQL statements or stored procedures.
SendReportToPrinter	Sends a report to the printer that is the default for the managed client on which the Report agent is running.

Knowledge Script	What It Does
SMTPMail	Sends mail using SMTP to one or more users.
SMTPMailRpt	Sends the first page of a report to a list of recipients.
SNMPTrap	Sends an extended SNMP trap to one or more computers.
StartServices	Starts specified Windows services.
StopServices	Stops specified Windows services.
Traceroute	Collects exception traceroute data between a specified source and target location in response to an event in a separate Knowledge Script.
TracerouteNetworks-RT	Collects exception traceroute data between a specified source and target location in response to an event in a separate Networks-RT Knowledge Script.
UpdateEventStatus	Provides AppManager event status details from one or more specified computers.
UXCommand	Runs a non-interactive UNIX command in response to an event.
WriteMsgToFile	Writes AppManager event information to a specified file.

11.1 Configuring Security Manager for Action Knowledge Scripts

To configure an Action Knowledge Script to use SQL Server authentication, in Knowledge Scripts where it is applicable and allowed, the SQL Server login username must have its password added to the Security Manager with the proper label indicating the name of the SQL Server instance to which it will be connected from the Knowledge Script.

On the **Custom** tab in Security Manager, complete the following fields for the SQL Server to which a connection is to be made:

Field	Description
Label	`sql$<Server Instance name>` For example, if the SQL Server instance to which the Knowledge Script is to connect is SERVERTEST, you would type `sql$SERVERTEST`.
Sub-label	SQL user name that exists in the SQL Server instance.
Value 1	Password for the user entered in the **Sub-Label** field.
Extended application support	Required field. Encrypts the user name and password in Security Manager.

Resource Objects

Windows servers running AppManager

Default Schedule

By default, this script is only run once for each computer.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise an event if discovery succeeds?	Select **Yes** to raise an event when discovery succeeds. The default is unselected.
Event severity when discovery succeeds	Set the event severity level, from 1 to 40, to reflect the importance of an event in which when the discovery succeeds. The default is 25.
Raise event if discovery fails?	Select **Yes** to raise an event when discovery fails. The default is Yes.
Event severity when discovery fails	Set the event severity level, from 1 to 40, to reflect the importance of an event in which the discovery fails. The default is 5.
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the Discovery_AMHealth job itself fails. The default is 35.
SQL Server login	Specify the SQL user name required for access to the AppManager repository (QDB). Leave this field blank to use Windows NT authentication. **NOTE:** If you want to use a specific SQL Server login account, use Security Manager to update the AppManager repository with the SQL Server logins that you want to use. .

11.2 Diagnose

Run this Knowledge Script to trigger AppManager Diagnostic Console to diagnose a problem on a *target* computer that has raised an event. The diagnosis is driven from the *console* computer specified in the **Location** field on the **Action** tab.

When launched, Action_Diagnose performs the following steps:

- Verifies that Diagnostic Console 2.1 or later is installed. If Diagnostic Console 2.1 or later is not installed, this script raises an event indicating that the diagnosis cannot be performed.
- Determines whether Action_Diagnose is already running on the console computer. If a diagnosis is already running, Action_Diagnosis raises an event indicating that a diagnosis is in progress. Only one instance of Action_Diagnose can be running at any given time.
- Invokes Diagnostic Console to perform the diagnosis and generate an `.html` diagnostic report. Diagnostic Console collects Windows, Exchange, or Active Directory data based on the configuration of the target computer.

NOTE: The target computer must have Diagnostic Console version 2.1 or later installed in order for Active Directory data to be collected. Windows and Exchange data can be collected with Diagnostic Console version 2.0 or later.

- Generates a report using the *Output folder prefix* and *Use Report Agent settings* parameters, or the *Full path to root of output folders* parameter if you are not using the Report agent.
- Upon completion of the diagnosis, raises an event that contains the results of the diagnosis. An event for a successful diagnosis will contain either a URL to the `default.htm` file (if *Use Report Agent settings* is set to y) or the computer name and full path of the location of the output files. An event for an unsuccessful diagnosis contains an error message explaining why the diagnosis was unsuccessful.

Prerequisite

NetIQ Object Linking and Embedding (`NetIQOLE`) must be registered on the computer on which this script runs. `NetIQOLE` is an automation object that allows AppManager to be run from a command-line. For more information, see the *Administrator Guide for AppManager* at the NetIQ AppManager Documentation Web site.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Length of time to run diagnosis	Enter the amount of time that you want a Diagnosis to run. The default is 300 seconds. The maximum allowable run time is 900 seconds. A longer run time will produce more data than reports can generate in a timely fashion. The minimum allowable run time is 60 seconds. A shorter run time is not enough for the agent to collect data and send it to the repository.
SQL Login Name	Enter the SQL user name required for access to the Appmanager repository when collecting Exchange and Active Directory data. Leave this parameter blank to use NT Authentication for accessing the repository.
SQL Password	Enter the SQL password required for access to the AppManager repository when collecting Exchange and Active Directory data.
Output folder prefix	Enter a prefix for the output folder that is generated by the diagnosis. The output folder then uses this prefix in the following naming convention: *Prefix_ComputerName_DateTime*. The default prefix is `Diag`. **NOTE:** The `ComputerName` is the name of the computer being diagnosed.
Use Report Agent settings?	Set to **y** to specify that the Diagnostic results should be integrated into the AppManager Web management server (the Report Binder). The default is y.
Full path to root of output folders (if not using Report Agent)	Enter the full path to the root of where the output folders will be created. **NOTE:** This parameter is ignored if **Use Report Agent settings** is enabled.

Parameter	How to Set It
Event severity when …	Enter a severity level, between 1 and 40, to indicate the importance of the following events: ◆ error. Raises an event when the diagnosis does not complete successfully. The default is 15. ◆ successful. Raises an event when the diagnosis completes successfully. The default is 35.

11.3 DiagnoseNortelIPT

Run this Knowledge Script in your Nortel Communication Server 1000 IP Telephony environment to trigger NetIQ Vivinet Diagnostics to diagnose a call quality problem between two Nortel IP phones.

Configure this Action on the NortelCS_Alarms Knowledge Script. A Diagnosis is triggered when the Alarms script raises events for the following QoS alarms (SNMP traps): QOS0022, QOS0024, QOS0026, QOS0028, QOS0030, QOS0032, and QOS0034. The Diagnosis makes use of the RTCP-XT statistics included in the SNMP trap.

You must run this Action on a computer where Vivinet Diagnostics 2.0 (or later) is installed. In addition, you must have already configured Vivinet Diagnostics with the security information for accessing the Call and Signaling Servers. For more information, see the *User Guide for Vivinet Diagnostics*.

Only one Diagnosis can run at any time. If a second Action is triggered while a Diagnosis is already in progress, the second Action will complete, but indicate that it could not run the Diagnosis because another was already in progress. To see the status of the Action for any event, click the **Action** tab of the event.

When the Diagnosis has completed, the Action Knowledge Script raises an event that identifies the location of the Vivinet Diagnostics `.dgv` file, which contains the results of the Diagnosis. In addition, if the Web management server and Report agent are installed on the computer that is running the Action, you can enable the **Use Report Agent settings** parameter, which integrates the Diagnosis results with other reports generated by the Report agent. The results are then easily accessible from the Web management server Report Binder and from the Operator Console's **Extensions > Report Viewer** function.

TIP: To allow this script to trigger a Diagnosis with Vivinet Diagnostics whenever a problem occurs, you need to disable or modify the "event collapsing" feature on the NortelCS_Alarms script. Event collapsing allows AppManager to suppress, or collapse, what it considers to be duplicate events. However, you will probably want Vivinet Diagnostics to diagnose a problem each time one occurs, even if it occurs between the same two targets. And you cannot do that if AppManager has collapsed all call quality events between the same targets into one event. Use the Advanced tab of the NortelCS_Alarms script to disable event collapsing, or at least to modify the 20-minute collapsing interval.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Output folder prefix	Enter a prefix for the output folder that is generated by the Diagnosis. The output folder then uses this prefix in the following naming convention: *Prefix_JobID_Phone1_Phone2_DateTime*. The default prefix is `Diag`. `Phone1` and `Phone2` are the IP addresses of the two Nortel phones being diagnosed. The `ComputerName` is the name of the computer being diagnosed.
Use Report Agent settings?	If set to yes, the diagnostic results are integrated into the AppManager Web management server (the Report Binder). The default is y.
Full path to root of output folders	Enter the full local or UNC path to the root of the directory in which you want to create the output folders. Make sure that the `NetIQmc` service (NetIQ AppManager Client Resource Monitor) is configured to run as a user that has access to the UNC path. The default setting of "local system" does not have access to the UNC path. Without access to the path, Vivinet Diagnostics will not be able to save a Diagnosis to the output folder. **NOTE:** This parameter is ignored if **Use Report Agent settings** is set to y.
Event Notification	
Severity when diagnosis successful	Enter a severity level, between 1 and 40, to indicate the importance of an event that is raised when the Diagnosis completes successfully. The default is 35.
Severity when error encountered	Enter a severity level, between 1 and 40, to indicate the importance of an event that is raised when an error prevents the Diagnosis from completing successfully. The default is 15.

11.4 DiagnoseVoIPQuality

Use this Action Knowledge Script to trigger NetIQ Vivinet Diagnostics to run a Diagnosis of voice quality between two phones or two endpoints. A Diagnosis is performed when a threshold is exceeded when you run any of the following Knowledge Scripts:

- **AvayaCM_CallQuality**. Vivinet Diagnostics can diagnose the problem when average MOS, average R-Value, average jitter, average latency, and average packet loss fall below or exceed their thresholds.
- **AvayaCM_PhoneQuality**. Vivinet Diagnostics can diagnose the problem when MOS, R-Value, jitter, latency, and packet loss fall below or exceed their thresholds during the data collection interval.
- CiscoCallMgr_CallQuality. Vivinet Diagnostics can diagnose the problem when jitter, latency, and percentage of lost data exceed their thresholds.
- CiscoCallMgr_CallFailures. Vivinet Diagnostics can diagnose the problem when the number of failed calls exceeds its threshold.

- **NortelCS2x_CallQuality**. Vivinet Diagnostics can diagnose the problem when end-of-call values for MOS and R-value fall below their thresholds, and when end-of-call values for jitter, latency, and packet loss exceed their thresholds.
- **NortelCS2x_PhoneQuality**. Vivinet Diagnostics can diagnose the problem when mid-call values for MOS and R-value fall below their thresholds, and when mid-call values for jitter, latency, and packet loss exceed their thresholds.
- **PhoneQuality_CiscoPhoneQuality**. Vivinet Diagnostics can diagnose the problem when the values for listening MOS and listening R-value fall below their thresholds, and when the values for average jitter, maximum jitter, and packet loss exceed their thresholds.
- VoIPQuality_CallPerf_<name of script>. Vivinet Diagnostics can diagnose the problem when MOS, R-factor, delay, jitter buffer loss, and percentage of lost data exceed their thresholds.

NOTE: This script is supported only when Vivinet Diagnostics 1.1 or later is installed. In addition, this script does not work when triggered by Knowledge Scripts other than those listed above.

When launched, Action_DiagnoseVoIPQuality performs the following steps:

- Verifies that Vivinet Diagnostics 1.1 or later is installed. If Vivinet Diagnostics 1.1 or later is not installed, this script raises an event indicating that Vivinet Diagnostics 1.1 is required.
- Verifies that you have elected to run this Action Knowledge Script based on an event raised by the running of an applicable script. If you choose to initiate this script based on some other Knowledge Script, this Action script raises an event indicating that the Action script cannot invoke Vivinet Diagnostics from the <*script name*> Knowledge Script.
- Determines the number of Diagnoses that need to be performed based on the parameters that you set in the above-mentioned scripts, as well as the thresholds that were reached or exceeded.
- Invokes Vivinet Diagnostics to perform the Diagnosis and generate an `.html` diagnostic report.
- If *Use Report Agent settings?* is disabled, generates a `default.rptIndex.xml` file and a small `default.htm` file that contains hyperlinks to the `.dgv` file and the Vivinet Diagnostic report.
- Upon completion of the Diagnosis, raises an event that contains the results of the Diagnosis. An event for a successful Diagnosis contains either a URL to the `default.htm` file (if *Use Report Agent settings?* is enabled) or the name of the computer and full directory path to the output files. An event for an unsuccessful Diagnosis contains an error message explaining why the Diagnosis was unsuccessful.

For a more in-depth discussion of the integration of Vivinet Diagnostics and AppManager, see the *User Guide for Vivinet Diagnostics*.

NOTE: Before running this script, you must first configure Vivinet Diagnostic with information about your VoIP setup. AppManager passes phone data, such as the calling party or the called party, to Vivinet Diagnostics, and then Vivinet Diagnostics starts the diagnosis using the phone data from AppManager and the VoIP setup information you configured in Vivinet Diagnostics. Be aware that AppManager will not raise an alert if the information about your VoIP setup is missing from Vivinet Diagnostics.

The following list covers what you must configure for Vivinet Diagnostics, based on your VoIP setup:

- Nortel Call Server Signaling Server (for Nortel phones)
- Cisco Call Manager (for Cisco phones)
- SNMP information (for all VoIP phones)

For more information, see the *User Guide for Vivinet Diagnostics*.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event severity when error	Set the severity level, between 1 and 40, to indicate the importance of an event in which a diagnosis does not complete successfully. The default is 15.
Event severity when successful	Set the severity level, between 1 and 40, to indicate the importance of an event in which a diagnosis completes successfully. The default is 35.
Output folder prefix	Enter a prefix for the output folder that is created by the Diagnosis. The output folder then uses this prefix in its naming convention as follows: *Prefix_JobID_ComputerName_DateTime*. **NOTE:** The *ComputerName* is the name of the CallManager computer or talker/listener computer. If a single event triggers multiple Diagnoses, a sequence number (0, 1, 2, 3, 4) is appended to the output folder name. The output folder contains the `.dgv` file, the diagnostic `.html` report file, and, if integrated with the report-enabled agent, a `default.rptIndex.xml` file, and a `default.htm` file that contains hyperlinks to both the `.dgv` file and the diagnostic `.html` report.
Use Report Agent settings?	Set to **y** to integrate the Diagnostics results into the AppManager Web management server (the Report Binder). The default is y.
Full path to root of output folders	Provide the full path to the root of where the output folders will be created. **NOTE:** Ignore this parameter if *Use Report Agent settings?* is set to **Yes**.
Maximum diagnoses	Specify a number between 1 and 5 to indicate the maximum number of Diagnoses that can be triggered by a single event. **NOTE:** This parameter is applicable only for events generated by the CiscoCallMgr_CallQuality and CiscoCallMgr_CallFailures scripts, where one event may identify multiple pairs of phones that indicate a problem.

11.5 DominoCommand

Use this Knowledge Script to issue a Domino command to a Domino Server. This script can enable you to run a Domino command as a corrective action in response to an event. For example, the Knowledge Script Domino_LogSniff monitors the Notes log database for specific messages or search strings. If you locate corruption in a database by running Domino_LogSniff, you can set that Knowledge Script to run the Action_DominoCommand Knowledge Script with the corrective Domino command or Domino agent you specify. The Domino command `Fixup`, for example, locates and repairs corrupted databases.

DominoCommand raises an event when the command is completed. The event message indicates whether or not the command was successful, and provides an explanation if the command fails.

NOTE: This Action can run only on the managed computer as a Managed Client Action. Select **MC** (Managed Client) on the Action tab of the Properties dialog box when enabling this Action.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Command	Specify the Domino command to be executed, using Domino command delimiters. For example, the default command, 0|#show task, includes the following elements: ◆ the partition number of the Domino server to which the command is being sent. If the computer contains only one instance of a Domino server, enter 0 ◆ the vertical bar, | ◆ the pound sign, # ◆ the command, show task Domino commands must follow this format.

11.6 DosCommand

Use this Knowledge Script to run a non-interactive DOS command when an event is raised. For example, use this script to run a batch command for virus scanning, disk backup, or logging an entry in a trouble-ticket system.

You can include arguments in the command string. This script can also test your command-line syntax.

Use this Knowledge Script to create a script file that contains a series of commands to diagnose or correct problems on a server you are monitoring. You can then have this Action launch your script file when an event is detected. For example, enter: `cmd /c \fixitscript.bat`.

To ensure the command runs successfully:

- Include the full path to the executable you want to run. For example, to issue a `Ping` command, enter a command similar to the following:

  ```
  cmd /c \ping.exe 164.210.210.1.
  ```

- Be sure that the command you want to run does not require any user input.
- To run this Action on the managed computer, select **MC (Managed Client)** as the Location on the Action tab of the Properties dialog box.
- Check whether the computer where you want to run a command or script file accepts commands from the management server you are using. This access is controlled through the `AllowDosCmd` registry key setting. By default, the `AllowDosCmd` key is set to `*` to allow all management servers to initiate DOS commands. To restrict the management servers that are allowed to run DOS commands, you can set this key to a comma-separated list of computer names. For example, `AllowDosCmd:REG_SZ:shasta,dynamo`.
- Verify that the AppManager Client Resource Monitor (`NetIQmc`) service account, whether it be the `LocalSystem` account or a user account, has permission to execute the command you want to run on the computer where you want the Action executed.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity -- Action failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the DosCommand job fails. The default is 5 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	

Parameter	How to Set It
Non-interactive DOS command	Specify the command to run. Do not enter a command that requires user input. The command you enter should take care of any input and output redirection or handling required. The default is `del\temp\yunk.txt`. **NOTE:** If the command you are entering includes quotation marks ("), enclose the quoted string in a second set of quotation marks. For example, if the DOS command is `net send "message"`, enter the following: `cmd /c net send ""message""`. You can use the following keywords in the command: ◆ `$ShortMsg$` (short event message) ◆ `$DetailMsg$` (detailed event message) ◆ `$Time$` (date and time of the event) ◆ `$JobID$` (ID of the job that raised the event) ◆ `$MachineName$` (name of the computer where the event was raised) ◆ `$Severity$` (severity of the event) ◆ `$KSName$` (name of the Knowledge Script that raised the event) ◆ `$ObjectName$` (name of the AppManager resource object where the event was raised) ◆ `$EventID$` (event ID) For `$ShortMsg$` and `$DetailMsg$,` you can use number and wildcard options to indicate specific portions of the text string to include. For example: ◆ `$DetailMsg$[5]` includes the fifth word of the detailed event message ◆ `$ShortMsg$[1-5]` includes the first through fifth words of the short message event ◆ `$DetailMsg$[*5]` includes the first through fifth words of the detailed event message ◆ `$ShortMsg$[5*]` includes the fifth through last words of the short event message If you do not enter a word specifier, AppManager returns the entire string. **Example** To print a detail message starting from the eighth word into `c:\temp\log.txt`, type the following command: `echo $DetailMsg$[8*] > c:\temp.log.txt`
Normal/Expected exit code	Set the normal/expected exit code for the DOS command you enter. The default is 0.

11.7 DumpTran

Use this Action Knowledge Script with selected SQL Knowledge Scripts (such as DataSpace, DBSpace, and LogSpace) to dump the transaction log of a database when an event is raised. For example, if the DBSpace Knowledge Script detects that the database space available has fallen below the threshold, you can use this Action to automatically dump the transaction log to free up space. Syntax and permission checking is handled by SQL Server.

When configuring this action, keep in mind:

- The Action can run only on the managed computer as a Managed Client Action. Be sure to select **MC** (Managed Client) as the Location on the Action tab of the Properties dialog box.
- This script requires an account with System Administrator privileges or dbo privileges to run. If you run this Action on SQL Server 7, the Dump Transaction can be done by a dbo or db_backup operator account. For more information about the permissions required for a Dump Transaction command, see your SQL Server documentation.
- This script requires a database name supplied by the SQL Knowledge Script to perform the dump. If the Knowledge Script that raises the event is running with the *Dynamically observe databases at each interval?*

  parameter enabled (so that it dynamically discovers database names at run time), the Action will fail. To use this Action, disable the *Dynamically observe databases at each interval?* parameter in the DataSpace, DBSpace, or LogSpace Knowledge Script.

This Action can only operate on a database whose recovery model is either Full or Bulk-Logged.

Setting Parameter Values

Set the following parameters as needed:

Parameter	**How to Set It**
SQL login	Specify the database user account used to run this Knowledge Script, for example, `sa`. You can run this Knowledge Script using other user accounts that have been set up in the SQL Server of the managed client and have been given permission to run SQL Knowledge Scripts through the AppManager Security Manager.
Truncate only?	Set to **y** to truncate the transaction log, without saving the truncated information to any location. If set to n, you must specify where the transaction log should be sent in the TO statement parameter. The default is n.
TO statement	If *Truncate only?* is disabled, enter a TO statement to specify where the truncated transaction log should be sent. The default is `to diskdump`.

11.8 ExtendedSNMPTrap

Use this Knowledge Script to send an extended SNMP trap message with AppManager event information to a specified list of computers. The event information includes the event severity level.

Each computer you specify must be able to receive SNMP trap messages on UDP port 162.

If you do not specify a value for any of the parameters, this Knowledge Script uses the corresponding value found in the registry under HKEY_LOCAL_MACHINE\Software: NetIQ\AppManager\4.0\NetIQmc\SNMPTRAP\Config.

For example, if you do not specify an object identifier in the OID field, the Knowledge Script checks the registry for the OID key entry: OID: REG_SZ: 1.3.6.1.4.1.1691.1.

When associating the ExtendedSNMPTrap Knowledge Script with a monitoring job, carefully choose the location of the action. Location options are available on the Action tab of the Properties dialog box

- When Location = MC, the trap will not include fields for Event Identifier, Repository Name, and Repository Server, because this information is not available on the AppManager agent.
- When Location = MS or Location = proxy, the trap will include fields for Event Identifier, Repository Name, and Repository Server. However, if many jobs are configured to send traps with the management server, performance on the management server computer may be adversely affected.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
List of computers to receive SNMP message	Provide the name of the computer to receive the SNMP trap message. The receiving port is port 162. To specify multiple recipients, separate computer names with commas and no spaces. For example, Nancy01,Finance03 If this field is left blank, the local host is the recipient by default.
Community string	Provide a valid SNMP community string. Leave this parameter blank to use the SNMP community string entered in AppManager Security Manager. If no SNMP community string is entered in Security Manager, the “public” SNMP community string is used by default.
Object identifier	Enter an object identifier in OID notation (for example, 1.2.3.456.78). If no value is entered, this script uses the OID notation entered in the following registry key: SOFTWARE\NetIQ\AppManager\4.0\NetIQmc\SNMPTRAP\Config
Specific trap number	Enter a trap number. The trap number can be specific to your application. If no value is entered, this script uses the trap number entered in the following registry key: SOFTWARE\NetIQ\AppManager\4.0\NetIQmc\SNMPTRAP\Config

11.9 IISContinueSite

Use this Knowledge Script to continue a paused IIS site. This script raises an event if the script is unable to continue a paused IIS site.

This Action can run only on the managed computer as a managed client Action. Be sure to select **MC** (managed client) as the Location on the Action tab of the Properties dialog box. This Action cannot run as a management server (MS) Action.

When you use this Action with a Knowledge Script that supports dynamic observation and you enable the *Dynamic observation* parameter, you can only run the Knowledge Script on one Web site at a time. If you disable dynamic observation, you can run the Knowledge Script on all Web sites.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if attempt to continue fails?	Set to **y** to raise an event if the attempt to continue a paused IIS site fails. The default is y.
Event severity when site cannot be continued	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a paused IIS site cannot be continued. The default severity level is 7 (red event indicator).

11.10 IISPauseSite

Use this Knowledge Script to temporarily pause an IIS site. This script raises an event if the IIS site cannot be paused.

This Action can only run on the managed computer as a managed client Action. Be sure to select **MC** (managed client) as the Location on the Action tab of the Properties dialog box. This Action cannot run as a management server (MS) Action.

When you use this Action with a Knowledge Script that supports dynamic observation and you enable the *Dynamically observe Web servers at each interval* parameter, you can only run this Knowledge Script on one Web site at a time. If you disable dynamic observation, you can drop the Knowledge Script on all Web sites.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if attempt to pause fails?	Set to **y** to raise an event if the attempt to pause an IIS site fails. The default is y.
Event severity when site cannot be paused	Set the event severity level, from 1 to 40, to indicate the importance of an event in which an IIS site cannot be paused. The default severity level is 7 (red event indicator).

11.11 IISRestartServer

Use this Knowledge Script to stop and then restart an IIS server. This script raises an event if the attempt to stop or restart a service fails or succeeds. Any services that are stopped when the job runs can also be detected and started.

When you use this Action with a Knowledge Script that supports dynamic observation and you enable the *Dynamically observe Web servers at each interval* parameter, you can only run this Knowledge Script on one Web site at a time. If you disable dynamic observation, you can run the Knowledge Script on all Web sites.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if attempt to restart fails or succeeds?	Set to **y** to raise an event if the IIS server cannot be restarted, or if the server is successfully restarted. The default is y.
Restart server?	Set to **y** to restart the server. The default is y.
Start all stopped services?	Set to **y** to start all of the services. The default is n.
Severity when restart...	Set the event severity level, from 1 to 40, to indicate the importance when the attempt to restart stopped services: **... fails**. Type a value that indicates the service is down and AppManager cannot restart it. The default is 10 (red event indicator). **... succeeds**. Type a value that indicates the service was down and AppManager successfully restarted it. The default is 20 (blue event indicator).

11.12 IISRestartSite

Use this Action Knowledge Script to shut down and restart an IIS site instance. This script raises an event if the IIS site cannot be shut down or restarted.

This Action can only run on the managed computer as a managed client action. Be sure to select **MC** (managed client) as the Location on the Action tab of the Properties dialog box. This Action cannot run as a management server (MS) Action.

When you use this Action with a Knowledge Script that supports dynamic observation and you enable the *Dynamically observe Web servers at each interval* parameter, you can only run this Knowledge Script on one Web site at a time. If you disable dynamic observation, you can run the Knowledge Script on all Web sites.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if attempt to shut down or restart fails?	Set to **y** to raise an event if the attempt to shut down or restart an IIS site instance fails. The default is y.

Parameter	How to Set It
Restart site after shutdown?	Set to **y** to restart an IIS site after it is shut down. The default is y.
Event severity when attempt to shut down or restart IIS site fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which an IIS site cannot be shut down or restarted. The default severity level is 7 (red event indicator).

11.13 MapiMail

Use this Knowledge Script to send a MAPI email message with AppManager event information to a specified list of recipients.

By default, the event information includes the computer name of the managed client and the event severity. You can select additional information to include. You can also construct a custom message to send to recipients.

This script raises an event if you select the custom message format but neglect to enter any text for the custom message body. Under these circumstances, the script continues to execute and uses the standard message format.

You can attach a file to the email message by entering the path to the file.

The email message is sent using the Microsoft MAPI mechanism. The recipients can be one or many MAPI clients.

NOTE: Because Microsoft tightened security in recent versions of Microsoft Outlook, the MapiMail script works only with Outlook 2000 and Outlook 2003 SP1. This script is not supported on the following versions of Outlook:

- Outlook 2003 without service packs
- Outlook 2003 SP2
- Outlook 2007
- Outlook 2010

As an alternative, consider using the SMTPMail Knowledge Script.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity -- Action warning	Set the severity level, from 1 to 40, to indicate the importance of an event in which the MapiMail job returns a warning. The default is 35 (magenta event indicator).
Event severity -- Action failure	Set the severity level, from 1 to 40, to indicate the importance of an event in which the MapiMail job fails. The default is 5 (red event indicator).
Severity Configuration	

Parameter	How to Set It
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
Profile name	Provide the profile name of the managed client, such as the default netiq account or an account set up specifically for the client. The profile must be an account with Mail capability.
List of recipients (in address book)	Provide the email address for the recipient of the message, using names in the address book. Separate multiple names with semicolons (;). For example: `Chris Lin;pat@bigcorp.com;gwest`. **NOTE:** Be sure the names you enter are not ambiguous. If the script cannot definitively identify the recipient, mail is not sent.
Full path to mail attachment	Provide the full path to the attachment you want to send. If you are not attaching a file, leave this field blank.
Message format	Select the format you want to use for the message sent by this script: ◆ **Standard** format generates a message based upon the selections you make from the *Standard Message Options* parameters. ◆ **Custom** format generates a message based upon the subject and message body you supply in the *Custom Message Options* parameters. The default is Standard.
Standard Message Options	
Include date/timestamp?	Select **Yes** to include the date/timestamp in the standard message. The default is unselected.
Include JobID?	Select **Yes** to include the job ID in the standard message. The default is unselected.
Include agent computer name?	Select **Yes** to include the name of the agent computer in the standard message (the computer hosting the agent that initiated the action). The default is Yes.
Include event severity?	Select **Yes** to include the severity of the event in the standard message. The default is Yes.
Include Knowledge Script name?	Select **Yes** to include the Knowledge Script name in the standard message (the Knowledge Script that initiated the Action). The default is unselected.
Include AppManager object name?	Select **Yes** to include the AppManager resource object name in the standard message (the AppManager resource object where the event was raised). The default is unselected.
Include AppManager event ID (only on MS Action)?	Select **Yes** to include the AppManager event ID in the standard message (possible only in cases when the Action is carried out by the management server). The default is unselected.

Parameter	How to Set It
Include event detail message?	Select **Yes** to include the event detail message. The default is unselected.
Custom Message Options	
Custom message subject	Provide the text you want to use for the custom message subject line.
Custom message body	Provide the text you want to include in your custom message. You can use the keywords listed below to indicate the information you want to include in the body of your custom message. Add a space before and after keywords to ensure that the keywords display properly. ◆ `$ShortMsg$` (short event message) ◆ `$DetailMsg$` (detailed event message) ◆ `$Time$` (date and time of the event) ◆ `$JobID$` (ID of the job that raised the event) ◆ `$MachineName$` (name of the computer where the event was raised) ◆ `$Severity$` (severity of the event) ◆ `$KSName$` (name of the Knowledge Script that raised the event) ◆ `$ObjectName$` (name of the AppManager resource object where the event was raised) ◆ `$EventID$` (event ID) For `$ShortMsg$` and `$DetailMsg$` you can use number and wildcard options to indicate specific portions of the text string to include. For example: ◆ `$DetailMsg$[5]` includes the fifth word of the detailed event message ◆ `$ShortMsg$[1-5]` includes the first through fifth words of the short message event ◆ `$DetailMsg$[*5]` includes the first through fifth words of the detailed event message ◆ `$ShortMsg$[5*]` includes the fifth through last words of the short event message If you do not enter a word specifier, AppManager returns the entire string. The following are examples of the types of messages you can construct using these keywords: ◆ Event from `$MachineName$`: The `$ShortMsg$[1-3]` has failed. The last command was `$DetailMsg$[4*]`. ◆ A severity `$Severity$` event has occurred! Call the owner of `$MachineName$` immediately!

11.14 Messenger

Use this Knowledge Script to use the Windows Messenger service to send a message containing AppManager event information to a specified computer.

By default, the event information includes the computer name of the managed client and the event severity. You can select additional information to include.

You can also construct a custom message to send to recipients.

This script raises an event if you select the custom message format but neglect to enter any text for the custom message body. Under these circumstances, the script continues to execute and uses the standard message format.

The destination computer must be running the Windows Messenger service. To send the message to multiple computers, enter a comma-separated list of computer names.

NOTE

- This Knowledge Script is not supported on Windows operating systems later than Windows Server 2003.
- If you are using this Knowledge Script to send a message from a Windows Server 2003 computer to a Windows NT 4 computer, the Messenger service must be running on both computers.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity -- Action warning	Set the severity level, from 1 to 40, to indicate the importance of an event in which the Messenger job returns a warning. The default is 35 (magenta event indicator).
Event severity -- Action failure	Set the severity level, from 1 to 40, to indicate the importance of an event in which the Messenger job fails. The default is 5 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
List of computers to receive message	Provide a computer name or click the Browse [...] button to select the recipient of the Messenger service message. To send a message to multiple recipients, enter a comma-separated list of computer names. For example: `QELAB,PORT1,Chris`. Each specified computer must be running the Messenger service.

Parameter	How to Set It
Message format	Select whether you want to use the standard message format or create a custom message. The default is Standard. Use the Standard message format if you want the message text to be generated by the Knowledge Script. Use the Custom message format if you want to create your own message.
Standard Message Options	
Include date/timestamp?	Select **Yes** to include the date and time of the event. The default is unselected.
Include JobID?	Select **Yes** to include the ID of the Knowledge Script job that raised the event. The default is unselected.
Include agent computer name?	Select **Yes** to include the name of the computer on which the event was raised. The default is Yes.
Include event severity?	Select **Yes** to include the event severity. The default is Yes.
Include Knowledge Script name?	Select **Yes** to include the name of the Knowledge Script that raised the event. The default is unselected.
Include AppManager object name?	Select **Yes** to include the name of the AppManager object where the event was raised. The default is unselected.
Include AppManager Event ID (only on MS action)?	Select **Yes** to include the event ID number when the Action is initiated by the AppManager management server. The default is unselected.
Include event detail message?	Select **Yes** to include the event detail message. The default is unselected.
Custom Message Options	

Parameter	How to Set It
Custom text (can include substitutions)	Provide the text you want to include in your custom message. You can use the keywords listed below to indicate the information you want to include in the body of your custom message. Add a space before and after keywords to ensure that the keywords display properly. ◆ `$ShortMsg$` (short event message) ◆ `$DetailMsg$` (detailed event message) ◆ `$Time$` (date and time of the event) ◆ `$JobID$` (ID of the job that raised the event) ◆ `$MachineName$` (name of the computer where the event was raised) ◆ `$Severity$` (severity of the event) ◆ `$KSName$` (name of the Knowledge Script that raised the event) ◆ `$ObjectName$` (name of the AppManager resource object where the event was raised) ◆ `$EventID$` (event ID) For `$ShortMsg$` and `$DetailMsg$` you can use number and wildcard options to indicate specific portions of the text string to include. For example: ◆ `$DetailMsg$[5]` includes the fifth word of the detailed event message ◆ `$ShortMsg$[1-5]` includes the first through fifth words of the short message event ◆ `$DetailMsg$[*5]` includes the first through fifth words of the detailed event message ◆ `$ShortMsg$[5*]` includes the fifth through last words of the short event message If you do not enter a word specifier, AppManager returns the entire string. The following are examples of the types of messages you can construct using these keywords: ◆ Event from `$MachineName$`: The `$ShortMsg$[1-3]` has failed. The last command was `$DetailMsg$[4*]`. ◆ A severity `$Severity$` event has occurred! Call the owner of `$MachineName$` immediately!
Retry count	Set the number of time this script attempts to send the message. The default is 5.

11.15 NotesMail

Use this Knowledge Script to send a mail message containing AppManager event information to one or more Lotus Domino/Notes email users. To use this script, you must have a Domino Notes server, and the Notes server must be on the computer initiating the Action, either on the management server or on the managed client.

By default, the event information includes the computer name of the managed client and the event severity. You can select additional information to include.

You can also construct a custom message to send.

This script raises an event if you select the custom message format but neglect to enter any text for the custom message body. Under these circumstances, the script continues to execute and uses the standard message format.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity -- Action warning	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the NotesMail job returns a warning. The default is 35 (magenta event indicator).
Event severity -- Action failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the NotesMail job fails. The default is 5 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Set the minimum event severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum event severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
List of recipients	Provide a list of recipients for the message, separated by commas (,) with no spaces, using the Notes username format (for example, `user/company`).
Sender name	Provide the mail sender name. It is displayed in the **From** field of the mail message that has the AppManager event information. The default is NetIQ AppManager.
Message format	Select the format you want to use for the message sent by this script: ◆ **Standard** format generates a message based upon the selections you make from the *Standard message options* parameters. ◆ **Custom** format generates a message based upon the subject and message body you supply in the *Custom message options* parameters. The default is Standard.

Parameter	How to Set It
Standard message options	
Include date/timestamp?	Select **Yes** to include the date/timestamp in the standard message. The default is unselected.
Include JobID?	Select **Yes** to include the job ID in the standard message. The default is unselected.
Include agent computer name?	Select **Yes** to include the name of the agent computer in the standard message (the computer hosting the agent that initiated the Action). The default is Yes.
Include event severity?	Select **Yes** to include the severity of the event in the standard message. The default is Yes.
Include Knowledge Script name?	Select **Yes** to include the Knowledge Script name in the standard message (the Knowledge Script that initiated the Action). The default is unselected.
Include AppManager object name?	Select **Yes** to include the AppManager resource object name in the standard message (the AppManager resource object where the event was raised). The default is unselected.
Include AppManager event ID (only on MS Action)?	Select **Yes** to include the AppManager event ID in the standard message (possible only in cases when the Action is carried out by the management server). The default is unselected.
Include event detail message?	Select **Yes** to include the event detail message. The default is unselected.
Custom message options	
Custom message subject	Provide the text you want to use for the custom message subject line.

Parameter	How to Set It
Custom message body	Provide the text you want to include in your custom message. You can use the keywords listed below to indicate the information you want to include in the body of your custom message. Add a space before and after keywords to ensure that the keywords display properly. ◆ $ShortMsg$ (short event message) ◆ $DetailMsg$ (detailed event message) ◆ $Time$ (date and time of the event) ◆ $JobID$ (ID of the job that raised the event) ◆ $MachineName$ (name of the computer where the event was raised) ◆ $Severity$ (severity of the event) ◆ $KSName$ (name of the Knowledge Script that raised the event) ◆ $ObjectName$ (name of the AppManager resource object where the event was raised) ◆ $EventID$ (event ID) For $ShortMsg$ and $DetailMsg$ you can use number and wildcard options to indicate specific portions of the text string to include. For example: ◆ $DetailMsg$[5] includes the fifth word of the detailed event message ◆ $ShortMsg$[1-5] includes the first through fifth words of the short message event ◆ $DetailMsg$[*5] includes the first through fifth words of the detailed event message ◆ $ShortMsg$[5*] includes the fifth through last words of the short event message If you do not enter a word specifier, AppManager returns the entire string. The following are examples of the types of messages you can construct using these keywords: ◆ Event from $MachineName$: The $ShortMsg$[1-3] has failed. The last command was $DetailMsg$[4*]. ◆ A severity $Severity$ event has occurred! Call the owner of $MachineName$ immediately!

11.16 NTEventLog

Use this Knowledge Script to write AppManager event information to the Windows event log. By default, the event is written to the Windows Application event log on the computer where the Action is initiated. You can select another event log where the event will be written, and you can select the event type: Error, Warning, or Information. You can also specify a custom event message or use the default message.

NOTE: This Action is performed on all physical nodes of a cluster when the NTEventLog Knowledge Script runs on a cluster server.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity -- Action warning	An event is raised when you select the custom message format but neglect to enter any text for the custom message body. Under these circumstances, the script continues to execute and uses the standard message format. Set the severity level, from 1 to 40, to indicate the importance of an event in which the NTEventLog job returns a warning. The default is 35 (magenta event indicator).
Event severity -- Action failure	Set the severity level, from 1 to 40, to indicate the importance of an event in which the NTEventLog job fails. The default is 5 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
Event log name	Select the log where the event message is written. The default is Application.
Event type	Select the type of event message. The default is Error.
Destination computer	Provide the name or IP address of the computer to whose log the event message is written. Leave this parameter blank for a local computer. You can also click **Browse [...]** to select from a list of computers in the same domain as the agent computer.
Event source	Specify a name for the source of the event. The default is AppManager.
Event category	Specify a numerical identifier for the event category. Enter a number from 0 to 32766. The default is 0.
Event ID	Specify a numerical ID for the event. Enter a number from 0 to 65535. The default is 260.

Parameter	How to Set It
Message format	Select the format you want to use for the message sent by this script: ◆ **Standard** format generates a message based upon the selections you make from the *Standard message options* parameters. ◆ **Custom** format generates a message based upon the subject and message body you supply in the *Custom message options* parameters. The default is Standard.
Standard Message Options	
Include date/timestamp?	Select **Yes** to include the date/timestamp in the standard message. The default is unselected.
Include JobID?	Select **Yes** to include the job ID in the standard message. The default is unselected.
Include agent computer name?	Select **Yes** to include the name of the agent computer in the standard message (the computer hosting the agent that initiated the Action). The default is Yes.
Include event severity?	Select **Yes** to include the severity of the event in the standard message. The default is Yes.
Include Knowledge Script name?	Select **Yes** to include the Knowledge Script name in the standard message (the Knowledge Script that initiated the Action). The default is unselected.
Include AppManager object name?	Select **Yes** to include the AppManager resource object name in the standard message (the AppManager resource object where the event was raised). The default is unselected.
Include AppManager event ID (only on MS Action)?	Select **Yes** to include the AppManager event ID in the standard message (possible only in cases when the Action is carried out by the management server). The default is unselected.
Include event detail message?	Select **Yes** to include the event detail message. The default is unselected.
Custom Message Options	

Parameter	How to Set It
Custom text (can include substitutions)	Provide the text you want to include in your custom message. You can use the keywords listed below to indicate the information you want to include in the body of your custom message. Add a space before and after keywords to ensure that the keywords display properly. ◆ `$ShortMsg$` (short event message) ◆ `$DetailMsg$` (detailed event message) ◆ `$Time$` (date and time of the event) ◆ `$JobID$` (ID of the job that raised the event) ◆ `$MachineName$` (name of the computer where the event was raised) ◆ `$Severity$` (severity of the event) ◆ `$KSName$` (name of the Knowledge Script that raised the event) ◆ `$ObjectName$` (name of the AppManager resource object where the event was raised) ◆ `$EventID$` (event ID) For `$ShortMsg$` and `$DetailMsg$` you can use number and wildcard options to indicate specific portions of the text string to include. For example: ◆ `$DetailMsg$[5]` includes the fifth word of the detailed event message ◆ `$ShortMsg$[1-5]` includes the first through fifth words of the short message event ◆ `$DetailMsg$[*5]` includes the first through fifth words of the detailed event message ◆ `$ShortMsg$[5*]` includes the fifth through last words of the short event message If you do not enter a word specifier, AppManager returns the entire string. The following are examples of the types of messages you can construct using these keywords: ◆ Event from `$MachineName$`: The `$ShortMsg$[1-3]` has failed. The last command was `$DetailMsg$[4*]`. ◆ A severity `$Severity$` event has occurred! Call the owner of `$MachineName$` immediately!

11.17 Page

Use this Knowledge Script to send a paging call with AppManager event information to one or more recipients. Paging systems and target recipients (individuals or groups) are defined in the `%systemroot%\netiqpage.ini` file on the computer where the action is targeted, either the

AppManager Management Server computer (MS) or the AppManager agent computer (MC). Before using this script, review and edit the `netiqpage.ini` file to identify the groups, phone numbers, and other parameters appropriate for your specific paging system.

By default, the event information includes the name of the agent computer and the event severity. You can select additional information to include.

You can also construct a custom message to send to recipients.

This script raises an event if you select the custom message format but neglect to enter any text for the custom message body. Under these circumstances, the script continues to execute and uses the standard message format.

Example of How this Script Is Used

Because each paging system has its own command-line syntax or API requirements, you need to define some information about the paging systems you are using in the `netiqpage.ini` file before using this Knowledge Script. The `netiqpage.ini` file specifies:

- Path to the paging server interface. For example, the path to the command-line program used to send the page.
- Command-line parameters or API syntax used to construct the page. For example, a specific paging interface may require a pager number, sender ID, or start time as command line arguments.
- Target group or profile names that contain the rules for contacting groups or individuals. For example, some paging systems allow an administrator to set up templates that define contact flow to control when specific groups can be reached by pager.

The following information is defined in the `netiqpage.ini` file in two sections:

- The `[system]` section, which defines the paging system, the path to the interface, and the command line parameters to be passed in from the `[group_name]` section depending on the *Name of the group to page* you enter in the Knowledge Script.
- The `[group_name]` sections, which define the details for target groups.

The following in an example of a `netiqpage.ini` file with definitions for three paging systems and two target groups, `QA` and `Sales`:

```
;;
;; sample netiqpage.ini file
;;
[system]
;; For the command line syntax for these paging systems:
;; first %s maps to the target_name [target]
;; second %s maps additional parameters [param]
;; third %s is the message passed in from the Knowledge Script
;;
attention=c:\AttnClient\attn -t %s %s %s
telalert=c:\usr\telalert\telalertc -c %s %s -m %s
hiplink=c:\hiplink\cms\hlclp -r:%s %s -m:'%s'  ;; msg in quotes
[QA]
    pageco1=hiplink
    target1=M
    param1=
    start_time1=00/00/00 00:00:00
    stop_time1=00/00/00 23:59:59
[Sales]
    pageco1=telalert
    target1=Pager
    param1= -n 4083031937
    start_time1=00/00/00 00:00:00
    stop_time1=00/00/00 23:59:59
    pageco2=telalert
    target2=Pager
    param2= -n 4083031937
    start_time2=5/12/98 00:00:00
    stop_time2=6/30/98 00:00:00
```

The Action_Page Knowledge Script uses the information defined in this file and the Action properties entered to construct the required command line to send the page. For example, if you set the *Name of the group to page*

parameter to `Sales`, AppManager sends a page to the Sales pager number (`408-303-1937`) using the `telalert` paging system.

Defining a Paging Schedule

Within the `netiqpaqe.ini` file, you can set a paging start time and end time for each person or group. This allows you to define specific periods when the individuals in a group can be paged. For example, if you have a Tech Support group with two employees who can be paged any day of the week between the hours of midnight and 8:00 a.m. and one employee who can be paged at any hour during specific dates, you might create entries similar to the following in the `netiqpaqe.ini` file:

```
[TechSupport]
    pageco1=telalert
    target1=Blake                    // Blake can be paged
    param1= -n 4083031937            // between 12:00 a.m.
    start_time1=00/00/00 24:00:00    // and 8:00 a.m.
    stop_time1=00/00/00 08:00:00     // (no start date or
                                     // end date)
    pageco2=telalert
    target2=Andy                   // Andy has the same
    param2= -n 4084551037          // schedule as Blake
    start_time2=00/00/00 24:00:00
    stop_time2=00/00/00 08:00:00
    pageco3=telalert
    target3=Alex                   // Alex can be paged any
    param3= -n 4156542200          // hour from midnight
    start_time3=7/12/98 00:00:00   // July 12, 1998 until
    stop_time3=7/30/98 10:30:00    // 10:30a.m. July 30
```

Both the `start_time` and `stop_time` parameters consist of two parts—the date and time. If you do not want to specify a start date or an end date, set the first part of the appropriate parameter to `00/00/00` (as illustrated with `Blake` and `Andy` in the example above). If you do not want to specify a start time or an end time, set the second part of the appropriate parameter to `00:00:00` (as illustrated with `Alex` in the example).

You cannot use the `start_time` and `stop_time` parameters to set up weekly scheduling. You can only define scheduling profiles or templates using parameters associated with your paging system. For example, if your paging system supports a `-s schedule_profile` command-line parameter, you can include this in the netiqpaqe.ini file as you do other parameters. For example:

```
[system]
page_app=c:\PageSysClient\sendpage -t %s -n %s %s -m %s
[WeekdayCrew]
    pageco1=page_app
    target1=scott
    param1= -n 4083031937 -s weekday_profile
                             // name of a template that
                             // allows paging Mon-Fri
    start_time1=00/00/00 00:00:00
    stop_time1=00/00/00 00:00:00
```

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity -- Action warning	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the Page job returns a warning. The default is 35 (magenta event indicator).
Event severity -- Action failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the Page job fails.
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
Name of the group to page (in netiqpage.ini)	Provide the name of the individual or group to receive this page. Valid names are the [group_name] sections you defined in the `netiqpage.ini` file.
Send a test page to a file?	Select **Yes** to send a test page to a file. The default is unselected.
Full path to test page file	Provide the full path to the file where you want to send your test page. The default is `c:\page.log`.
Message format	Select whether you want to use the standard message or create a custom message. The default is Standard. Use the Standard message format if you want the message text to be generated by the Knowledge Script. Use the Custom message format if you want to create your own message.
Standard Message Options	
Include date/timestamp?	Select **Yes** to include the date and time of the event. The default is unselected.

Parameter	How to Set It
Include JobID?	Select **Yes** to include the ID of the Knowledge Script job that raised the event. The default is unselected.
Include agent computer name?	Select **Yes** to include the name of the computer on which the event was raised. The default is Yes.
Include event severity?	Select **Yes** to include the event severity with the page. The default is Yes.
Include Knowledge Script name?	Select **Yes** to include the name of the Knowledge Script that raised the event. The default is unselected.
Include AppManager object name?	Select **Yes** to include the name of the AppManager object where the event was raised. The default is unselected.
Include AppManager event ID (MS Action only)?	Select **Yes** to include the event ID number when the Action is initiated by the AppManager management server. The default is unselected.
Include event detail message?	Select **Yes** to include the text of the event detail message with the page. The default is unselected.
Custom Message Options	
Custom message	Provide the message, up to 255 characters, you want to send with the page. If you do not specify a message, AppManager constructs a default message including the name of the agent computer and the severity level of the event.

11.18 RebootSystem

Use this Knowledge Script to shut down and restart a computer when an event is raised. This Action can run only on the managed computer as a Managed Client Action; select *Managed Client Action* in the Knowledge Script Properties dialog box.

To run this Knowledge Script, you need to be identified as a user with administrator privileges or have been granted permission to use this Action by the AppManager administrator.

This Knowledge Script also requires a registry setting under `HKEY_LOCAL_MACHINE\Software`: `NetIQ\AppManager\4.0\NetIQmc\Security`.

By default, the `AllowReboot` registry key is set to `null` to prevent *any* management servers from rebooting clients. If you have administrative privileges, you can change the registry settings to specify individual management servers or all management servers (using the wildcard *). For example: `AllowReboot:REG_SZ:mktg02;salesNA;190.12.1.28`.

You can use the NTAdmin_RegistrySet Knowledge Script to modify the registry key with the appropriate management server computer names.

Because of an Exchange mechanism, this Knowledge Script takes some time to reboot a computer if any Exchange service is running on that computer. Before using this Action, check whether any Exchange service is running on the target computers. If any Exchange service is running:

- Consider stopping the service before starting a Knowledge Script job that uses this Action.
- Consider whether you want to use this Action Knowledge on the selected computer.
- Allow for a longer-than-normal shutdown and reboot period.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Restart computer after shutdown?	Set to **y** to automatically restart the computer after shutting down. The default is y.
Force applications with unsaved changes to be closed?	Set to **y** to force any open applications to close when shutting down. In most cases, unsaved changes will be lost. The default is y.
Message to be displayed in the shutdown dialog box	Provide the message you want displayed in the shutdown dialog box. For example, if you are forcing applications to close, you may want the shutdown message to include a warning that unsaved changes may be lost.
Number of seconds to display shutdown dialog box	Specify the number of seconds you want the shutdown dialog displayed. The default is 60 seconds.
Number of hours to wait between restarts	Specify the number of hours to wait after the last restart before attempting another reboot. This parameter allows you to prevent the Knowledge Script job from continuously shutting down and rebooting a computer. The default is 4 hours.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of the event when this Action Knowledge Script encounters problems in shutting down or restarting the computer. The default is 7 (red event indicator).

11.19 RestartServices

Use this Knowledge Script to stop and restart Windows services. Enter the services you want to stop and restart as a comma-separated list.

Enable the *Restart dependent services?* parameter to restart services that depend on the ones you stopped. By default, this script restarts dependent services.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity -- Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the RestartServices job fails. The default is 10 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.

Parameter	How to Set It
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
List of services	Provide a comma-separated list of the services you want to stop and restart.
Service start/stop delay	Set the number of seconds to wait between stopping and restarting a service. The default is 30.
Restart dependent services?	Select **Yes** to restart services that depend on the ones you stopped. The default is Yes. For example, if you stop the `MSSQLSERVER` service, the dependent service `SQLSERVERAGENT` is also stopped. If you select **Yes**, the `MSSQLSERVER` service will be restarted, and then the `SQLSERVERAGENT` service will also be restated. If you deselect this check box, any dependent services of the service you specified are stopped and not restarted.

11.20 RunDiscoveryNetworkDevice

Use this Knowledge Script to discover network device resources as a result of an event raised by the NetworkDevice_Device_Uptime Knowledge Script.

In the event of a network device reboot, you can set the Device_Uptime Knowledge Script to run this Action to rediscover network device resources on the rebooted device.

Prerequisite

NetIQ Object Linking and Embedding (`NetIQOLE`) must be registered on the computer on which this script runs. `NetIQOLE` is an automation object that allows AppManager to be run from a command-line. For more information, see the *Administrator Guide for AppManager* at the NetIQ AppManager Documentation Web site.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
View name to use for KS filter	Select the view by which you want to filter the list of Knowledge Scripts to choose from. For example, select Master to include all Knowledge Scripts.
Show Report scripts?	Set to **y** to include Report scripts in the list of scripts to choose from.
Show Discovery scripts?	Set to **y** to include Discovery scripts in the list of scripts to choose from.
Show Admin scripts?	Set to **y** to include Admin scripts in the list of scripts to choose from.

Parameter	How to Set It
First KS to run	From the filtered list of Knowledge Scripts, select the first script that you want to run in response to an error. The script will run using all of the default parameter values unless you select and enter information in the *Parameter to pass* and *Values for parameters* parameters.
Parameter to pass	Select the parameters whose default values you want to change. Leave this field blank if you want to use the default values. If you select parameters in this field, you must also enter values for the *Values for parameters* parameter.
Values for parameters	Specify the values for the parameters that you want to change. Separate the values with a comma. For example, enter `y,y,20`. Leave this field blank if you leave *Parameters to pass* blank.
Second KS to run	From the filtered list of Knowledge Scripts, select the second script that you want to run in response to an error. The script will run using all of the default parameter values unless you select and enter information for *Parameter to pass* and *Values for parameters*.
Parameters to pass	Select the parameters whose default values you want to change. Leave this field blank if you want to use the default values. If you select parameters in this field, you must then enter values for the *Values for parameters* parameter.
Values for parameters	Enter the values for the parameters that you want to change. Separate the values with a comma. For example, enter `y,y,20`. Leave this field blank if you leave the *Parameters to pass* parameter blank.
Third KS to run	From the filtered list of Knowledge Scripts, select the third script that you want to run in response to an error. The script will run using all of the default parameter values unless you select and enter information for *Parameter to pass* and *Values for parameters*.
Parameters to pass	Select the parameters whose default values you want to change. Leave this field blank if you want to use the default values. If you select parameters in this field, you must also enter values for the *Values for parameters* parameter.
Values for parameters	Specify the values for the parameters that you want to change. Separate the values with a comma. For example, enter `y,y,20`. Leave this field blank if you leave *Parameters to pass* blank.
Use trusted connection?	Set to y to use the credentials of the `netiqmc` service to connect to the repository server. The default is n. If you accept the default, then you must enter a *User ID* and *Password*.
QDB Server	Specify the name of the repository server. Leave this field blank if the repository server is the local server.
User ID	Specify the user id for an un-trusted connection to the repository server. You must enter a user ID if you disable the *Use trusted connection?* parameter.
Password	Specify the password for an un-trusted connection to the repository server. You must enter a password if you disable the *Use trusted connection?* parameter.
Run this KS once	Set to y to schedule this script to run once, thereby overriding the default schedule of the scripts you have selected. If you disable this parameter, the scripts you selected will run according to the default.

Parameter	How to Set It
Machine to run KS on	Specify the name of the server on which you want to run the scripts you selected. Leave this field blank if the server is the same as the server of the parent job that raised the event.

11.21 RunKS

Use this Knowledge Script to run up to three other Knowledge Scripts. You can also specify parameter settings for the Knowledge Scripts if you do not want to use the default settings.

This script uses the login credentials of the `NetIQmc` service (the AppManager agent) on the management server computer.

Prerequisite

NetIQ Object Linking and Embedding (`NetIQOLE`) must be registered on the computer on which this script runs. `NetIQOLE` is an automation object that allows AppManager to be run from a command-line. For more information, see the *Administrator Guide for AppManager* at the NetIQ AppManager Documentation Web site.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity -- Action failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the RunKS job fails. The default is 5 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
Knowledge Script Configurations	
View name to use for Knowledge Script filter	Select an Operator Console view and computer by which you can filter the list of available Knowledge Scripts. The default is Master. Selecting an Operator Console view limits the list of available Knowledge Scripts to those visible in that view. Selecting a computer further limits the list of available Knowledge Scripts to ones that can run on that computer.
Show Report scripts?	Select **Yes** to include Report scripts in the list of available Knowledge Scripts. The default is unselected.

Parameter	How to Set It
Show Discovery scripts?	Select **Yes** to include Discovery scripts in the list of available Knowledge Scripts. The default is unselected.
Show Administrator scripts?	Select **Yes** to include all Administrator scripts (for example, AMAdmin scripts), in the list of available Knowledge Scripts. The default is unselected.
First Knowledge Script to run	Select the first Knowledge Script run by this Action script.
Parameters to pass	Select the parameters whose default settings you want to change (for example, a different threshold value). To change the parameters you selected, select **CLEAR SELECTION** in the Select a Parameter dialog box, then click **Finish**. Once AppManager clears the previous parameters, click Browse [...] again and select new parameters for which you want to change the default settings. The browser for selecting parameters lists both the parameter name as it appears in the code for the script (for example, `TH_Physical`) and the description of the parameter as it appears in the Values tab of the Knowledge Script Properties dialog box (for example, `Maximum physical memory threshold`).

Values for parameters	Provide a comma-separated list of the values for the parameters you have selected in an order that mirrors the parameter selection.
	For example, if you selected the following threshold parameters from NT_MemUtil for monitoring memory use:
	`TH_Physical,TH_Virtual,TH_Paging`
	then you would enter their values as follows:
	`95,95,95`
	NOTE: The value you enter must be in the format expected by the Knowledge Script. For example:
	◆ If the possible values for a parameter are numbers from 0-100, you must enter a number in that range.
	◆ If the possible values are **y** or n, then you must enter y or n.
	One exception to this is where a Knowledge Script uses a selected check box to specify a **Yes** value and a cleared check box to specify a **No** value. In this case, the underlying values are y or n, and you must enter y or n.

Second Knowledge Script to run	Select a second Knowledge Script run by this Action script.

Parameter	How to Set It
Parameters to pass (comma-separated)	Select the parameters whose default settings you want to change (for example, a different threshold value). To change the parameters you selected, select **CLEAR SELECTION** in the Select a Parameter dialog box, then click **Finish**. Once AppManager clears the previous parameters, click Browse [...] again and select new parameters for which you want to change the default settings. The browser for selecting parameters lists both the parameter name as it appears in the code for the script (for example, `TH_Physical`) and the description of the parameter as it appears in the Values tab of the Knowledge Script Properties dialog box (for example, `Maximum physical memory threshold`).

Values for parameters	Provide a comma-separated list of the values for the parameters you have selected in an order that mirrors the parameter selection.
	For example, if you selected the following threshold parameters from NT_MemUtil for monitoring memory use:
	`TH_Physical,TH_Virtual,TH_Paging`
	then you would enter their values as follows:
	`95,95,95`
	NOTE: The value you enter must be in the format expected by the Knowledge Script. For example:
	◆ If the possible values for a parameter are numbers from 0-100, you must enter a number in that range.
	◆ If the possible values are y or n, then you must enter y or n.
	One exception to this is where a Knowledge Script uses a selected check box to specify a **Yes** value and a cleared check box to specify a **No** value. In this case, the underlying values are y or n, and you must enter y or n.

Third Knowledge Script to run	Select a third Knowledge Script run by this Action script.
Parameters to pass	Select the parameters whose default settings you want to change (for example, a different threshold value). To change the parameters you selected, select **CLEAR SELECTION** in the Select A Parameter dialog box, then click **Finish**. Once AppManager clears the previous parameters, click Browse [...] again and select new parameters for which you want to change the default settings. The browser for selecting parameters lists both the parameter name as it appears in the code for the script (for example, `TH_Physical`) and the description of the parameter as it appears on the **Values** tab of the Knowledge Script Properties dialog box (for example, `Maximum physical memory threshold`).

Parameter	How to Set It

Values for parameters	Provide a comma-separated list of the values for the parameters you have selected in an order that mirrors the parameter selection.
	For example, if you selected the following threshold parameters from NT_MemUtil for monitoring memory use:
	`TH_Physical,TH_Virtual,TH_Paging`
	then you would enter their values as follows:
	`95,95,95`
	NOTE: The value you enter must be in the format expected by the Knowledge Script. For example:
	◆ If the possible values for a parameter are numbers from 0-100, you must enter a number in that range.
	◆ If the possible values are y or n, then you must enter y or n.
	One exception to this is where a Knowledge Script uses a selected check box to specify a **Yes** value and a cleared check box to specify a **No** value. In this case, the underlying values are y or n, and you must enter y or n.

Run this Knowledge Script once?	Select **Yes** to run the specified Knowledge Scripts only once regardless of their default schedules. The default is Yes.
Computer to run Knowledge Script	Specify the name of the computer on which the Knowledge Scripts will run, or click **Browse [...]** to select a computer in the Computer Browser dialog box. If you leave this parameter blank, the scripts will run on the same computer as the Knowledge Script that initiated the Action.
AppManager Repository Configuration	
Use trusted connection?	Select **Yes** to use a trusted connection when running the Knowledge Scripts. The default is Yes. If you use a trusted connection, the Knowledge Scripts run under the logon account used by the `netiqmc` service on the computer running the Knowledge Script that initiated the Action. If you do not use a trusted connection, the Knowledge Scripts run with the username and password specified in the parameters below.a
Repository server	Provide the name of the SQL Server used for the AppManager repository that will manage the Knowledge Script jobs. If you leave this value blank, the local computer name is used (the local computer is the one running the Knowledge Script that initiated the Action).
Repository database	Provide the name of the AppManager repository that will manage the Knowledge Script jobs. If you leave this value blank, the default repository name, `QDB`, is used.
User ID (non-trusted connection)	Provide the username for the account under which the Knowledge Scripts will run (if you disabled the *Use trusted connection?* parameter).

Parameter	How to Set It
Password (non-trusted connection)	Provide the password for the account under which the Knowledge Scripts will run (if you disabled the *Use trusted connection?* parameter).

11.22 RunPowerShell

Use this Knowledge Script to run a non-interactive Microsoft Windows PowerShell cmdlet or PowerShell script (`.PS1` file) when an event is raised by another Knowledge Script. You can also use this script to run any command that can be run from a Windows PowerShell command prompt, such as `dir c:\temp`. PowerShell accepts commands in cmdlet, `.PS1`, and Windows `cmd.exe` formats.

The Action_RunPowerShell script makes a number of callback and helper functions available to the PowerShell commands or scripts being run. For more information, see Appendix A, “Using PowerShell Callback and Helper Functions” in the management guide.

Prerequisites

- Microsoft Windows PowerShell version 1.0 or later
- Microsoft .NET Framework 3.0 or later
- AppManager for Windows version 7.6 or later

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Event Notification	
Event Severity - Action failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the RunPowerShell job fails. The default is 5.
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action Knowledge Script. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action Knowledge Script. The default is 40.
Action	

Parameter	How to Set It

PowerShell command to run	Provide the PowerShell scripts, cmdlets, or code blocks you want to run. Do not provide a command that requires user input. The command should take care of any required input and output redirection or handling. You can string multiple commands together. You can use the following keywords in a command:
	◆ `$ShortMsg$` (short event message)
	◆ `$DetailMsg$` (detailed event message)
	◆ `$Time$` (date and time of the event)
	◆ `$JobID$` (ID of the job that raised the event)
	◆ `$MachineName$` (name of the computer where the event was raised)
	◆ `$Severity$` (severity of the event)
	◆ `$KSName$` (name of the Knowledge Script that raised the event)
	◆ `$ObjectName$` (name of the AppManager resource object where the event was raised)
	◆ `$EventID$` (event ID)
	For `$ShortMsg$` and `$DetailMsg$`, you can use number and wildcard options to indicate specific portions of a text string to include. If you do not include an option, AppManager returns the entire text string. For example:
	◆ `$DetailMsg$[5]` includes the fifth word of the detailed event message
	◆ `$ShortMsg$[1-5]` includes the first through fifth words of the short event message
	◆ `$DetailMsg$[*5]` includes the first through fifth words of the detailed event message
	◆ `$ShortMsg$[5*]` includes the fifth through last words of the short event message
	For example, the following string runs the `echo` command in PowerShell and prints the detailed event information, starting from the eighth word, into the `log.txt` file on the agent computer:
	`Echo $DetailMsg$[8*] > c:\temp\log.txt`
	Hint If the command you are entering includes quotation marks ("), enclose the quoted string within a second set of quotation marks. For example:
	`Send-MailMessage -From me@mycompany.com -To you@yourcompany.com -Subject ""Hello!"" -SmtpServer smtp.mycompany.com`

11.23 RunSql

Use this Knowledge Script to run SQL statements or stored procedures. You can enter the SQL statements to run, or you can supply the full path to a file from which to load SQL statements.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if SQL statement succeeds?	Set to **y** to raise an event when the SQL statement is successful. The default is y. An event is always raised if an error occurs.
SQL Server name and instance if applicable	You can specify a SQL Server or a SQL Server and a named instance. Use the form “`SQL\Instance`”. By default, the default SQL Server instance on the computer where the Action runs is used.
SQL login	Specify the database user account that will run the SQL statements (for example, `sa)`. However, it is also possible to run this Knowledge Script using other user accounts that have been set up in the SQL Server of the managed client and been given permission to run SQL Knowledge Scripts through AppManager Security Manager. **NOTE:** In general, permission to run specific SQL commands and statements is derived from the permissions granted to the login account you are using to run this Knowledge Script. However, the dbcc command can only be run by: ◆ `dbo` account (SQL Server 6.x or 7) ◆ `db_backup operator` account (SQL Server 7)
Load SQL script from a file?	Set to **y** to have SQL statements read from a file. Set to **n** to enter the SQL statements in the SQL Statements field. The default is n.
Full path to SQL script file	If you are entering SQL statements from a file, provide the complete file path (for example, `F:\netiq\Sample.sql`). If the NetIQmc service is running as a system account, do not provide a path in the form of `\\machine\dir\Sample.sql`
SQL statement	Specify the T-SQL statement to be executed. The default is `sp_who2`. **Tip** Unless you are entering very simple queries, typing SQL statements into this field may be error-prone. Use the *Load SQL Script from a file* parameter to read T-SQL statements from a file. Or, if you have an AppManager Developer’s license, you can check the Knowledge Script out of the repository, use the script editor to paste the desired SQL statements into the SQL Statement field, then check the modified Knowledge Script back in.
Save query results to a file?	Set to y to save query results to an external file. The default is n. **NOTE:** Only the first 100 lines are written to the external file.
Full path to results file	Specify a full path to the file where you want query results saved.
Severity when SQL Statement fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the SQL statement fails to run. The default is 10 (red event indicator).
Severity when SQL Statement succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the SQL statement runs successfully. The default is 20 (yellow event indicator).

11.24 SendReportToPrinter

Use this Knowledge Script to send a report to the printer that acts as the default printer for the managed client computer where the Report agent is running.

When running this script, consider the following:

- Install the Report agent and the Windows Management Server on the same computer.
- On the computer where the Report agent is installed, configure the managed client to run as `user`. Ensure that this user account has a default printer configured.
- Configure the `URLMapping` registry key to change the default output path of the Report agent to `<hostname>\AMReports\report`. Use the AMAdmin_SetReportPaths Knowledge Script to change the output path.

  NOTE: You can also manually configure the output path of the Report agent. At a command prompt, enter regedit. In the Registry Editor window, navigate to `NetIQ\Common\AMREPORTS`. In the right panel, right-click on **URLMapping** and select **Modify**. In the Edit String dialog box, enter `<hostname>\AMReports\report` in the **Value data** field, then click **OK**.

- To use this Action when you are configuring report properties, click the Action tab, click New, select SendReportToPrinter, and then change the Location to MC.

Configuring the Managed Client to Run As User

Before you can use SendReportToPrinter, you must create a user for the NetIQ Client Resource Monitor (`NetIQmc`, also called the managed client) service on the computer where the Report agent is installed. The log-on type for this service is probably `LocalSystem` — but you cannot enable the printer function from a `LocalSystem` logon account.

To create a user for the managed client:

1. From the Windows Control Panel of the computer you want to configure, double-click Administrative Tools, and then double-click Services.
2. Right-click NetIQ AppManager Client Resource Monitor, and select Properties.
3. Click the Logon tab. If `Local System account` is selected, select This account and enter your user ID and password in the appropriate fields.
4. Ensure the user ID and password have permission to access the AppManager repository.
5. Click OK. In the Services window, notice that the **Logon As** information for NetIQ AppManager Client Resource Monitor has changed from `LocalSystem` to your username.
6. In the Services window, right-click NetIQ AppManager Client Resource Monitor and select Restart. Once the service stops and restarts, it is ready to accept printer requests.

Using SendReportToPrinter to Print a PDF

You can use the SendReportToPrinter Action to print your report as a `PDF` (Portable Document Format) file. You must have PDF conversion software installed on the Report agent computer.

To print a report as a PDF:

1. Install your PDF conversion software on the Report agent computer. Refer to your PDF software documentation for details about installation and preferences.
2. On the Report agent computer, set the PDF computer as the default printer.

3. Configure your report properties.
4. On the Action tab, click **New**, select **SendReportToPrinter**, and then change **Location** to **MC**.
5. Generate your report. Look for the PDF in the default location specified by your conversion software.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if job succeeds?	Set to **y** to raise an event if the process is successful. The default is n.
Event severity when...	Set the event severity level, from 1 to 40, to indicate the importance of the event when: ◆ ... job succeeds. The default is 25 (blue event indicator). ◆ **... job fails**. The default is 5 (red event indicator).

11.25 SMTPMail

Use this Knowledge Script to send an SMTP mail message with AppManager event information to a list of one or more mail recipients.

By default, the event information includes the computer name of the monitored computer and the event severity. You can include additional information by enabling the appropriate parameters.

To override the default subject and body for the email message, select *Custom Message Options* parameters.

This script raises an event if you select the custom message format but neglect to enter any text for the body of the custom message. Under these circumstances, the script continues to run using the standard message format.

If you enter an invalid ending word number, for example, if an event detail message has 10 words in it, and the user specifies word 15, the entire short message or detail message for is returned. For example:

`$ShortMsg$[5-15]` or `$DetailMsg$[5-15]`

If the event message had 15 or more words in it, SMTPMail will return a message with words 5 through 15 in the message.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if SMTP server is not accessible?	Select **Yes** to raise an event if the SMTP server cannot be reached. The default is Yes.
Event severity -- SMTP server not accessible	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the SMTP server cannot be reached. The default is 35 (magenta event indicator).
Event severity -- Action failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the SMTPMail job fails. The default is 5 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Specify the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Specify the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
List of recipient email addresses	Provide the full email address for each recipient of the message. Use semicolons (;) to separate multiple recipient addresses. For example: `chris@abc.com;pat@def.com;jw@abc.com`. **NOTE:** The following characters are invalid in this parameter: `/ \ [ ] : | = , * ? < >`
Sender's email address	Provide the email address of the person sending the message. **Notes** ◆ In Microsoft Exchange Server 2007 environments, specify the sender name in SMTP format: `<name>@<domain>`. ◆ The following characters are invalid in this parameter: `/ \ [ ] : | = , * ? < >`
SMTP Server	
SMTP server name	Provide the host name or IP address of your SMTP server. The default is `inet01`.
SMTP port	Set the port number for your SMTP server. The default is 25.
Transmit message using Transport Layer Security (TLS)?	Select **Yes** to transmit the message using the Transport Layer Security. The default is unselected.
Ping SMTP server before sending mail message?	Select **Yes** to ping the SMTP server to verify TCP connectivity before attempting to send an email to your listed recipients. The default is Yes.
Message Content	

Parameter	How to Set It
Message format	Select the format you want to use for the message sent by this script: ◆ **Standard** format generates a message based upon the selections you make from the *Standard Message Options* parameters. ◆ **Custom** format generates a message based upon the subject and message body you supply in the *Custom Message Options* parameters. The default is Standard.
Standard Message Options	
Include date/timestamp?	Select **Yes** to include the date/timestamp in the standard message. The default is unselected.
Include JobID?	Select **Yes** to include the job ID in the standard message. The default is unselected.
Include agent computer name?	Select **Yes** to include the name of the agent computer that initiated the Action in the standard message. The default is Yes.
Include event severity?	Select **Yes** to include the severity of the event in the standard message. The default is Yes.
Include Knowledge Script name?	Select **Yes** to include the name of the Knowledge Script that initiated the Action in the standard message. The default is unselected.
Include AppManager object name?	Select **Yes** to include the name of the resource object where the event was raised in the standard message. The default is unselected.
Include AppManager event ID?	Select **Yes** to include the AppManager event ID in the standard message, possible only in cases when the Action is carried out by the management server. The default is unselected.

Include event detail message?	Select **Yes** to include the event detail message. The default is unselected.
	NOTE: In Microsoft Outlook 2003 SP2 environments, the detail message information may be unformatted and presented in one string of text, such as in the following example:
	`Detail Message: Memory Utilization: Top 10 Consuming`
	`Processes ==========================================`
	`Process Name: Rtvscan Process ID (PID): 1684 Utilization`
	`(KB): 51,528.00   Process Name: NetIQmc Process ID (PID):`
	`3424 Utilization (KB): 23,956.00   Process Name: svchost#4`
	`Process ID (PID): 832 Utilization (KB): 17,784.00`

Custom Message Options	
Custom message subject	Provide the text you want to use for the custom message subject line. **NOTE:** If you select **Custom** in the *Message Format* parameter, yet fail to specify anything in this parameter, the selected *Standard Message Options* parameters are used.

Parameter	How to Set It
Custom message body	Provide the text you want to include in your custom message. **NOTE:** If you select **Custom** in the *Message Format* parameter, yet fail to specify anything in this parameter, the standard message body is used instead. You can use the keywords listed below to indicate the information you want to include in the body of your custom message. Add a space before and after keywords to ensure that the keywords display properly. ◆ `$ShortMsg$` (short event message) ◆ `$DetailMsg$` (detailed event message) ◆ `$Time$` (date and time of the event) ◆ `$JobID$` (ID of the job that raised the event) ◆ `$MachineName$` (name of the computer where the event was raised) ◆ `$Severity$` (severity of the event) ◆ `$KSName$` (name of the Knowledge Script that raised the event) ◆ `$ObjectName$` (name of the AppManager resource object where the event was raised) ◆ `$EventID$` (event ID) ◆ `$tab$` inserts four whitespace characters in the message body ◆ `$lf$` inserts a line feed in the message body ◆ `$crlf$` inserts a carriage-return line feed in the message body ◆ `$cr$` inserts a carriage-return in the message body For `$ShortMsg$` and `$DetailMsg$` you can use number and wildcard options to indicate specific portions of the text string to include. For example: ◆ `$DetailMsg$[5]` includes the fifth word of the detailed event message ◆ `$ShortMsg$[1-5]` includes the first through fifth words of the short message event ◆ `$DetailMsg$[*5]` includes the first through fifth words of the detailed event message ◆ `$ShortMsg$[5*]` includes the fifth through last words of the short event message This script treats the following character values as separators between words: carriage return, line feed, carriage return/line feed combination, form feed, horizontal tab, and space. Everything between those character values in a custom message is considered a word. If you do not enter a keyword, AppManager returns the entire string. The following are examples of the types of messages you can construct using keywords: ◆ Event from `$MachineName$`: The `$ShortMsg$[1-3]` has failed. The last command was `$DetailMsg$[4*]`. ◆ A severity `$Severity$` event has occurred. Call the owner of `$MachineName$` immediately.

Parameter	How to Set It
Attachment Options	
List of attachments	Provide a list of attachments, including the full pathnames. Use semicolons (;) to separate multiple attachments. For example: `c:\temp\SysAdminContacts.log`. **NOTE:** If you enter an invalid path name, the attachment would not be found and an event would be raised if the *Raise event if attachment(s) not found?* parameter is set to Yes.
Raise event if attachment(s) not found?	Select **Yes** to raise an event if the attachment(s) is not found. The default is Yes.
Send message without attachment(s) if not found?	Select **Yes** to send a message if the attachment(s) is not found. The default is unselected.

11.26 SMTPMailRpt

This Knowledge Script supports the report-related (ReportAM) Knowledge Scripts, which can be application-specific reports or generic reports.

Use this Knowledge Script to email the first page of a report to a list of recipients.

The first page of the report contains a list of parameter values and hyperlinks to subsequent pages of the report. The hyperlinks refer to files on the computer where the report is located, so the email is lightweight (approximately 6-10 KB).

For this Action to succeed, generate reports on the same computer where the Report agent is installed, and use the AMAdmin_SetReportPaths Knowledge Script to display the location of each report as a hyperlink. Run AMAdmin_SetReportPaths on the AppManager Repositories object under the Report agent.

If the Admin_SetReportPaths Knowledge Script has not been used to display report locations as hyperlinks, the hyperlinks in the email message will not work.

If there is no data in the report, the email message will not be sent.

The **Location** for this Action must be defined as **MC** (referring to the managed computer where the Report agent is installed), and the Report agent computer must have Internet connectivity to be able to send mail.

NOTE: SMTPMailRpt will not send reports to Lotus Notes client because of a limitation with Lotus Notes in displaying the PNG images. For more information, see http://www-1.ibm.com/support/docview.wss?rs=0&q1=1090737&uid=swg21090737&loc=en_US&cs=utf-8&lang=

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	

Parameter	How to Set It
Raise event if SMTP server is not accessible?	Select **Yes** to raise an event when the SMTP server is not accessible. The default is Yes.
Event severity - SMTP server not accessible	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the SMTP server is not accessible. The default is 35 (magenta event indicator).
Event severity -- Action failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the SMTPMailRpt job fails. The default is 5 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this action. The default is 40.
Action	
List of recipient email addresses	Provide the email address of each individual who should receive the report, using the format `<recipient>@<domain>` (for example, `SLAAdmin@netiq.com`). Use commas (,) to separate multiple recipient addresses. **NOTE:** The following characters are invalid in this parameter: `/ \ [ ] : | = , * ? < >`
Sender's email address	Provide the email address of the person sending the message. **Notes** ◆ In Microsoft Exchange Server 2007 environments, specify the sender name in SMTP format: `<name>@<domain>`. ◆ The following characters are invalid in this parameter: `/ \ [ ] : | = , * ? < >`
SMTP server name	Specify the name of the SMTP mail server used to send mail from the Report agent computer.
SMTP server port	Set the port number for your SMTP server. The default is 25.
Message format	Select the format you want to use for the message sent by this script: ◆ **Standard** format generates a message based upon the selections you make from the *Standard message options* parameters. ◆ **Custom** format generates a message based upon the subject and message body you supply in the *Custom message options* parameters. The default is Standard.
Standard Message Options	
Include date/timestamp?	Select **Yes** to include the date/timestamp in the standard message. The default is Yes.
Include JobID?	Select **Yes** to include the job ID in the standard message. The default is Yes.

Parameter	How to Set It
Include agent computer name?	Select **Yes** to include the name of the agent computer in the standard message (the computer hosting the agent that initiated the action). The default is Yes.
Include event severity?	Select **Yes** to include the severity of the event in the standard message. The default is Yes.
Include Knowledge Script name?	Select **Yes** to include the Knowledge Script name in the standard message (the Knowledge Script that initiated the action). The default is Yes.
Include AppManager object name?	Select **Yes** to include the AppManager resource object name in the standard message (the AppManager resource object where the event was raised). The default is Yes.
Include event detail message?	Select **Yes** to include the event detail message. The default is Yes.
Custom Message Options	
Custom message subject	Provide a subject line for the email message. This parameter is optional. If this parameter is not set, the report title is used for the subject line.

Parameter	How to Set It
Custom message body	Provide the text you want to include in your custom message. You can use the keywords listed below to indicate the information you want to include in the body of your custom message. Add a space before and after keywords to ensure that the keywords display properly. - `$ShortMsg$` (short event message) - `$DetailMsg$` (detailed event message) - `$Time$` (date and time of the event) - `$JobID$` (ID of the job that raised the event) - `$MachineName$` (name of the computer where the event was raised) - `$Severity$` (severity of the event) - `$KSName$` (name of the Knowledge Script that raised the event) - `$ObjectName$` (name of the AppManager resource object where the event was raised) - `$EventID$` (event ID) For `$ShortMsg$` and `$DetailMsg$` you can use number and wildcard options to indicate specific portions of the text string to include. For example: - `$DetailMsg$[5]` includes the fifth word of the detailed event message - `$ShortMsg$[1-5]` includes the first through fifth words of the short message event - `$DetailMsg$[*5]` includes the first through fifth words of the detailed event message - `$ShortMsg$[5*]` includes the fifth through last words of the short event message If you do not enter a word specifier, AppManager returns the entire string. The following are examples of the types of messages you can construct using these keywords: - Event from `$MachineName$`: The `$ShortMsg$[1-3]` has failed. The last command was `$DetailMsg$[4*]`. - A severity `$Severity$` event has occurred! Call the owner of `$MachineName$` immediately!

11.27 SNMPTrap

Use this Knowledge Script to send an SNMP trap message with AppManager event information to a specified list of computers. Each computer you specify must be able to receive SNMP trap messages on UDP port 162.

If you do not specify a value for any of the parameters, this Knowledge Script uses the corresponding value found in the registry under: `HKEY_LOCAL_MACHINE\Software\NetIQ\AppManager\4.0\NetIQmc\SNMPTRAP\Config`.

For example, if you do not specify an object identifier in the OID field, the Knowledge Script checks the registry for the OID key entry: `OID: REG_SZ: 1.3.6.1.4.1.1691.1`.

By default, the event information includes the computer name of the managed client and the event severity. You can select additional information to include by enabling the appropriate parameters.

You can also specify a custom message to forward.

This script raises an event if you select the custom message format but neglect to enter any text for the custom message body. Under these circumstances, the script continues to execute and uses the standard message format.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity -- Action warning	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the SNMPTrap job returns a warning. The default is 35 (magenta event indicator).
Event severity -- Action failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the SNMPTrap job fails. The default is 5 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
List of computers to receive SNMP message	Provide the name of the computer to receive the recipient of the SNMP trap message. The recipient computer must be able to receive SNMP traps on UDP Port 162. To specify multiple recipients, separate computer names with commas. For example, `Nancy01,10.41.40.16,finance03.us.netiq.corp`.
Community string	Provide a valid SNMP community string. Leave this parameter blank to use the SNMP community string entered in AppManager Security Manager. The default is `public`.
Destination port	Provide the number of the port where you want the trap sent. The default is 162.
Object identifier	Provide an object identifier in OID notation (for example, `1.2.3.456.78`). The default is the NetIQ enterprise OID, `1.3.6.1.4.1.1691`.
Specific trap number	Specify a trap number. The trap number can be specific to your application. The default is 1.

Parameter	How to Set It
Message format	Select the format you want to use for the message sent by this script: ◆ **Standard** format generates a message based upon the selections you make from the *Standard message options* parameters. ◆ **Custom** format generates a message based upon the subject and message body you supply in the *Custom message options* parameters. The default is Standard.
Standard Message Options	
Include JobID?	Select **Yes** to include the job ID in the standard message. The default is unselected.
Include agent computer name?	Select **Yes** to include the name of the agent computer in the standard message (the computer hosting the agent that initiated the Action). The default is Yes.
Include event severity?	Select **Yes** to include the severity of the event in the standard message. The default is Yes.
Include Knowledge Script name?	Select **Yes** to include the Knowledge Script name in the standard message (the Knowledge Script that initiated the Action). The default is unselected.
Include AppManager object name?	Select **Yes** to include the AppManager resource object name in the standard message (the AppManager resource object where the event was raised). The default is unselected.
Include AppManager event ID (only on MS Action)?	Select **Yes** to include the AppManager event ID in the standard message (possible only in cases when the Action is carried out by the management server). The default is unselected.
Include event detail message?	Select **Yes** to include the event detail message. The default is unselected.
Custom Message Options	

Parameter	How to Set It
Custom message (can include substitutions)	Provide the text to include in your custom message. Enter the custom message text without quotes. Use the following keywords to indicate the information to include in the message: ◆ `$ShortMsg$` (the short event message) ◆ `$DetailMsg$` (the detailed event message) ◆ `$Time$` (the date and time of the event) ◆ `$JobID$` (the ID of the job that raised the event) ◆ `$MachineName$` (the name of the computer where the event was raised) ◆ `$Severity$` (the severity of the event) ◆ `$KSName$` (the name of the Knowledge Script that raised the event) ◆ `$ObjectName$` (the name of the AppManager resource object where the event was raised) ◆ `$EventID$` (the event ID) For `$ShortMsg$` and `$DetailMsg$` you can use number and wildcard options to indicate specific portions of the text string to include. For example: ◆ `$DetailMsg$[5]` includes the fifth word of the detailed event message ◆ `$ShortMsg$[1-5]` includes the first through fifth words of the short message event ◆ `$DetailMsg$[*5]` includes the first through fifth words of the detailed event message ◆ `$ShortMsg$[5*]` includes the fifth through last words of the short event message If you do not enter a word specifier, AppManager returns the entire string. The following are examples of the types of messages you can construct using these keywords: ◆ Event from `$MachineName$`: The `$ShortMsg$[1-3]` has failed. The last command was `$DetailMsg$[4*]`. ◆ A severity `$Severity$` event has occurred! Call the owner of `$MachineName$` immediately!

11.28 StartServices

Use this Knowledge Script to start Windows services in response to an event.

You can specify a number of seconds to wait after initiating the `service start` command to check the status of the service and make that it was successfully started.

NOTE: If the service you specify has dependent services, enter the names of dependent services after the name of the primary service. For example, SQLServerAgent is a dependent service of MSSQLServer. To start MSSQLServer and its dependent service, specify `MSSQLServer,SQLServerAgent` for the *Service name(s)* parameter. In cases with multiple dependent services, specify the least dependent service first and the most dependent service last.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if service is successfully started?	Select **Yes** to raise an event when the services you specify are successfully started. The default is unselected.
Event severity -- Service started	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a service is successfully started. The default is 25 (blue event indicator).
Event severity -- Action Failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the StartServices job fails. The default is 10 (red event indicator). This Knowledge Script always raises an event when it is unable to start a service.
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
Service name(s)	Provide a comma-separated list of the services you want to start. You can use the *service name* or *display name* listed in the Properties dialog box for the service. **NOTE:** If the service you specify has dependent services, enter the names of dependent services after the primary service. For example, `SQLServerAgent` is a dependent service of `MSSQLServer`. If you want to start `MSSQLServer` and all of its dependent services, specify `MSSQLServer,SQLServerAgent`.
Service start timeout	Set the number of seconds to attempt to start the specified services before timing out. The default is 10.

11.29 StopServices

Use this Knowledge Script to stop Windows services in response to an event.

NOTE: If the service you specify has dependent services, you must list the dependent services before the primary service. For example, SQLServerAgent is a dependent service of MSSQLServer. To stop MSSQLServer and its dependent service, specify `SQLServerAgent,MSSQLServer` for the *Service name(s)* parameter. In cases with multiple dependent services, specify the most dependent service first and the least dependent service last.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if service is successfully stopped?	Select **Yes** to raise an event if the services you specify are successfully stopped. The default is unselected.
Event severity -- Service stopped successfully	Set the event severity level, from 1 to 40, to indicate the importance of an event in which services are successfully stopped. The default is 25 (blue event indicator). This Knowledge Script always raises an event when it is unable to stop a service.
Raise event if service is already stopped?	Select **Yes** to raise an event if the services you specify are already stopped. The default is unselected.
Event severity -- Service already stopped	Set the event severity level, from 1 to 40, to indicate the importance of an event in which specified services are already stopped. The default is 25 (blue event indicator).
Raise event if service is missing?	Select **Yes** to raise an event if the services you specify are missing. The default is unselected.
Event severity -- Service missing	Set the event severity level, from 1 to 40, to indicate the importance of an event in which specified services cannot be found. The default is 25 (blue event indicator).
Event severity -- Action failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the StopServices job fails. The default is 10 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
Service name(s)	Provide a comma-separated list of the services you want to stop. You can use the *service name* or *display name* listed in the Properties dialog box for the service. **NOTE:** If the service you specify has dependent services, enter the dependent services before the primary service. For example, `SQLServerAgent` is a dependent service of `MSSQLServer`. If you want to stop `MSSQLServer` and all of its dependent services, specify `SQLServerAgent,MSSQLServer`.

Parameter	How to Set It
Service stop timeout	Set the number of seconds to attempt to stop the specified services before timing out. The default is 30.

11.30 Traceroute

Use this Knowledge Script to collect exception traceroute data between a specified source and target location in response to an event in another Knowledge Script.

When you enable this script to run automatically in association with another Knowledge Script job, you must specify the source and target locations of the traceroute as parameters. The source location *must* have the ResponseTime for Networks managed object installed and discovered.

When you associate this Action with a monitoring Knowledge Script, you must set the **Location** to **MC** to run the Action on the managed client. Otherwise, this Action creates an error event and will not collect traceroute data when it is invoked.

NOTE: Although any managed client can be selected as the Location, the ResponseTime for Networks managed object must be installed on the managed client.

On the Actions tab, set the Action **Type** value to **Repeat Event - 1** to run a new traceroute each time an event occurs. The **Type** value is dependent on the settings for event collapsing and on the schedule of the associated Knowledge Script. If the Knowledge Script runs and generates events more often than the event collapsing interval (default is 20 minutes), the traceroute Action will not occur at every event. A new child event must be generated for the Action to be executed.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Traceroute source location	The source is where the traceroute is run from. Select a ResponseTime for Networks node. The field may not be left blank. Specify only one source.
Traceroute target location	The target is where the traceroute will be run to. Select a ResponseTime for Networks node, some other AppManager node, an IP address, or a URL. The field may not be left blank. Specify only one target. The script will validate the source and target locations are not the same, and will generate an error if they are identical.
Maximum number of hops	Set the maximum number of hops, from 1 to 30, allowed in the traceroute. The default is 30.
Event when traceroute fails?	Select **Yes** to raise events if the Traceroute job fails. The default is Yes.
Event severity -- Traceroute failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the Traceroute job fails. The default is 20.

Example of How this Script Is Used

Before you launch a Knowledge Script (other than one of the Networks-RT scripts), double-click it to see its Properties dialog box. Click the **Actions** tab. Click **New** and select **Action_Traceroute** from the list. Then click **Properties** to specify the source location and target location for the traceroute. If an event is generated by the Knowledge Script, the Action_Traceroute Knowledge Script is launched automatically. It collects traceroute data between the source and target you selected and stores the traceroute data in the AppManager repository.

The traceroute data is associated with the event that triggered the traceroute. Run the Report_TracerouteException Knowledge Script to generate a report that compares the traceroute data collected for this event with the historical traceroute data for the associated source and target locations.

11.31 TracerouteNetworks-RT

Use this Knowledge Script to collect exception traceroute data between a specified source and target location in response to an event in a separate Networks-RT Knowledge Script.

You do not have to specify source or target information when associating the Action script with the Knowledge Script. This script automatically determines the source and target locations for the traceroute, based on the event details from the Knowledge Script.

When you associate this Action with a monitoring Knowledge Script, you must set the **Location** to **MC** to run the Action on the managed client. Otherwise, this Action creates an error event and will not collect traceroute data when invoked.

NOTE: Although any managed client can be selected as the Location, the ResponseTime for Networks managed object must be installed on the managed client.

On the Actions tab, set the Action **Type** value to **Repeat Event - 1** to run a new traceroute each time an event occurs. The **Type** value is dependent on the settings for event collapsing and on the schedule of the associated Knowledge Script. If the Knowledge Script runs and generates events more often than the event collapsing interval (default is 20 minutes), the traceroute Action will not occur at every event. A new child event must be generated for the Action to be executed.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Maximum number of hops	Set the maximum number of hops, from 1 to 30, allowed in the traceroute. The default is 30.
Event when traceroute fails?	Select **Yes** to raise an event if the TracerouteNetworks-RT job fails. The default is Yes.
Event severity for traceroute failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the TracerouteNetworks-RT job fails. The default is 20.

Example of How this Script Is Used

Before you launch a Networks-RT Knowledge Script, double-click it and click the **Actions** tab on the Properties dialog box. Click **New**, and select **Action_TracerouteNetworks-RT** from the list. If an event is generated by the Knowledge Script, the Action_TracerouteNetworks-RT Knowledge Script is launched automatically. It collects traceroute data between the source and target locations associated with the event, and stores the traceroute data in the AppManager database.

The traceroute data is associated with the event that triggered the traceroute. Run the Report_TracerouteException Knowledge Script to generate a report that compares the traceroute data collected for this event with the historical traceroute data for the given pair of endpoints.

11.32 UpdateEventStatus

Use this Action Knowledge Script to acknowledge or close AppManager events from one or more specified computers. If you do not specify any computer, the computer that submitted the event is used. This script raises an event if an action fails.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Severity for Action failure	Set the severity level, from 1 to 40, to indicate the importance of the event when the UpdateEventStatus job fails. The default is 5 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
SQL Server login	Specify the SQL Server user account used to run this Knowledge Script, for example, sa. This account must exist on the SQL Server instance where the AppManager repository is installed whose event status will be updated. The password for this specified account must be configured in AppManager's Security Manager. For more information about configuring, see Configuring Security Manager for Action Knowledge Scripts. If this parameter is empty, Windows authentication will be used, and the account running the NetIQ AppManager Client Resource Monitor service should have authority to connect to, query, and update data on the SQL Server instance where the AppManager repository is installed.
Repository Server	Provide the name of the server that hosts the AppManager repository.
Repository Database Name	Provide the name of the AppManager repository. The default is QDB.

Parameter	How to Set It
Update event status to...	Select the update event status as closed or acknowledged. By default, the update event status is Acknowledged.
Filter Options	
Filter by computers?	Select **Yes** to filter events by computers. The default is Yes.
Filter by the computer that raised the event?	Select **Yes** to include the computer that raised the event. The default is Yes.
Filter by these computers	Select the computers or provide a comma-separated list of computer names by which you want to filter events. For example: `QELAB,PORT1,Chris`
Filter by these Knowledge Scripts	Provide a comma-separated list of Knowledge Scripts by which you want to filter events. For example, `Action_IISContinueSite,Action_Messenger,Action_Diagnose`
Filter by event severity	Specify the severity level by which you want to filter events. By default, the event severity is 1 to 40.
Filter by event message	Specify the message by which you want to filter events. The event message indicates whether or not the action was successful, and provides an explanation if the action fails.
Filter by event age	Specify the age of the event. The default is 0 minutes.
Include only events that are...	Indicate whether to include newer or older events based on the age of the event. The default is Older.

11.33 UXCommand

Use this Knowledge Script to run a non-interactive UNIX command in response to an event. For example, you can use this Knowledge Script to run a batch command for appending a log file or stopping a process.

You can include arguments in the command-line string, but you need to escape any double quote (") or special characters by typing a preceding backslash (\). Special characters for Perl-based scripts include the dollar sign ($), percentage (%), and at symbol (@). In addition, avoid using the ampersand (&), the dollar sign ($), or backquotes (`) in the command string.

Setting Parameter Values

Set the following parameter as needed:

Parameter	How to Set It
Non-interactive UNIX command	Specify the command to run. Do not enter a command that requires user input. If your command line includes double quotes or any other special characters, use a backslash to escape the characters. For example: `grep -l \"Abnormal shutdown\" applog* > /tmp/fail`. The command you enter should include all necessary arguments and handle any input and output redirection or file management required.

Example of How this Script Is Used

Use this Action is to create a script file that contains a series of commands to diagnose or correct problems on a server you are monitoring and have this Action launch your script file when an event is detected.

To run this Action on the managed UNIX computer, select **MC** (Managed Client) as the Location on the **Action** tab of the Properties dialog box. Also verify that the NetIQ UNIX agent account has permission to execute the command you want to run on the computer where you want the Action executed.

11.34 WriteMsgToFile

Use this Knowledge Script to write AppManager event information to a file. This file gets written to the computer that is running the action. The designation of that computer is controlled by the Action dialog when the monitoring job is created, and the options are `MC` (agent), `MS` (management server), or `proxy`, where proxy in turn prompts the user to designate any agent computer that is known to AppManager.

By default, the event information includes the agent computer name and the event severity level. You can select additional information to include by enabling the appropriate parameters.

You can also construct a custom message.

An event is raised when you select the custom message format but neglect to enter any text for the custom message body. Under these circumstances, the script continues to execute and uses the standard message format.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Event severity -- Action warning	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the WriteMsgToFile job returns a warning. The default is 35 (magenta event indicator).
Event severity -- Action failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the WriteMsgToFile job fails. The default is 5 (red event indicator).
Severity Configuration	
Minimum event severity for Action	Set the minimum severity level, from 1 to 40, for an event that triggers this Action. The default is 1.
Maximum event severity for Action	Set the maximum severity level, from 1 to 40, for an event that triggers this Action. The default is 40.
Action	
Full path to file	Provide the complete path to the file where you want to store event information or click Browse [...] to find the file. The default is `c:\temp\NetiQACT_Dump.txt`.

Parameter	How to Set It
Append event information?	Select **Yes** to append event information to the specified file. If you do not select Yes, the log file is overwritten each time the Action runs. The default is Yes.
Number of carriage returns between messages.	Set the number of carriage returns from 0 to 20. The default value is 2.
Create folder if it does not exist?	Select **Yes** to create folders specified in the file path if they do not exist. The default is Yes.
Message format	Select the format you want to use for the message sent by this script: ◆ **Standard** format generates a message based upon the selections you make from the *Standard Message Options* parameters. ◆ **Custom** format generates a message based upon the subject and message body you supply in the *Custom Message Options* parameters. The default is Standard.
Standard Message Options	
Include date/timestamp?	Select **Yes** to include the date/timestamp in the standard message. The default is unselected.
Include JobID?	Select **Yes** to include the job ID in the standard message. The default is unselected.
Include agent computer name?	Select **Yes** to include the name of the agent computer in the standard message (the computer hosting the agent that initiated the Action). The default is Yes.
Include event severity?	Select **Yes** to include the severity of the event in the standard message. The default is Yes.
Include Knowledge Script name?	Select **Yes** to include the Knowledge Script name in the standard message (the Knowledge Script that initiated the Action). The default is unselected.
Include AppManager object name?	Select **Yes** to include the AppManager resource object name in the standard message (the AppManager resource object where the event was raised). The default is unselected.
Include AppManager event ID?	Select **Yes** to include the AppManager event ID in the standard message (possible only in cases when the Action is carried out by the management server). The default is unselected. **NOTE:** This Knowledge Script also displays the event ID on the proxy computer.
Include event detail message?	Select **Yes** to include the event detail message. The default is unselected.
Custom Message Options	

Parameter	How to Set It
Custom text (can include substitutions)	Provide the text you want to include in your custom message. You can use the keywords listed below to indicate the information you want to include in the body of your custom message. Add a space before and after keywords to ensure that the keywords display properly. ◆ `$ShortMsg$` (short event message) ◆ `$DetailMsg$` (detailed event message) ◆ `$Time$` (date and time of the event) ◆ `$JobID$` (ID of the job that raised the event) ◆ `$MachineName$` (name of the computer where the event was raised) ◆ `$Severity$` (severity of the event) ◆ `$KSName$` (name of the Knowledge Script that raised the event) ◆ `$ObjectName$` (name of the AppManager resource object where the event was raised) ◆ `$EventID$` (event ID) ◆ `$tab$` inserts four whitespace characters in the message body ◆ `$lf$` inserts a line feed in the message body ◆ `$crlf$` inserts a carriage-return line feed in the message body ◆ `$cr$` inserts a carriage-return in the message body For `$ShortMsg$` and `$DetailMsg$` you can use number and wildcard options to indicate specific portions of the text string to include. For example: ◆ `$DetailMsg$[5]` includes the fifth word of the detailed event message ◆ `$ShortMsg$[1-5]` includes the first through fifth words of the short message event ◆ `$DetailMsg$[*5]` includes the first through fifth words of the detailed event message ◆ `$ShortMsg$[5*]` includes the fifth through last words of the short event message If you do not enter a word specifier, AppManager returns the entire string. The following are examples of the types of messages you can construct using these keywords: ◆ Event from `$MachineName$`: The `$ShortMsg$[1-3]` has failed. The last command was `$DetailMsg$[4*]`. ◆ A severity `$Severity$` event has occurred! Call the owner of `$MachineName$` immediately!

12 NTAdmin Knowledge Scripts

The NTAdmin category provides Knowledge Scripts for performing Windows administrative tasks or special one-time activities. To run Knowledge Scripts in this category, you need to be defined as a user with administrator privileges on the computer on which you run the scripts.

Some Knowledge Scripts perform AppManager administrative tasks or operations that should be restricted to a limited number of users. To control access to these Knowledge Scripts, the scripts are only made available to users who are assigned to the Administrator role through AppManager Security Manager.

Most of the administrative Knowledge Scripts are in the Action, AMAdmin, and NTAdmin categories, and by default these Knowledge Scripts are designated as being for administrators only. For more information, see the *Administrator Guide for AppManager*.

From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. In the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
AddGroup	Adds a local or domain Windows group account.
AddGroupViaAD	Adds a domain group account to Active Directory.
AddUser	Adds a local or domain Windows user account.
AddUserViaAD	Adds a domain user account to Active Directory.
ChangePassword	Changes the password for a specified local or domain user account.
CheckServicePack	Compares the Windows service pack level on a managed computer to a specified level.
CloseSharedFiles	Closes a shared file and removes the file lock.
DeleteGroup	Deletes a local or domain Windows group account.
DeleteUser	Deletes a local or domain Windows user account.
FileCheck	Checks whether a particular file exists.
ModifyServiceConfig	Changes the configuration for specified services.
RegistrySet	Sets or creates a Windows registry key.
RestartService	Schedules a service to automatically stop and restart a service after a specified time interval.
RunDOS	Runs a non-interactive DOS command.
SNMPSet	Sets a value for a selected SNMP MIB variable.
SyncTime	Synchronizes the system time among computers.
UnixAgentHealthProxy	Checks the availability of a remote managed UNIX computer and monitors the health of the remote UNIX agent. This Knowledge Script uses a proxy Windows agent to monitor remote UNIX agents.

12.1 AddGroup

Use this Knowledge Script to add a Windows domain group account or local group account. Domain groups are added to the domain associated with the managed computer where the script runs. This script raises an event when the operation is successful or when it fails.

NOTE: You cannot create a local group on a Windows Domain Controller.

Prerequisite

Before running this script, ensure that the AppManager Client Resource Monitor service (`netiqmc.exe`) is set to log on as a domain user account in the same domain as, or a domain trusted by, the target computer.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Name of group to be added	Provide the name of the domain group account to add. Active Directory Users and Computers displays the group name in **Group name**.
Event if group was added?	Set to **y** to raise an event if the domain group was added. The default is n.
Event if group was not added?	Set to **y** to raise an event if the domain group was not added. The default is y.
Add local group?	Set to **y** to create a local group account on the computer where the script job is running. If set to n, you must enable *Add domain group?* to add a domain group. The default is n.
Add domain group?	Set to **y** to associate the domain group account with the domain of the computer where the script job is running. If set to n, you must enable *Add local group?* to add a local group. The default is y.
Event severity: Group was not added	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a domain group was not added. The default is 12 (yellow event indicator).
Event severity: Group was added	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a domain group was added. The default is 25 (blue event indicator).

12.2 AddGroupViaAD

Use this Knowledge Script to add a domain group account to Active Directory. You must run this script on a computer that is a Domain Controller. The group is added to the domain associated with the managed computer where the script runs. This script raises an event when the operation is successful or when it fails.

Prerequisite

Before running this script, ensure that the AppManager Client Resource Monitor service (`netiqmc.exe`) is set to log on as a domain user account in the same domain as, or a domain trusted by, the target computer.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Name of group to be added	Provide the name of the domain group account to add. Active Directory Users and Computers displays the group name in **Group name**.
Event if group was added?	Set to **y** to raise an event if the domain group was added. The default is n.
Event if group was not added?	Set to **y** to raise an event if the domain group was not added. The default is y.
Event severity: Group was not added	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a domain group was not added. The default is 12 (yellow event indicator).
Event severity: Group was added	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a domain group was added. The default is 25 (blue event indicator).

12.3 AddUser

Use this Knowledge Script to add a domain user account or local user account. A domain user account is added to the domain associated with the managed computer where the script runs. This script raises an event when the operation succeeds or fails.

NOTE: You cannot create a local user on a Windows Domain Controller.

Prerequisite

Before running this script, ensure that the AppManager Client Resource Monitor service (`netiqmc.exe`) is set to log on as a domain user account in the same domain as, or a domain trusted by, the target computer.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Name of user to be added	Provide the name of the user account to add.
Password of user to be added	Provide the password of the user account to add.
Event if user was added?	Set to **y** to raise an event if the user was added. The default is n.
Event if user was not added?	Set to **y** to raise an event if the user was not added. The default is y.
Add domain user?	Set to **y** to add the user as a domain user in the current client's domain. If set to y, the user account is associated with the domain of the computer where the script is running. If set to n, a local user account is created on the computer where the script is running. The default is y.
Event severity: User was not added	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the user was not added. The default is 12 (yellow event indicator).
Event severity: User was added	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the user was added. The default is 25 (blue event indicator).

12.4 AddUserViaAD

Use this Knowledge Script to add a user account to Active Directory. You must run this script on a computer that is a Domain Controller. The user account is added to the domain associated with the computer where the script runs. This script raises an event when the operation is successful or when it fails.

Prerequisite

Before running this script, ensure that the AppManager Client Resource Monitor service (`netiqmc.exe`) is set to log on as a domain user account in the same domain as, or a domain trusted by, the target computer.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Name of user to be added	Provide the name of the user account to add.
Password of user to be added	Provide the password of the user account to add.
Event if user was added?	Set to **y** to raise an event if the user account was added. The default is y.
Event if user was not added?	Set to **y** to raise an event if the user account was not added. The default is y.
Event severity: User was not added	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the user was not added. The default is 12 (yellow event indicator).
Event severity: User was added	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the user was added. The default is 25 (blue event indicator).

12.5 ChangePassword

Use this Knowledge Script to change the password for a specified domain user account or local user account. This script raises an event when the change password operation is successful or when it fails.

Prerequisite

Before running this script, ensure that the AppManager Client Resource Monitor service (`netiqmc.exe`) is set to log on as a domain user account in the same domain as, or a domain trusted by, the target computer.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Name of user account	Provide the name of the user account whose password should be changed. The default is guest.
New password for user account	Provide the new password for the user account.
Is the account a domain user?	Set to **y** to change the password for a domain user account. Set to **n** to change the password for a local user account. If set to y, the affected domain user account is one associated with the domain of the managed client where the Knowledge Script job is running. The default is y.
Event if password was changed	Set to **y** to raise an event if the password is changed successfully. The default is n.
Event if password was not changed	Set to **y** to raise an event if the password is not changed. The default is y.
Event severity: Password was not changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the password was not changed. The default is 12 (yellow event indicator).
Event severity: Password was changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the password was changed. The default is 25 (blue event indicator).

12.6 CheckServicePack

Use this Knowledge Script to compare the installed version of a Microsoft Windows service pack to a specified minimum threshold. This script raises an event if the version of the service pack installed on the target computer falls below the minimum threshold.

This script queries the computer’s registry to determine the current service pack level.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Weekly**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Create event if service pack does not meet minimum?	Select **Yes** to raise an event if the version of the service pack installed on the target computer falls below the threshold. The default is Yes.
Severity - Service pack does not meet minimum	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service pack version falls below the threshold. The default is 8 (red event indicator).
Severity - Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the CheckServicePack job fails. The default is 8 (red event indicator).
Data Collection	
Collect service pack data?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the target computer's configuration, including the Windows version and service pack number. The default is unselected.
Monitoring	
Check Windows XP Service Pack?	Select **Yes** to check the Windows XP service pack level on the target computer. The default is Yes.
Threshold - Service pack minimum	Specify the minimum version of the Windows XP service pack that must be installed to prevent an event from being raised. For example, to check whether Service Pack 1 (or later) is installed, enter 1. The default is 3.
Check Windows Server 2003 Service Pack?	Select **Yes** to check the Windows Server 2003 service pack level on the target computer. The default is Yes.
Threshold - Service pack minimum	Specify the minimum version of the Windows Server 2003 service pack that must be installed to prevent an event from being raised. For example, to check whether Service Pack 1 (or later) is installed, enter 1. The default is 2.
Check Windows Vista Service Pack?	Select **Yes** to check the Windows Vista service pack level on the target computer. The default is Yes.
Threshold - Service pack minimum	Specify the minimum version of the Windows Vista service pack that must be installed to prevent an event from being raised. For example, to check whether Service Pack 1 (or later) is installed, enter 1. The default is 2.
Check Windows Server 2008 Service Pack?	Select **Yes** to check the Windows Server 2008 service pack level on the target computer. The default is Yes.
Threshold - Service pack minimum	Specify the minimum version of the Windows Server 2008 service pack that must be installed to prevent an event from being raised. For example, to check whether Service Pack 1 (or later) is installed, enter 1. The default is 2.
Check Windows 7 Service Pack?	Select **Yes** to check the Windows 7 service pack level on the target computer. The default is Yes.
Threshold - Service pack minimum	Specify the minimum version of the Windows 7 service pack that must be installed to prevent an event from being raised. For example, to check whether Service Pack 1 (or later) is installed, enter 1. The default is 0.
Check Windows Server 2008 R2 Service Pack?	Select **Yes** to check the Windows Server 2008 R2 service pack level on the target computer. The default is Yes.

Parameter	How to Set It
Threshold - Service pack minimum	Specify the minimum version of the Windows Server 2008 R2 service pack that must be installed to prevent an event from being raised. For example, to check whether Service Pack 1 (or later) is installed, enter 1. The default is 0.
Check Windows 8 Service Pack?	Select **Yes** to check the Windows 8 service pack level on the target computer. The default is Yes.
Threshold - Service pack minimum	Specify the minimum version of the Windows 8 service pack that must be installed to prevent an event from being raised. For example, to check whether Service Pack 1 (or later) is installed, enter 1. The default is 0.
Check Windows Server 2012 Service Pack?	Select **Yes** to check the Windows Server 2012 service pack level on the target computer. The default is Yes.
Threshold - Service pack minimum	Specify the minimum version of the Windows Server 2012 service pack that must be installed to prevent an event from being raised. For example, to check whether Service Pack 1 (or later) is installed, enter 1. The default is 0.

12.7 CloseSharedFiles

Use this Knowledge Script to close one or more shared files and remove corresponding file locks. You must run this script on the server where the file is shared.

You must specify a shared file by its file identifier. You can view the file identifier in one of the following ways:

- At a Command Prompt, run the `net file` command to see a list of files, corresponding identifiers, and lock information.
- See the NT SharedFiles Knowledge Script.

Prerequisite

Before running this script, ensure that the AppManager Client Resource Monitor service (`netiqmc.exe`) is set to log on as a domain user account in the same domain as, or a domain trusted by, the target computer.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
List of file identifiers	Specify the file identifier for each file you want to close. Use a comma to separate multiple identifiers. The default is 1,2,3.
Event if file was closed?	Set to **y** to raise an event if the file is closed successfully. The default is n.
Event if file was not closed?	Set to **y** to raise an event if the file is not closed. The default is y.
Event severity: File was not closed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the file was not closed. The default is 12 (yellow event indicator).
Event severity: File was closed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the file was closed. The default is 25 (blue event indicator).

12.8 DeleteGroup

Use this Knowledge Script to delete a specified domain group account or local group account. This script raises an event when the group is deleted successfully or when the deletion fails.

Prerequisite

Before running this script, ensure that the AppManager Client Resource Monitor service (`netiqmc.exe`) is set to log on as a domain user account in the same domain as, or a domain trusted by, the target computer.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Name of the group to be deleted	Provide the name of the group account you want to delete.
Event if group is deleted?	Set to **y** to raise an event if the group account is deleted successfully. The default is n.

Parameter	How to Set It
Event if group is not deleted?	Set to **y** to raise an event if the group account is not deleted. The default is y.
Delete local group?	Set to **y** to delete a local group. If set to n, you must enable *Delete domain group?* to delete a domain group. The default is n.
Delete domain group?	Set to **y** to delete a domain group account. If set to n, enable *Delete local group?* to delete a local group. If set to y, the domain account affected is one associated with the domain of the computer where the job is running. The default is y.
Event severity: Group was not deleted	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the group was not deleted. The default is 12 (yellow event indicator).
Event severity: Group was deleted	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the group was deleted. The default is 25 (blue event indicator).

12.9 DeleteUser

Use this Knowledge Script to delete a specified domain user account or local user account. This script raises an event when the user account is deleted successfully or when the deletion fails.

Prerequisite

Before running this script, ensure that the AppManager Client Resource Monitor service (`netiqmc.exe`) is set to log on as a domain user account in the same domain as, or a domain trusted by, the target computer.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Name of the user to be deleted	Provide the name of the user account you want to delete.
Event if user is deleted?	Set to **y** to raise an event if the user account is deleted successfully. The default is n.
Event if user is not deleted?	Set to **y** to raise an event if the user account is not deleted. The default is y.

Parameter	How to Set It
Delete domain user?	Set to **y** to delete a domain user account. Set to **n** to delete a local user account. If set to n, the local user account affected is one associated with the domain of the computer where the script is running. The default is y.
Event severity: User was not deleted	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the user was not deleted. The default is 12 (yellow event indicator).
Event severity: User was deleted	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the user was deleted. The default is 25 (blue event indicator).

12.10 FileCheck

Use this Knowledge Script to check whether a particular file exists on one or more computers. This script raises an event if it finds the file you specified.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Full path to file	Provide the full path to the file you want to check, for example: `%systemroot%\system32\clock.exe` **Notes** ◆ If you include spaces in the filepath, the event details include a broken link to the file. ◆ If the file is on a network-mounted drive, the computer where you run this script must be running with a Windows domain account that can access the remote drive. Otherwise, the job may report an error because it cannot access files on the remote drive.
Event when found?	Set to **y** to raise an event if the specified file is found. If set to n, an event is raised when the file cannot be found. The default is y.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the specified file is found or not found. The default is 5 (red event indicator).

12.11 ModifyServiceConfig

Use this Knowledge Script to change the configuration for specified services. To apply your changes, you must manually restart the services. This script raises an event indicating whether the service configuration change you specified was successful.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
List of services	Provide the names of the services whose configurations you want to modify, separating the names with commas (,). Enter the internal service names or the service names displayed in the Services Control Panel.
Collect data?	Set to **y** to collect data. If enabled, data collection returns: ◆ **100** -- service successfully modified, or ◆ **0** -- service modification failed. The default is n.
Service type	Select a new service type. Valid types are: ◆ own ◆ share ◆ interact ◆ kernel ◆ filesys Select `NO_CHANGE` to keep the current configuration. Default is `NO_CHANGE`.
Start type	Select a new start type. Valid types are: ◆ boot ◆ system ◆ auto ◆ demand ◆ disabled Select `NO_CHANGE` to keep the current configuration. Default is `NO_CHANGE`.

Parameter	How to Set It
Error control	Select the level of error control. Valid types are: ◆ normal ◆ severe ◆ critical ◆ ignore Select NO_CHANGE to keep the current configuration. Default is NO_CHANGE.
Log On As account	Specify the system or user account name for the service to log on as. Although most services log on as a system account, you can configure some services to log on using special user accounts. Default is NO_CHANGE. **NOTE:** The account you specify here must already have the right to log on as a service.
Password for Log On As account	Specify the password for the logon user account.
Display name of the service	Provide a new display name for the service. The default is NO_CHANGE.
Event severity: Service was changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service was changed. The default is 20 (yellow event indicator).
Event severity: Service was not changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the service was not changed. The default is 8 (red event indicator).

12.12 RegistrySet

Use this Knowledge Script to set or create a Windows registry key. You can specify a list of computers where you want to set the key value, either directly using the *List of computers* parameter or in a file containing a list of computer names or addresses. This script creates a new key if the specified key does not exist. In addition, this script raises an event if the set operation fails. This script can also raise an event if the set operation completes successfully.

On 64-bit Windows systems, you can configure this script to set registry information for 32-bit or 64-bit programs.

- To set registry information for a 64-bit application, disable the *Set 32-bit program registry keys on a 64-bit system?* parameter and specify the registry path and key under `HKEY_LOCAL_MACHINE\Software`.
- To set registry information for a 32-bit application, enable the *Set 32-bit program registry keys on a 64-bit system?* parameter and specify the registry path and key exactly as it would be specified on a 32-bit system, under `HKEY_LOCAL_MACHINE\Software`.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event when set?	Set to **y** to raise an event when the registry key is set successfully. The default is n.
Set 32-bit program registry keys on a 64-bit system?	On a 64-bit Windows system, set this parameter to **y** to set registry information for 32-bit programs. The default value, n, enables you to set registry information for 64-bit programs. On a 32-bit Windows system, this parameter is not applicable and will be ignored. **Tip** To set registry information for 32-bit programs and 64-bit programs, configure separate jobs.
Root key	Specify the registry root. Valid root options are: ◆ `HKEY_LOCAL_MACHINE` ◆ `HKEY_CLASSES_ROOT` ◆ `HKEY_CURRENT_USER` ◆ `HKEY_USERS` The default is `HKEY_LOCAL_MACHINE`.
Full path to registry key	Provide the full path to the registry key. You can enter any path under the root key as long as you have write permission. The default path is: `Software\NetIQ\AppManager\4.0\NetIQmc\Tracing` To specify the path to the registry key for a 32-bit or 64-bit program, specify a path under `HKEY_LOCAL_MACHINE\Software`. **NOTE:** Although the registry keys for 32-bit programs on a 64-bit system are stored under the key `HKEY_LOCAL_MACHINE\Software\Wow6432Node`, **do not** specify the `Wow6432Node` component of the path. Instead, specify the path without the `Wow6432Node` component, and enable the *Set 32-bit program registry keys on a 64-bit system?* parameter.
Key name	Specify the name of the key to set or create. The default is `TraceMC`.
Key type	Specify the type for the key. Valid types are: ◆ `REG_SZ` ◆ `REG_MULTI_SZ` ◆ `REG_DWORD` ◆ `REG_EXPAND_SZ` The default is `REG_DWORD`.
Value of the key	Specify the value to which the registry key should be set. Hex values can be entered in the form `0x`*nnnn*. Default is 1.
NULL Character (Only for REG_MULTI_SZ)	Specify the character separator for `REG_MULTI_SZ` type. For example to set *foo bar*, you need to be able to indicate the blank space between *foo* and *bar* (foo+bar). The default is +. **NOTE:** Choose a character that is not part of the name.

Parameter	How to Set It
List of computers	Specify the computers for which you want to set this value. If you leave this parameter blank, the registry key located on the local computer is set. To set the value for multiple computers, enter the hostname or IP address for each computer in a comma-separated list. For example: `CORP01,ENGR02,10.15.221.5.`

Full path to file with list of computers	To set values for more computers than is convenient to enter one at a time in the *List of computers* parameter, create a file that contains a list of the computers you want to monitor.
	Provide the full path to the file containing a list of the computers whose registry key values you want to set. The file should contain the hostname or IP address for each computer in one or more lines. Each line can have multiple computer names, separated by commas. For example:
	`NYC01,NYC02`
	`SALES01,10.15.221.5,SFO01`
	`LABMACH, QATEST`

Severity: Key was set	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the key is set. The default is 16 (yellow event indicator).
Severity: Key was not set	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the key is not set. The default is 5 (red event indicator).

12.13 RestartService

Use this Knowledge Script to schedule a service to automatically stop and restart after a specified interval. You can also have the script restart any services that depend on the ones you have stopped. This script raises an event if it fails to restart a service. Note that you cannot use this script to stop the NetIQ Client Resource Monitor (`netiqmc`) agent service.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Daily**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Create event if service restarts successfully?	Select **Yes** to raise an event if the script successfully restarts a service and, optionally, its dependent services. The default is Yes.

Parameter	How to Set It
Severity - Service restarted	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the script successfully restarts a service. The default is 25 (blue event indicator).
Create event if service restart fails?	Select **Yes** to raise an event if the script fails to restart a service. The default is Yes.
Severity - Service start failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the script fails to restart a service. The default is 5 (red event indicator).
Severity - Service stop failed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the script fails to stop a service. The default is 5 (red event indicator).
Create event if service is missing?	Select **Yes** to raise an event if a specified service is missing. The default is unselected.
Severity - Service missing	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a specified service is missing. The default is 8.
Create event if service is disabled?	Select **Yes** to raise an event if the script disables a service. The default is Yes.
Severity - Service disabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the script disables a service. The default is 12.
Create event if service is shut down normally?	Select **Yes** to raise an event if the script shuts down a service normally. The default is Yes.
Severity - Service shut down normally	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the script shuts down a service normally. The default is 30.
Severity - Job Failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the RestartService job fails unexpectedly. The default is 10 (red event indicator).
Data Collection	
Collect data?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns: ◆ **100** -- service was restarted successfully, or ◆ **0** -- service did not restart. The default is unselected.
Administration	
Service list	Provide the names of the services to you want to stop. Separate each name with a comma (,) and no spaces. The default is `Alerter,Messenger`.
Service start/stop delay	Specify the number of seconds to wait after a service is stopped before attempting to automatically restart it. The default is 30.
Restart dependent services?	Select **Yes** to also restart any services that depend on the services you stopped. The default is Yes. For example, if you stop the `MSSQLSERVER` service, the `SQLSERVERAGENT` service is also stopped. You can use this parameter to restart the agent service along with the server.

Parameter	How to Set It
Service start/stop retry count	Specify the number of times to attempt to restart a service after it has stopped. The default is 3 times.
Restart service if shut down normally?	Select **Yes** to restart a service that is shut down normally. By default, this script restarts services that are shut down normally.

12.14 RunDOS

Use this Knowledge Script to run a non-interactive DOS command. This script raises an event if an unexpected exit code is raised. For example, use this script to run a batch command for virus scanning, disk backup, or logging an entry in a trouble ticket system.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Every 30 minutes**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Create event if command executes successfully?	Select **Yes** to raise an event if the DOS command you specified executes successfully. The default is unselected.
Severity - Command executed successfully	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the specified DOS command executes successfully. The default is 25 (blue event indicator).
Create event if process failed to execute?	Select **Yes** to raise an event if the DOS command you specified did not execute successfully. The default is Yes.
Severity - Command failed to execute	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the specified DOS command did not execute successfully. The default is 5 (red event indicator).
Severity - Job failure	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the RunDOS job fails unexpectedly. The default is 5 (red event indicator).
Data Collection	
Collect exit code data?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns exit code data; including the specified command string, current exit code, normal exit code, and explanation. The default is unselected.
Administration	

Parameter	How to Set It
Normal/Expected exit code	Specify the exit (return) code expected when the command runs and exits normally (successfully). The default is 0.
Command or full path to script file	Specify the DOS command or script filename to run. For example, `C:\temp\myscript.bat`. Do not use a command that requires input. **NOTE:** If the command you are entering includes quotation marks ("), enclose the quoted string in a second set of quotation marks. For example, if the DOS command is `net send "message"` you would enter: `net send ""message""`.
Full path to command output file	Provide the full path and filename to specify a file to write command output. If you do not specify the full path, AppManager creates the file in `%systemroot%\System32\` or `%systemroot%\SysWOW64`.
Full path to error output file	Provide the full path and filename to specify a file to write error output. If you do not specify the full path, AppManager creates the file in `%systemroot%\System32\` or `%systemroot%\SysWOW64`.
Append to output files?	Select **Yes** to add information to the command and error output files. The default is Yes.
Time to wait for process to complete	Specify the maximum number of seconds the specified DOS command or script file should take to execute. The default is 10. Set to 0 to run the DOS command without checking the exit code.

12.15 SNMPSet

Use this Knowledge Script to perform an SNMP v1 `Set` operation for the selected SNMP MIB variable. You can specify a list of computers where you want to set the variable value directly using the *List of computers* parameter or in a file containing a list of computer names or addresses. This script raises an event if the `Set` operation encounters an error and, optionally, if the `Set` operation completes successfully. This script requires the Microsoft SNMP Service to be running and security for the service set to allow Read and Write privileges.

Prerequisite

By default, the security for the Microsoft SNMP Service is typically set to the read-only permission level. To use this script, the Microsoft SNMP Service must be set to the read/write permission level. For more information, see the following topics:

- “Checking SNMP Service Rights” on page 446
- “Changing the Permission Level for the SNMP Service” on page 446

Resource Objects

Windows server or HP SIM server

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the `Set` operation completes successfully. The default is n. This script always raises an event if the `Set` operation encounters an error.
Object identifier	For the MIB variable, specify the MIB object identifier in the OID notation (for example `.1.2.3.456.78`) or ODE notation (for example, `system.sysUptime.0`). The default is `system.sysContact.0`. If you are using the object identifier (OID), you must include the dot (.) at the beginning of the identifier. If you are using the object descriptor (ODE), use a case-sensitive descriptor. You can use the object descriptor if the `mib.bin` file has been compiled on the agent machine (in the `%windir%/system32` directory). For information about compiling the `mib.bin`, see the Windows Resource Kit.
Community string	Provide a valid SNMP community string name. Leave this parameter blank to use the SNMP community name entered in AppManager Security Manager. The default is public.
Set type	Specify the data type for the MIB variable to set. Valid values are: ◆ `OCTETSTRING` ◆ `INTEGER` ◆ `COUNTER` ◆ `GAUGE` ◆ `TIMETICKS` ◆ `IPADDRESS` The default is `OCTETSTRING`.
Set value	Specify the value you want to set.
List of computers	Specify the computers whose values you want to set. If you leave this parameter blank, this script sets the SNMP MIB variable located on the managed computer where the Knowledge Script job is running. To set the value for multiple computers, enter the hostname or IP address for each computer in a comma-separated list. For example: `CORP01,ENGR02,10.15.221.5`

Full path to computer list file	To set values on more computers than is convenient to list one at a time in the *List of computers* parameter, create a file containing a list of the computers whose values you want to set.
	Enter the full path to the file. The file should contain the hostname or IP address for each computer in one or more lines. Each line can have multiple computer names, separated by commas. For example:
	`NYC01,NYC02`
	`SALES01,10.15.221.5,SFO01`
	`LABMACH, QATEST`

Parameter	How to Set It
Event severity: Variable was not set	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the specified value was not set. The default is 5 (red event indicator). This script always raises an event when a `Set` operation fails.
Event severity: Variable was set	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the variable was set. Default is 16 (yellow event indicator). The *Event?* parameter must be enabled to raise an event for a successful `Set` operation.
Threshold for attempts to contact SNMP agent	Specify the maximum number of times the script should try to contact the SNMP agent before raising an event. The default is 3 times.
Threshold (in seconds) for a response	Specify the maximum number of seconds the script should wait for a response from the SNMP agent before timing out and raising an event. The default is 5 seconds.

Checking SNMP Service Rights

By default, the Microsoft SNMP Service is typically configured with Read-only permission. This configuration will prevent the SNMPSet Knowledge Script from setting any values. You can check your current configuration for any computer using the Services Control Panel.

To check the rights associated with the Microsoft SNMP Service:

1. In the Services Control Panel, select the Microsoft SNMP Service, and then click the **Security** tab.
2. Verify the rights associated with the community name you are using for the computer are `READ WRITE`.

Changing the Permission Level for the SNMP Service

To change the rights associated with the Microsoft SNMP Service:

1. In the Services Control Panel, select the Microsoft SNMP Service, and then click the **Security** tab.
2. Select the community name you are using, then click **Edit**.
3. Select the `READ WRITE` permission level from the Community rights list.
4. Click **OK**.

12.16 SyncTime

Use this Knowledge Script to synchronize the system time among computers. This script uses the Windows `net time` command to synchronize the system time to a specified Windows computer, such as the Primary Domain Controller in a workgroup. If you do not specify a computer name, the Domain Controller for the local computer is used. This script raises an event if the time synchronization operation fails.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is **Daily**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event?	Set to **y** to raise an event if the time synchronization operation fails. The default is y.
Name of computer with standard time	Specify the name of the computer whose system time you want the target computers synchronized to match. If you do not specify a computer name, this script uses the Domain Controller associated with the managed computer on which the script is running.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the time synchronization operation fails. The default is 12 (yellow event indicator).

12.17 UnixAgentHealthProxy

Run this Knowledge Script on a proxy Windows agent to remotely monitor the health of UNIX agents. This script performs the following tasks:

- Checks the availability of a managed UNIX computer by first sending an ICMP Echo request to the managed UNIX computer. If the remote computer does not respond, this script sends an ICMP Echo request to the managed UNIX computer's default router and raises an event.
- Monitors the health of the UNIX agent by checking a time stamp value created by the UNIX agent. Normally, the UNIX agent creates a time stamp value every 90 seconds. If the age of the time stamp value exceeds the threshold, this script raises an event and restarts the UNIX agent.

Use this script to validate the health of the UNIX agent on a scheduled basis or for diagnostic purposes (for example, if there are gaps in data collection). This script can detect a problem with a remote agent and reliably notify the AppManager administrator.

The remote UNIX computer to be monitored must be accessible through the network from the computer where the proxy Windows agent is installed. To use this script to monitor more than one remote managed UNIX computer, all the computers you want must be accessible using the same **root** user account information.

NOTE: If you specify an incorrect password for the root account when running this script with Secure Shell (SSH) as the connection method to the remote UNIX or Linux computer, the script raises an event that incorrectly states that the login attempt was successful. If you see an event message similar to the event message below, you must update the job properties to specify the correct root password and start the job:

```
Output: Permission denied at /usr/netiq/UnixAgent/bin/UnixAgentHealthProxy.pl
More Info: "SSH login OK to <machine> with root Using SSH/SFTP combination."
```

Resource Objects

A managed UNIX computer where the AppManager UNIX agent is installed. The UNIX agent must be configured to run as the root user account.

Default Schedule

The default interval for this script is **Every 10 minutes**.

To avoid raising false events, do not configure this script to run more frequently than the UNIX agent updates its time stamp. Ideally, the interval should be more than four minutes.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if age of timestamp exceeds threshold?	Select **Yes** to raise an event if the age of the time stamp exceeds the threshold you set. The default is Yes.
Threshold - Maximum age of timestamp	Specify the maximum age a time stamp can attain before an event is raised. The minimum threshold is 3 minutes and the maximum threshold is 99999 minutes. The default is 9 minutes.
Event severity when age of timestamp exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the age of the time stamp exceeds the threshold. The default is 8.
Remote Host Connection	
UNIX computers to monitor	Specify the IP addresses of the remote UNIX computers you want to monitor, separating the addresses with commas and no spaces.
Password for root user account	Specify the root user account password that the proxy agent computer must use to connect to the remote UNIX computer. This is a mandatory field.
Connection Transport	Specify the connection mode between the proxy agent computer and the monitored UNIX computer: ◆ **Telnet/FTP** to connect using Telnet. ◆ **SSH/FTP** to connect using SSH. This script can use either the Secure Shell (SSH) program with root password authentication or Telnet to make a secure connection to the remote UNIX or Linux computer. If you choose to use Telnet, you must supply a non-root user account name and password. **NOTE:** Telnet and FTP send your user name, password, and other information across the network in cleartext, making it easy for others to read this data.

Parameter	How to Set It
Telnet non-root user account	Provide the Telnet non-root user account if you are using Telnet to connect to the monitored UNIX computer. Leave this parameter value blank if you are using SSH to connect to the monitored UNIX computer.
Telnet non-root user password	Provide the Telnet non-root user password if you are using Telnet to connect to the monitored UNIX computer. Leave this parameter value blank if you are using SSH to connect to the monitored UNIX computer.
Restart UNIX agent if age of timestamp exceeds threshold?	Set to **y** to restart the UNIX agent if the age of the time stamp exceeds the threshold you set. The default is y.

13 AMAdmin Knowledge Scripts

The NT category contains Knowledge Scripts that perform administrative tasks for Windows agents and your AppManager site. To run Knowledge Scripts in this category, your user account needs administrator privileges.

From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. In the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	What It Does
AgentConfigMSRestrictions	Configures the agent restrictions for communicating with management servers.
AgentConfigSecurityKey	Updates the security key file on a managed client computer.
AgentConfigSecurityLevel	Configures the security level for the AppManager agent remotely on Windows computers in your network.
ConfigAdminEvents	Monitors the Windows Application log for self-monitoring events raised by the agent services (`NetIQmc` and `NetIQccm`).
AgentSelfMon	Monitors the health of the scripting engine on the AppManager agent on a Windows computer.
ChangeFooter	Changes the graphic and hyperlink that appear at the bottom of each report page.
ConcurrentRpt	Queries the registry on a computer where you have installed a report agent to get or set the number of concurrent reports.
ConfigAdminEvents	Updates the registry with severity settings for AppManager agent and Management Service events
ConfigSiteCommType	Configures communication between managed client computers and the management server.
ConfigSiteNetFlowCtrl	Configures the size and frequency of agent communications with the management server.
DeleteExpiredReports	Deletes expired reports generated by an AppManager report agent.
DisableSiteComm	Temporarily disables network communication from a managed client on Windows to the current management server and saves messages (including events, data, and job status) in the managed client's local repository.
EnableSiteComm	Resumes regular ongoing network communication from a managed client on Windows to the management server.
IISContinueSite	Continues a paused Internet Information Services (IIS) site.
IISPauseSite	Pauses an Internet Information Services (IIS) site.
IISRestartServer	Restarts an Internet Information Services (IIS) server.
IISRestartSite	Restarts an Internet Information Services (IIS) site.

LRReadParameters	Raises an event that displays local repository (LR) configuration information of a managed client computer.
LRRemoveParameters	Removes local repository (LR) configuration information from a managed client computer.
LRWriteParameters	Stores local repository (LR) configuration information in a managed client computer.
MonitorMSCommunications	Monitors the Windows Application log for events that indicate the agent and the management server are not using compatible encryption keys.
MSHealth	Monitors the Windows Application log for events generated by the agent service and the management server.
RemovePrimaryMS	Removes a designated primary management server from a managed client on Windows.
SchedMaint	Sets an application server maintenance period for a managed computer on Windows. During the maintenance period, regularly scheduled AppManager jobs can be prevented from running.
SetAllowMS	Restricts the management servers that can control a particular agent in sites with multiple management servers or multiple repositories.
SetDataTimeStamp	Sets the timestamp for data as it is referenced for reports. This setting affects all reports, but it does not affect areas other than reporting.
SetDeploymentWebService	Sets or changes the deployment Web service with which the managed client communicates to install the agents remotely.
SetKSStandby	Designates a selected managed client as a standby managed client for specified Knowledge Script categories and for the master managed client.
SetLocalRPSize	Modifies the maximum number of events or data points that can be stored in the local repository of a managed client on Windows.
SetPrimaryMS	Sets the primary management server for a managed client on Windows in multiple management server configurations.
SetReportPaths	Sets the output path of the Report Agent. Sets the URL Mapping registry value so that the paths to reports are displayed as hyperlinks in report event messages.
SetResDependency	Defines the resources required to run Knowledge Script jobs on Windows computers.
SiteSchedUpload	Specifies a schedule for uploading data and/or events from the local repository of a managed client on Windows to the current management server. Sets up specific schedules for data, events, or both.
UpgradeJobs	Upgrades all child jobs for a specified parent job on managed Windows or UNIX servers to the latest Knowledge Script version.

13.1 AgentConfigMSRestrictions

Use this Knowledge Script to check for and configure agent restrictions for communicating with management servers. By default, this script raises an event if an agent is configured to allow communication with anonymous management servers.

You should restrict the management servers from which an AppManager agent will accept job requests to ensure that only authorized management servers communicate with the agent.

An *anonymous* management server is a management server with which the agent has not explicitly authorized communication.

The list of management servers with which the agent communicates is stored in the following registry key:

```
\HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\4.0\NetIQMC\Security\AllowMS
```

If the value of this key is ***** (asterisk), the agent allows anonymous communication.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

About Authorizing Management Servers

When you upgrade the agent to AppManager 7.x, the upgrade process allows you to automatically restrict the authorized servers to the designated primary and secondary, or keep the current configuration until you change the agent's designated primary and secondary management server using the SetPrimaryMS script. If you do not change the management server designation during the upgrade, you can use this script after you upgrade to restrict the authorized management servers.

AppManager 7.0 (or later) agents by default are configured to authorize communication with their designated primary and secondary management servers. If you did not designate the primary and secondary management server during installation, you can use this script after installation to restrict the authorized management servers.

Authorizing Management Servers in a Single-Site Configuration

If you are managing a client computer from a single AppManager site (repository), you should restrict the authorized management servers to the agent's designated primary and secondary management server.

Authorizing Management Servers in a Multiple-Site Configuration

Within a site, after you designate an agent's primary and secondary management server, the agent receives job information and sends events and data only to its designated primary or secondary management server. However, if you have more than one AppManager site, you may want to allow the agent to accept job requests from another site. To do so, you can use this script to authorize the management servers from each site.

When allowing the agent to accept communication from additional management servers, make sure you choose the **Append** option to **add** the management servers to the authorized list (instead of replacing the existing list of authorized management servers). This will allow you to run the SetPrimaryMS script on the agent from the other site and properly configure the agent to accept communication from management servers in both sites.

Reading the Current Configuration

If you are unsure of the agent settings, view the current configuration by choosing **Read configuration** from the *Select operation* parameter and selecting an option to raise an event:

- **If an insecure configuration is detected** -- The event message indicates the agent's configuration allows anonymous management server communication. If you choose this option, you can also set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 10.
- **To report current configuration** -- The event message indicates the agent configuration. If you choose this option, you can also set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25.

When reading the current configuration, by default this script raises an event of severity level 10 if an insecure configuration is detected.

This script always raises an event if the job fails.

This script raises an event for each agent to report the agent's configuration. To view the results, click the **Message** tab.

The event message contains the following sections:

- **Current configuration** -- Indicates whether the agent allows anonymous communication (that is, communication with management servers with which the agent has not explicitly authorized communication).

Result	What It Means
Never allow anonymous MS	The agent is configured to restrict anonymous management server communication. You must run this script to allow anonymous management server communication. This result only applies to version 7.0 (or later) agents.
Do not allow anonymous MS at this time	The agent is configured to restrict anonymous communication.
Allow anonymous MS until Primary/Secondary MS is set	The agent is configured to allow anonymous communication until you designate a primary management server. If the agent that has not been configured to have a designated primary server, select this option to secure management server communication after the primary management server is designated.
Allow anonymous MS at this time	The agent is configured to remove restrictions on anonymous management server communication. This setting is not recommended.

- **Specified management servers currently allowed to communicate with this agent** -- Lists the management servers that are authorized to communicate with the agent.

  If this section lists the value as **[Blank]**, the agent does not have an authorized list of management servers with which to communicate. In this case, the agent can still communicate with its designated primary and secondary management server. We recommend that you authorize the agent to communicate with its primary and secondary management server.

Changing the Current Configuration

To change the current configuration, choose **Write configuration** from the *Select operation* parameter and specify the following parameters. By default, this script reads the configuration.

This script always raises an event if the job fails.

Set or change the following parameters as needed:

Parameter	How to Set It
Select operations...	
Read Options	
Raise Event	Set this script to raise an event: ◆ if insecure configuration is detected. When enabled lets you also set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 10 (red event indicator). By default, this option is enabled. ◆ to report current configuration. When enabled, lets you also set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 20 (yellow event indicator). By default, this option is enabled.
Write Options	
Restrict management server communication	Select an option to restrict management server communication: ◆ **Never allow anonymous MS** -- For AppManager 7.0 (or later) agents, this option restricts anonymous management server communication. **TIP:** To configure an agent to allow anonymous communication, run this script and select the **Allow anonymous MS at this time** option. If you choose this option, make sure the agent is configured to authorize communication with at least one management server. This is the default. ◆ **Do not allow anonymous MS at this time** -- This option restricts anonymous communication for all versions of the agent. ◆ **Allow anonymous MS until Primary/Secondary MS is set** -- This restricts anonymous management server communication after the primary management server is designated. If the agent that has not been configured to have a designated primary server, select this option to allow anonymous communication until you designate a primary management server. ◆ **Allow anonymous MS at this time** -- This option removes restrictions on anonymous management server communication. This setting is not recommended.

Parameter	How to Set It
List of authorized management servers	Specify the management servers you want to authorize: ◆ **Management servers to include** -- Specify a comma-separated list of the management servers with which you want the agent to communicate. ◆ **Append or replace current list?** -- Select one of the following options: **Append** to add your specified management servers to the list of authorized management servers. This is the default. **Replace** to remove the existing list of authorized management servers and replace with your specified management servers. ◆ **Management servers to remove** -- Specify a comma-separated list of the management servers with which you do **not** want the agent to communicate.
Event notification	Set this script to raise an event and specify the severity if: ◆ **Configuration succeeds --** Select **Yes** to raise an event if the configuration succeeds. When enabled, lets you also set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 20 (yellow event indicator). By default, this option is enabled. ◆ **Configuration failed** -- Select **Yes** to raise an event if the configuration fails. When enabled, lets you also set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 10 (red event indicator). By default, this option is enabled.

Avoiding Orphaned Agents

If you use this script to remove or replace the list of authorized management servers and the agent is configured to never allow anonymous management server communication, make sure you authorize at least one valid management server. If the agent is configured to never allow anonymous management server communication and the agent is not configured to authorize a management server, the agent cannot be managed by AppManager.

To resolve this problem, manually edit the registry on the managed client computer to specify an authorized management server list in `\HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\4.0\NetIQMC\Security\AllowMS`.

13.2 AgentConfigSecurityKey

Use this Knowledge Script to remotely update the security key information on your managed Windows computers in an AppManager site if you are using encrypted communication or authentication and encryption to secure communication between management servers and managed clients.

Within an AppManager site, all management server computers and managed Windows clients must use the same security key information. This key information is stored in the repository and extracted from the repository into an encrypted and password-protected agent key file. You can then use this script to distribute this password-protected key file to remote agents. If the AppManager repository has different key information than an AppManager agent, agent will not be able to decrypt information from the management server and communication will fail.

NOTE: Use this script only to distribute the key file to managed Windows clients. You must create the key information and the agent key file separately before using this script. In most cases, you create key information when you install the AppManager repository or manually using the NQKeyGenWindows utility. If you have an existing key file generated by the NetIQ Encryption utility (rpckey.exe) in a previous release, you can continue to use that key file and distribute it to AppManager 7.x agents, if needed, until you are ready to replace the old key file with one generated with the NQKeyGenWindows utility.

After you distribute an updated key file to all of the Windows agents, including the agent on the management server computers within your site in your site, you will experience a temporary loss of communication between the management server and the agents. To have the management server receive the new security key information from the repository database and resume communication with the updated Windows agents, you must stop and restart the NetIQ AppManager Management Service (`NetIQms`).

NOTE: If you already distributed an AppManager 7.x key file to your Windows agents, you can use the same key file for both Encrypted and Encrypted and Authenticated communication. You do not need to re-key your Windows agents to change the security level for those agents.

For more information, see “Using Secure Communication for Windows Agents” and “Key File Utility for Windows Agents” in the *Administrator Guide for AppManager*. For more information about configuring security after an AppManager upgrade, see the *Upgrade and Migration Guide for AppManager*.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Location of key file	Provide the full path to the agent key file. For example: `C:\temp\nqWindowsPublic0.key` To specify some other computer in the environment rather than the target computer, type the UNC path to the file. For example, if the key is stored in the `E:\Temp` folder on the computer `zebra`: `\\zebra\e$\temp\nqWindowsPublic0.key`
Encryption password	Provide the password you specified when you created the agent key file. The characters that you type appear as asterisks to protect your password.
Raise event if the update succeeds?	Set to **y** to raise an event when the key is successfully updated on the target computer. The default is y.

Parameter	How to Set It
Event severity when the update succeeds	Set the event severity level, from 1 to 40, to indicate the importance of a successful registration of the management server. The default severity level is 25 (blue event indicator).

13.3 AgentConfigSecurityLevel

Use this Knowledge Script to remotely update the agent security level on the managed Windows computers in your site. When configuring the security level for the agent, keep in mind that all managed Windows clients and management server computers in an AppManager site must be configured to use the same security level. For more information about implementing AppManager secure communication, see the *Administrator Guide for AppManager*.

Use this script to change the security level on the managed Windows clients in your AppManager site either before or after you change the security level on the repository database. The new security level takes effect on the agent as soon as the script completes.

If your repository database is configured to use Encryption or Encryption and Authentication, and you change the security level on the agent to Cleartext, this script will not immediately raise a successful event. In this case, the agent cannot communicate with the management server until you change the security level on the repository database and restart the management server.

The following security levels are available:

- **0 - Cleartext -- no security** indicates that all communication between the agent and the management server is in cleartext and is not encrypted. This option is available for all supported versions of the AppManager agent.
- **1 - Encryption -- medium security** indicates that all communication between the agent and the management server is encrypted but the agent does not authenticate the identity of the management server. This option is available for all supported versions of the AppManager agent.
- **2 - Encryption and authentication -- highest security** indicates that the agent will attempt to authenticate the identity of the management server before sending and receiving encrypted communication. This option is available for version 7.0 (or later) of the AppManager agent.

To use the AgentConfigSecurityLevel script to increase the security level on your Windows agents (for example, from Cleartext to Encryption and Authentication):

1. Use the `NQKeyGenWindows.exe` utility to generate an encryption key file and insert it into the repository database. You can find this utility in the `Program Files\NetIQ\AppManager\bin` directory.

   NOTE: The same key file can be used for both encrypted and encrypted and authenticated communication.

2. If you have not done so already, use the AgentConfigSecurityKey script to distribute the agent portion of the key file to all of your Windows agents, including the agent on the management server computers within your site.
3. Use the `NQKeyGenWindows.exe` utility to change the security level on the repository database.
4. Use the AgentConfigSecurityLevel script to change the security level on the agents, including the agent on the management server computers within your site.
5. Stop and restart the NetIQ AppManager management server service (`NetIQms`) to communicate at the specified security level.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Security level	Select the security level you want the managed Windows computer to use: ◆ **0 - Clear text** if you want all communication between the agent and the management server to be in clear text and is not encrypted. This option is best for closed network environments, testing, or troubleshooting communication issues. ◆ **1 - Encryption** if you want all communication between the agent and the management server to be encrypted but do not require authentication. ◆ **2 - Encryption and authentication** if you want the management server to be authenticated before sending and receiving encrypted communication. Keep in mind that, for a single repository, all managed Windows clients must use the same security level setting. Any time you update security, you must do so for all of your Windows agents. If you cannot update all of your WIndows agents at once, the management server will not be able to communicate with those agents and the interruption in communication may result in missing critical events or data. Therefore, you should plan any change to the security level carefully to minimize the chance of communication failures. The default is 0 - Clear text.
Raise event when update succeeds?	Set to **y** to raise an event when the security level is successfully updated. This script always raises an event if the job does not run successfully. If enabled, you can configure the severity level of the event. The default is y.
Event severity when the update...	Set the event severity level, from 1 to 40, to reflect the importance when the job: ◆ **... succeeds**. If you set this script to raise an event when the job succeeds, set the event severity level for a successful discovery. The default is 25 (blue event indicator). ◆ **... fails**. The default is 5 (red event indicator).

13.4 AgentHealth

Use this Knowledge Script to monitor the status of the AppManager agent, specifically, the Managed Client (MC) and Client Communication Manager (CCM) services. This script looks for self-monitoring events of several types (general, communication, job, security, and upgrade) in the Windows Application log. An event is raised when AppManager places a self-monitoring event in the Windows Application log.

You can filter the event log entries associated with the agent services by specifying a combination of include and exclude strings for each event field. All event log entries that match the filtering criteria are returned in the event detail message.

This script should be run on computers that do not have the Management Server installed. To monitor for self-monitoring events on Management Server computers, use the MSHealth script.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is Asynchronous.

Regardless of the schedule you select, once you start the script, its job status appears as Running.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Monitor events of type...	Set to **y** to raise an event when AppManager places a self-monitoring event in the Windows Application Log of any of the following types: ◆ **... general**. The default is y. ◆ **... communications**. The default is y. ◆ **... job**. The default is y. ◆ **... security**. The default is y. ◆ **... upgrade**. The default is y.
Event severity for events of type...	Set the event severity level, from 1 to 40, to reflect the importance when the following types of event are inserted in the Windows event log: ◆ **... general**. The default is 15 (yellow event indicator). ◆ **... communications**. The default is 15 (yellow event indicator). ◆ **... job**. The default is 15 (yellow event indicator). ◆ **... security**. The default is 15 (yellow event indicator). ◆ **... upgrade**. The default is 15 (yellow event indicator).

Parameter	How to Set It
Filter events by event description	If you set a filter here, the script looks for matching entries in the event log's **Description** field. Multiple strings can be entered, separated by commas. The filter string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you are specifying only include criteria, the colon is not necessary. For example: communication,cold start:mc,ccm
Use case-sensitive description filter?	Set to **y** to make all filter statements for this script case-sensitive. The default value is n (not case-sensitive).

13.5 AgentSelfMon

Use this Knowledge Script to monitor the health of the scripting engine in the AppManager agent on a Windows computer. In some cases, the scripting engine does not run jobs properly but the AppManager agent may respond to remote procedure calls from AppManager diagnostic utilities. In this case, restarting the AppManager agent and the scripting engine resolves the problem. This script is not applicable on UNIX computers.

This script monitors the health of the scripting engine on the agent by running a job that updates a timestamp value in the Windows registry. If the age of the timestamp value exceeds the threshold you specify, the Client Communication Manager service (`netiqccm.exe`) automatically restarts the AppManager agent service (`netiqmc.exe`).

During the first monitoring interval, the Client Communication Manager service (`netiqccm.exe`) writes the current timestamp value to the Windows registry; subsequent job iterations compare and then update the timestamp value.

The timestamp and threshold values are stored in the registry under `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\4.0` as follows:

Registry Entry	Registry Location
Timestamp value that is written each time the job runs	NetIQmc\Admin\LastMCCheck
Threshold, in seconds, for the maximum age of timestamp value	NetIQccm\Admin\MCFreezeThreshold

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every 5 minutes**.

The interval you select must be less than or equal to the value you specify in the *Threshold - Maximum age of timestamp value* parameter.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Threshold - Maximum age of timestamp value	Specify a threshold, in seconds, for the maximum age of the timestamp value. If the elapsed time exceeds the threshold, the Client Communication Manager service (`netiqccm.exe`) automatically restarts the managed client (`netiqmc.exe`). If you specify 0, the current value of the `MCFreezeThreshold` registry key is used. This feature allows you to configure the threshold value a single time. If a value of 0 is specified, the job will not run if the value of `MCFreezeThreshold` is less than the job's scheduled interval. The default value is 0. **NOTE:** If you specify a threshold other than 0, the threshold must be greater than or equal to the scheduled job interval. If the threshold is less than the interval, the job does not run and an event (severity level 40) is raised.

13.6 ChangeFooter

Use this Knowledge Script to change the graphic and hyperlink that appear at the bottom of each report page.

The NetIQ logo is the default graphic and the default hyperlink is to the NetIQ Web site. This script allows you to substitute a different graphic and hyperlink, or to restore those settings to the defaults.

Resource Object

AppManager repository object under the Report agent

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if set or restore operation succeeds?	Set to **y** to raise an events if the script job succeeds in setting a new report footer or in restoring the default report footer. By default, events are raised.
Set new footer or restore default footer	◆ Type **Set a new footer** to change the default settings. This option clears the existing values for the *Full path to new picture file*, *Hyperlink text*, and *URL* parameters. ◆ Select **Restore to NetIQ default footer** to restore the default settings. This option restores the default values for the *Full path to new picture file*, *Hyperlink text*, and *URL* parameters.)

Parameter	How to Set It
Full path to logo image file	Provide the full path to the new picture file you want to use. For example, `C:\LogoFiles\Logo1.gif`.
Hyperlink text	Provide the new text for the hyperlink that appears next to the picture file.
URL for hyperlink	Provide the URL referenced by the hyperlink.
Use actual image size?	Set to **y** to use the actual size of the new footer image. If this parameter is disabled, the image is scaled to the size of the default footer image. The default is y.
Event severity when set or restore operation succeeds	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red event indicator).

13.7 ConcurrentRpt

Use this Knowledge Script to get the setting for the number of concurrent report jobs that can be managed by a report agent, or to reset that registry key to another value. This script queries the registry on a computer where you have installed a report agent to get or set the number of concurrent reports.

Resource Object

AppManager repository object under the Report agent

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if query or set operation succeeds?	Set to **y** to raise events. The default is y. When you query the registry to get the value for the number of concurrent reports, the event detail message contains the current setting.
Query, or set new value?	Set the value to **query** to query the registry and get the value of the registry key. Set the value to **set** to change the value of the registry key. In order for the value to take effect, you must restart the managed client. If you select **set**, use the *Number of concurrent reports* parameter to define the number of concurrent jobs.

Parameter	How to Set It
Number of concurrent reports (optional)	Define the number of concurrent report jobs managed by the report agent. The default is 3. **NOTE:** Setting this value too high can overload the report agent, slow the generation of reports, and adversely affect the performance of the report agent computer.
Event severity when query or set operation succeeds	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

13.8 ConfigAdminEvents

Use this Knowledge Script to update event severity settings in the registry: `HKEY_LOCAL_MACHINE\Software\NetIQ\AppManager\4.0`. By using this script, you do not have to manually update severity settings in the registry for the following AppManager agent (`NetIQmc` and `NetIQccm`) and Management Service (`NetIQms`) events:

- Management Service events
  - `NetIQms` administrative alerts updated in the `\netiqms\admin\AdminEvtSev` registry key
  - `NetIQmc` job failures updated in the `netiqms\config\MC Job Abort Event Sev` registry key
  - Orphaned job events updated in the `\netiqms\config\Orphan Job Event Sev` registry key
  - General success events updated in the `\netiqms\admin\General Success Events` registry key
  - General failure events updated in the `\netiqms\admin\General Failure Events` registry key
- AppManager agent events
  - `NetIQmc` administrative alerts updated in the `\netiqmc\admin\AdminEvtSev` registry key
  - Knowledge Script failure events updated in the `\netiqmc\admin\Knowledge Script Failure Events` registry key
  - `NetIQccm` administrative alerts updated in the `\netiqmc\admin\AdminEvtSev` registry key

This script raises events if Management Service and agent event severity settings are applied to the registry, and if the Management Service is not installed on the agent computer.

NOTE

- For AppManager versions earlier than 8.x, you must restart the `NetIQmc`, `NetIQms`, and `NetIQccm` services after using this Knowledge Script to update the registry. The updated registry settings do not take effect until you restart the services.
- For more information about AppManager registry settings, see the *Administrator Guide for AppManager*, available on the NetIQ AppManager Documentation Web site.

Prerequisite

AppManager version 8.x or later is required to support registry updates for the following events. The registry keys for these events are not created in earlier versions of AppManager.

- General success events
- General failure events
- Knowledge Script failure events

Resource Objects

Windows 2003 Server or later

Default Schedule

By default, this script runs once.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
General Settings	
Job Failure Notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the ConfigAdminEvents job fails. The default is 5.
Apply Management Service severity settings to registry?	Select **Yes** to update the severity settings for the following Management Service events: ◆ Administrative alerts ◆ `NetIQmc` job failures ◆ Orphaned job events ◆ General success events ◆ General failure events The default is Yes.
Event severity for administrative alerts	Set the severity level, from 1 to 40, to indicate the importance of an event in which administrative alerts for the Management Service occur. The default is 40.
Event severity for NetIQmc job failures	Set the severity level, from 1 to 40, to indicate the importance of an event in which the NetIQmc job fails. The default is 10.
Event severity for orphaned job events	Set the severity level, from 1 to 40, to indicate the importance of an event in which job events are orphaned. The default is 5.

Parameter	How to Set It
Event severity for general success events	Set the severity level, from 1 to 40, to indicate the importance of an event in which general success events occur. The default is 35. **NOTE:** This parameter is not supported for versions of AppManager earlier than version 8.x.
Event severity for general failure events	Set the severity level, from 1 to 40, to indicate the importance of an event in which general failure events occur. The default is 5. **NOTE:** This parameter is not supported for versions of AppManager earlier than version 8.x.
Apply agent severity settings to registry?	Select **Yes** to update the registry with severity settings for the following agent events: ◆ NetIQmc administrative alerts ◆ Knowledge Script failures ◆ NetIQccm administrative alerts The default is Yes.
Management Client (NetIQmc)	
Event severity for administrative alerts	Set the severity level, from 1 to 40, to indicate the importance of an event in which NetIQmc administrative alerts occur. The default is 40.
Event severity for Knowledge Script failures	Set the severity level, from 1 to 40, to indicate the importance of an event in which the Knowledge Script fails. The default is 5. **NOTE:** This parameter is not supported for versions of AppManager earlier than version 8.x.
Suppress events with severity equal to	Use this parameter to ignore NetIQmc events with a severity equal to the value you provide. The default is a severity level of 0.
Client Communication Manager (NetIQccm)	
Event severity for administrative alerts	Set the severity level, from 1 to 40, to indicate the importance of an event in which NetIQccm administrative alerts occur.The default is 40.
Event Notification	
Raise event if Management Service severity settings are successfully applied to registry?	Select **Yes** to raise an event if Management Service severity settings are applied to the registry successfully. The default is Yes.
Event severity when settings successfully applied to registry	Set the severity level, from 1 to 40, to indicate the importance of an event in which Management Service severity settings are applied to the registry successfully. The default is 16.
Raise event if Management Service is not installed on agent computer?	Select **Yes** to raise an event if the Management Service is not installed on the agent computer on which you run this script. The default is Yes.
Event severity when Management service is not installed on agent computer	Set the severity level, from 1 to 40, to indicate the importance of an event in which the Management Service is not installed on the agent computer on which you run this script. The default is 25.

Parameter	How to Set It
Raise event if agent severity settings successfully applied to registry?	Select **Yes** to raise an event if AppManager agent severity settings are applied to the registry successfully. The default is Yes.
Event severity when settings successfully applied to registry	Set the severity level, from 1 to 40, to indicate the importance of an event in which AppManager agent severity settings are applied to the registry successfully. The default is 16.

13.9 ConfigSiteCommType

Use this Knowledge Script to configure the AppManager Client Communication Manager service, `NetIQccm`, to use an IP address or hostname to communicate with the management server.

By default, the Client Communication Manager service uses an IP address to locate and communicate with the management server. However, in some environments this may present problems. For example, if your management server and managed clients are connected through a remote dialup connection and use DHCP, IP addresses may be assigned dynamically and change from one connection time to the next, or your management server may be installed on a cluster requiring you to use a cluster name rather than a specific IP address. Use this script to change the default setting for the CCM service.

Configuring AppManager services to use an IP address or hostname is done on a site-by-site basis.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Use IP address to communicate with management server?	Set to **y** to have the `NetIQccm` service use an IP address to establish communication with the management server. Select **n** to have the `NetIQccm` service connect to the management server using a hostname. The default is y.
Raise event if communication configuration succeeds?	Set to **y** to raise an event indicating the success of the operation. The default is n.

Parameter	How to Set It
Event severity when job...	Set the event severity level, from 1 to 40, to reflect the importance of the event when the attempt to set the method used by the agent service to communicate with the management server: **... succeeds**. If you set this script to raise an event when the job succeeds, set the event severity level for a successful discovery. The default is 25 (blue event indicator). **... fails**. The default is 5 (red event indicator).

13.10 ConfigSiteNetFlowCtrl

Use this Knowledge Script to configure the characteristics of NetIQ Corporation AppManager Client Communication Manager service (`NetIQccm`) communication to the management server for the current repository.

This script allows you to control the flow of network traffic from a managed client to the management server by defining upper and lower bandwidth limits for the size of message batches transferred. One batch is sent at each communication interval. You can also set the length of the communication interval in seconds.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Network flow upper limit	Specify an upper bandwidth limit, in KB, for each network message batch from NetIQccm to the management server. The default of 0 indicates no upper limit.
Network flow lower limit	Specify a minimum bandwidth, in KB, for each network message batch from NetIQ Corporationccm to the management server. The default of 0 indicates no lower limit.
Communication interval	Specify the frequency (in seconds) with which NetIQccm can send message batches to the management server. The default is 0 indicates no delay between message batches.
Tune network flow dynamically?	Set to **y** to have the `NetIQccm` agent service dynamically tune and control the size of network message flows to the management server, based on the management server's current load. The default is n.
Raise event if job succeeds?	Set to **y** to raise an event indicating the success or failure of the operation. The default is n.

Parameter	How to Set It
Event severity when job...	Set the event severity level, from 1 to 40, to reflect the importance when the job: **... succeeds**. If you set this script to raise an event when the job succeeds, set the event severity level for a successful discovery. The default is 25 (blue event indicator). **... fails**. The default is 5 (red event indicator).

Example of How this Script Is Used

This script is intended to help you manage network bandwidth and control and tune the transfer of data from managed clients to the management server to suit your network capacity. Using this script, you can restrict the amount of data the `NetIQccm` agent service sends at any one time, as well as the frequency of data transfers.

For example, assume you define an upper limit of 100K, a lower limit of 2K, and a communication interval of one hour (3600 seconds). With this configuration, `NetIQccm` sends up to 100K of data per hour to the management server until the data waiting to be transferred falls below 2K. `NetIQccm` then stores the data in the local repository. At the next interval, if the data to be transferred is greater than 2K, `NetIQccm` resumes sending the data to the management server. If the data package is still below 2K, `NetIQccm` continues to store the data in the local repository until the next interval.

“Dynamic tuning” provides additional flexibility by allowing `NetIQccm` to respond to load changes on the management server. When the *Tune network flow dynamically?* parameter is enabled and the management server becomes busy, `NetIQccm` decreases the amount of data sent and increases the communication interval until the load on the management server is reduced.

Each time the `NetIQccm` service connects to transfer data, it checks the current load on the management server. If load has increased, `NetIQccm` further reduces the amount of data sent in each batch. `NetIQccm` continues to reduce the amount of data sent until the amount of data to be sent falls below the lower limit you set, or until load on the management server decreases, freeing up bandwidth.

13.11 DeleteExpiredReports

Use this Knowledge Script to delete expired reports generated by an AppManager report agent. If you have configured more than one report agent to generate reports to the same location, run this script on one of the report agents to delete all expired reports.

Resource Object

Report agent

Default Schedule

The default interval for this script is **Run once**.

You can set the schedule to periodically check for expired reports.

Setting Parameter Values

Set the following parameters as needed:

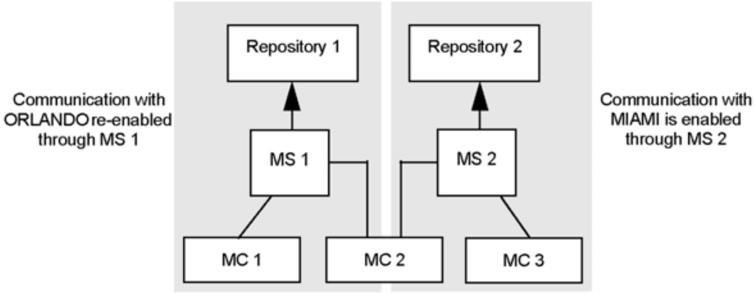
Parameter	How to Set It
Raise event if report successfully deleted?	Set to **y** to raise an event if expired reports are deleted. The default is y.
Generate deletion report?	Set to **y** to generate a report detailing which reports were deleted. The default is y.
Include parameter table?	Set to **y** to include a table in the report that lists parameter settings for the Report script. The default is y.
Select output folder	Set parameters for the output folder. The default folder name is DeleteExpRpts.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. The default is n. The job ID helps you correlate a specific instance of a Report script with the corresponding report.
Select properties	Provide a name for the report and set any other report parameters. The default report name is Expired Reports Deleted.
Add timestamp to title?	Set to **y** to append a timestamp to the title of the report, making each title unique. The default is n. A timestamp allows you to run consecutive iterations of the same report without overwriting previous output.
Event severity when report successfully generated	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Event severity when report has no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Event severity when report generation fails	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

13.12 DisableSiteComm

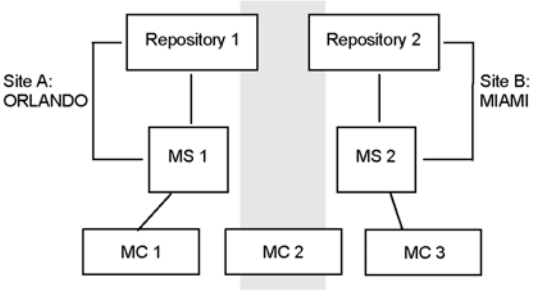
Use this Knowledge Script to temporarily disable network communication from a managed client on Windows to the repository. All messages, including events, data, and job status, are saved in the local repository of the managed client until network communication is re-enabled, at which point it is transferred to the management server. The size of message batches and frequency of delivery can be controlled through the ConfigSiteNetFlowCtrl script.

If a managed client is managed by more than one site (that is, if information for the managed client is stored in more than one repository) you can set *Disable communications for all sites* to **y** to disable communication from the managed client to all repositories with which the managed client communicates.

In this simplified example, MC 2 normally sends data and events to both `ORLANDO` and `MIAMI`.

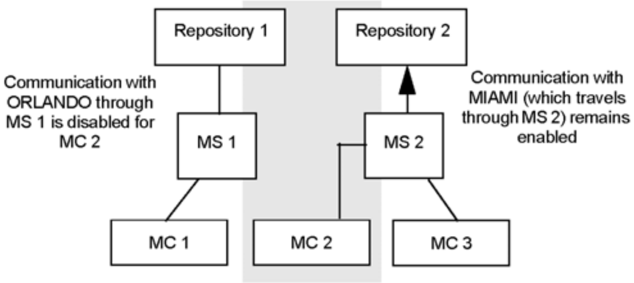

Repository 1
Repository 2
Communication with ORLANDO re-enabled through MS 1
Communication with MIAMI is enabled through MS 2
MS 1
MS 2
MC 1
MC 2
MC 3

If you run this script on MC 2 and enable the *Disable communications for all sites* parameter, communication to both `ORLANDO` (Repository 1) and `MIAMI` (Repository 2) is disabled for MC 2. Communications by MC 1 and MC 3 are not affected.



You can set the *Disable communications for all sites* parameter to **n** to disable communication from the managed client only to the repository you are currently logged onto.

For example, you have logged onto Repository 1 in AppManager (in the Login dialog box). You run this script on MC 2 and set *Disable communications for all sites* to **n**. Communication between MC 2 and Repository 1 is disabled, but communication between MC 2 and Repository 2 is unaffected. Communications by MC 1 and MC 3 are also unaffected.


Repository 1
Repository 2
Communication with ORLANDO through MS 1 is disabled for MC 2
Communication with MIAMI (which travels through MS 2) remains enabled
MS 1
MS 2
MC 1
MC 2
MC 3

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Disable communications for all sites?	Set to **y** to disable communication between the managed client and the repositories with which it communicates. Set to **n** to disable communication from the managed client to the repository you are logged onto. The default is n.
Raise event when attempt to disable communication succeeds?	Set to **y** to raise an event indicating the success of the operation. An event is always raised when the job fails. The default is n.
Event severity when attempt to disable communication...	Set the event severity level, from 1 to 40, to reflect the importance when the job: **... succeeds**. If you set this script to raise an event when the job succeeds, set the event severity level for a successful discovery. The default is 25 (blue event indicator). **... fails**. The default is 5 (red event indicator).

Example of How this Script Is Used

This script lets you intentionally stop the communication between managed clients and repositories. For example, if you are experiencing network problems, you may want to temporarily disable communication while you troubleshoot the problem.

You can also force data and events be stored in the local repository. For example, if you are experiencing high network activity, you can disable communication between the managed client and the management server and store data locally until the demand for server bandwidth is reduced. To set up a regular schedule for uploading events or data from the local repository, use the SiteSchedUpload script.

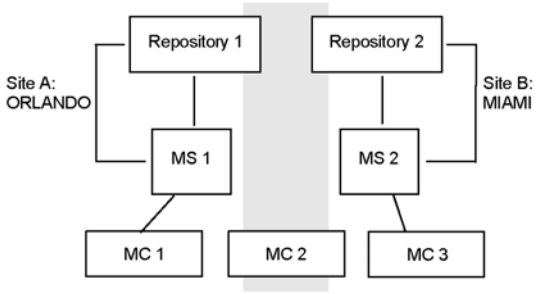
13.13 EnableSiteComm

Use this Knowledge Script to resume regular ongoing network communication from a managed client to the management server. Network communication may be disabled because you have run the DisableSiteComm script to disable network communication between the managed client and management server.

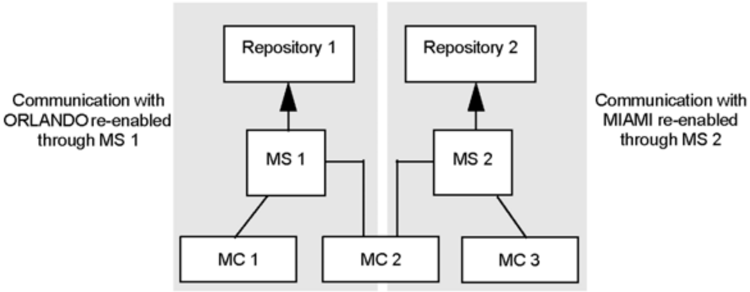
As soon as network communication is restored, any information temporarily stored in the local repository of the managed client while communication was disabled is transferred to the management server, either immediately or in the next scheduled upload if the managed client is configured to transfer events or data according to a schedule.

If a managed client is managed by more than one site (that is, if information for the managed client is stored in more than one repository) you can set the *Enable communications for all sites* parameter to y to enable communication from the managed client to all management sites with which the managed client communicates.

In this simplified example, MC 2 normally sends data and events to both `ORLANDO` and `MIAMI`, but communication with these sites has been temporarily disabled for this managed client.

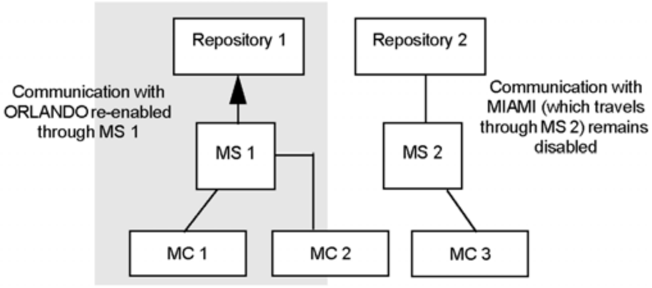


If you run this script on MC 2 and enable the *Enable communications for all sites* parameter, communication to both `ORLANDO` (Repository 1) and `MIAMI` (Repository 2) is re-enabled for MC 2.



You can set *Enable communications for all sites* to n to enable communication from the managed client only for the repository you are currently logged onto.

For example, if you have logged onto Repository 1 in AppManager (in the Login dialog box), and run this script on MC 2 with *Enable communications for all sites* Select **n**, communication between MC 2 and Repository 1 is enabled, but communication between MC 2 and Repository 2 remains disabled.



Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Enable communications for all sites?	Set to **y** to enable communication from the managed client to all repositories the managed client communicates with Set to **n** to enable communication from the managed client to the repository you are logged onto. The default is n.
Raise event when attempt to enable communication succeeds?	Set to **y** to generate an event indicating the success or failure of the operation. The default is n.
Event severity when attempt to enable communication succeeds	Set the event severity level, from 1 to 40, to reflect the importance when the communication succeeds. The default is 25.
Event severity when attempt to enable communication fail	Set the event severity level, from 1 to 40, to reflect the importance when the job fails. The default is 5.

13.14 IISContinueSite

Use this Knowledge Script to continue a paused IIS site remotely. If the script is unable to continue a paused IIS site, an event is raised.

This script is not supported on IIS version 7.x.

Resource Objects

Servers running Windows Server 2003 or Windows XP Professional

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if operation succeeds?	Select **Yes** to raise an event if the attempt to continue the site succeeds. The default is Yes.
Event severity if operation succeeds	Set the severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue event indicator).

Parameter	How to Set It
Raise event if operation fails?	Select **Yes** to raise an event if the attempt to continue the site fails. The default is Yes.
Event severity if operation fails	Set the severity level, from 1 to 40, to indicate the importance of the event. The default is 15 (yellow event indicator).
Event severity for unexpected error	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator). This script raises an event for unexpected script errors. For example, if a script aborts before the job starts or during the job, an event is raised.

13.15 IISPauseSite

Use this Knowledge Script to temporarily pause an IIS site. If the IIS site cannot be paused, an event is raised. This script raises an event if the script is unable to continue a paused IIS site.

This script is not supported on IIS version 7.x.

Resource Objects

Servers running Windows Server 2003 or Windows XP Professional

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if operation succeeds?	Select **Yes** to raise an event if the attempt to pause the site succeeds. The default is Yes.
Event severity if operation succeeds	Set the severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue event indicator).
Raise event if operation fails?	Select **Yes** to raise an event if the attempt to pause the site fails. The default is Yes.
Event severity if operation fails	Set the severity level, from 1 to 40, to indicate the importance of the event. The default is 15 (yellow event indicator).
Event severity for unexpected error	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator). This script raises an event for unexpected script errors. For example, if a script aborts before the job starts or during the job, an event is raised.

13.16 IISRestartServer

Use this Knowledge Script to stop and then restart an IIS server. This script raises events if the attempt to stop or restart a service fails or succeeds. You can detect and start any service that was stopped abruptly.

Resource Objects

Windows 2003 Server or later

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if attempt to restart server succeeds?	Select **Yes** to raise an event if the attempt to restart the IIS server succeeds. The default is Yes.
Event severity if operation succeeds	Set the severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue event indicator).
Raise event if attempt to restart server fails?	Select **Yes** to raise an event if the attempt to restart the IIS server fails. The default is Yes.
Event severity if server start fails	Set the severity level, from 1 to 40, to indicate the importance of the event. This Knowledge script raises this event when the attempt to start the IIS server fails. The default is 5 (red event indicator).
Event severity if server stop fails	Set the severity level, from 1 to 40, to indicate the importance of the event. This Knowledge script raises this event when the attempt to stop the IIS server fails. The default is 5 (red event indicator).
Event severity for unexpected error	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator). This script raises an event for unexpected script errors. For example, if a script aborts before the job starts or during the job, an event is raised.
Administration	
Restart the server?	Select **Yes** to restart the IIS server, after it is stopped. The default is Yes.
Service start/stop delay	Set the number of seconds to wait after the IIS server is stopped before attempting to automatically restart it. The default waiting time is 30 seconds.
Restart dependent services?	Select **Yes** to restart any services that depend on the server you stopped. The default is Yes.
Service start/stop retry count	Set the number of times to attempt to restart a service after it has stopped. The script attempts to restart a service 3 times (default).

13.17 IISRestartSite

Use this Knowledge Script to shut down and restart an IIS site instance. This script raises an event if the IIS site cannot be shut down or restarted.

Resource Objects

Windows 2003 Server or later

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if operation succeeds?	Select **Yes** to raise an event if the attempt to restart the IIS site succeeds. The default is Yes.
Event severity if operation succeeds	Set the severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue event indicator).
Raise event if operation fails?	Select **Yes** to raise an event if the attempt to restart the IIS site fails. The default is Yes.
Event severity if operation fails	Set the severity level, from 1 to 40, to indicate the importance of the event. The default is 15 (yellow event indicator).
Event severity for unexpected error	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator). This script raises an event for unexpected script errors. For example, if a script aborts before the job starts or during the job, an event is raised.
Administration	
Restart site after shutdown?	Select **Yes** to restart the site after shutdown. The default is Yes.

13.18 LRReadParameters

Use this Knowledge Script to view local repository (LR) configuration information on a managed client computer. You can specify the LR parameter or parameters you want to view, or view all parameters. This script raises an event each time you run the job, and the event details display the name and value of each parameter.

LR information consists of parameter values that are stored in the local repository. You can use this script in conjunction with AppManager *_Config* scripts (for example, General_ConfigMachineDown, NT_ConfigServiceDown) to read the configuration information written in the local repository by the *_Config* scripts.

LR information can be stored in the local repository by using the LRWriteParameters Knowledge Script.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if LR configuration retrieved successfully?	Select **Yes** to raise an event if the script is successful in accessing the local repository and reading the parameter values. The default is Yes.
Event severity when LR configuration retrieved successfully	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 15 (yellow event indicator).
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red event indicator). This script raises an event for unexpected script errors. For example, if a script aborts before the job starts or during the job, an event is raised.
Administration	
Retrieve all LR parameters?	Select **Yes** to retrieve all LR information in the AppManager agent's local repository. The default is Yes. If you deselect this check box, you must specify the name of the parameter or parameters you want to retrieve. For example, if there are two parameters in the local repository, **name** and **location**, to read the value for **location**, specify **location** in any of the *Parameter#* parameters. This script can read the value of up to 20 specified parameters each time you run the script.
Parameter 1 ... 20	Specify the name of the parameter or parameters you want to retrieve. Enter the parameter name without quotes. For example, if there are two parameters in the local repository, **name** and **location**, to read the value for **location**, specify **location** in any of the **Parameter#** parameters. If you deselect the **Retrieve all LR parameters?**, you must specify the parameter names. This script can read the value of up to 20 specified parameters each time you run the script.

13.19 LRRemoveParameters

Use this Knowledge Script to remove local repository (LR) configuration information from a managed client computer. Specify the parameter or parameters you want to remove, or remove all parameters from the local repository. This script raises an event each time you run the job. The event details display information about the parameters that were removed.

LR information consists of parameter values that are stored in the local repository. You can use this script in conjunction with the AppManager *_Config* scripts (for example, General_ConfigMachineDown, NT_ConfigServiceDown) to remove the configuration information entered by the *_Config* scripts from the local repository.

To view LR information that is stored in the local repository, use the LRReadParameters script.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if parameters deleted successfully?	Select **Yes** to raise an event if the script is successful in removing the parameter values. The default is Yes.
Event severity when parameters deleted successfully	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 15 (yellow event indicator).
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red event indicator). This script raises an event for unexpected script errors. For example, if a script aborts before the job starts or during the job.
Administration	
Delete entire LR?	Select **Yes** to remove all LR information from the AppManager agent's local repository. The default is unselected. If you select **Yes**, you must specify the name of the parameter or parameters you want to remove. For example, if there are 2 parameters in the local repository, **name** and **location**, to remove the **location** parameter, specify **location** in any of the *Parameter#* parameters. This script can remove up to 20 specified parameters each time you run the script.
Parameter 1 ... 20	Specify the name of the parameter or parameter you want to remove. Enter the parameter name without quotes. For example, if there are 2 parameters in the local repository, **name** and **location**, to remove the **location** parameter, specify **location** in any of the *Parameter#* parameters. If you select the *Delete entire LR?* parameter, you must specify the parameter names. This script can remove up to 20 specified parameters each time you run the script.

13.20 LRWriteParameters

Use this Knowledge Script to store local repository (LR) configuration information in a managed client computer. You can specify a name and value for each parameter you want to store. This script raises an event each time you run the job. The event details display information about the parameters that were set.

LR information consists of parameter values that are stored in the local repository. You can use this script to enter your own information into the local repository for use by custom or customized scripts.

This script contains the same capabilities as AppManager *_Config* scripts, except that you can write any named/value pair to the local repository as opposed to specifically named entries. You can use this script instead of the *_Config* scripts when you use the following named entries (shown with their matching *_Config* scripts):

Named Entry	Knowledge Script
_NQ_PingMachine_MachineList	Client_ConfigPingMachine
_NQ_PingMachine_MachineFile	Client_ConfigPingMachine
_NQ_MachineDown_MachineList	General_ConfigMachineDown
_NQ_MachineDown_MachineFile	General_ConfigMachineDown
_NQ_LogicalDisk_DriveList	NT_ConfigLogicalDisks
_NQ_LogicalDisk_TH_UTIL	NT_ConfigLogicalDisks
_NQ_LogicalDisk_TH_FREE	NT_ConfigLogicalDisks
_NQ_LogicalDisk_TH_XFERS	NT_ConfigLogicalDisks
_NQ_LogicalDisk_TH_READS	NT_ConfigLogicalDisks
_NQ_LogicalDisk_TH_WRITES	NT_ConfigLogicalDisks
_NQ_RemoteService_MachineList	NT_ConfigRemoteServiceDown
_NQ_RemoteService_MachineFile	NT_ConfigRemoteServiceDown
_NQ_RemoteService_ServiceList	NT_ConfigRemoteServiceDown
_NQ_Service_ServiceList	NT_ConfigServiceDown
_NQ_Service_ExcludeList	NT_ConfigServiceDown

To remove LR information from the local repository, use the LRRemoveParameters Knowledge Script.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Event Notification	
Raise event if LR parameters set successfully?	Select **Yes** raise an event if the script is successful in setting local repository configuration information. The default is Yes.
Event severity when LR parameters set successfully	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 15 (yellow event indicator).
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red event indicator). This script raises an event for unexpected script errors. For example, if a script aborts before the job starts or during the job.
Administration	
Overwrite value if it already exists?	Select **Yes** to overwrite a parameter value if it already exists in the AppManager agent's local repository. The default is Yes. **NOTE:** This script is not case-sensitive.
Parameter 1 ... 20	Provide the name and value for each parameter you want to set. Enter the parameter name without quotes. For example, to create a **location** parameter and set it to **San Jose**, under any *Parameter#* parameter, specify **location** in **Name** and **San Jose** in **Value**. If the **location** parameter already exists, by default, the value is overwritten. This script can set up to 20 specified parameters each time you run the script.

13.21 MonitorMSCommunications

Use this Knowledge Script to monitor secure agent communications with the management server.

This script monitors the Windows Application log for events that indicate a mismatch of the encryption key used by the agent (on the monitored computer) and the management server.

This script runs continuously and raises an event when a key mismatch is detected.

NOTE: Run this script on computers where the management server is installed.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default schedule for this script is Asynchronous.

Setting Parameter Values

Set the following parameter as needed:

Parameter	How to Set It
Event severity when key mismatch detected	Set the severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red event indicator).

13.22 MSHealth

Use this Knowledge Script to monitor for self-monitoring events associated with the managed client, Client Communication Manager agent service, and management server. It looks in the Windows Application log for events of several types: general, communication, job, security, and upgrade.

You can filter event log entries by event type and by specifying a combination of include and exclude strings for each event field. All event log entries that match the filtering criteria are returned in the event detail message. An event is raised anytime AppManager places a self-monitoring event in the Windows Application log.

Run this script on computers where the management server is installed. To check for self-monitoring events on computers that do not have the management server installed, use the AgentHealth script.

Resource Object

Windows 2003 Server or later

Default Schedule

The default interval for this script is Asynchronous.

Regardless of the schedule you select, once you start the script, its job status appears as Running.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Monitor events of type...	Set to **y** to raise an event when AppManager places a self-monitoring event in the Windows Application Log of any of the following types: **... general**. The default is y. **... communications**. The default is y. **... job**. The default is y. **... security**. The default is y. **... upgrade**. The default is y.

Parameter	How to Set It
Event severity for events of type...	Set the event severity level, from 1 to 40, to reflect the importance when the following types of event are inserted in the Windows event log: **... general**. The default is 15 (yellow event indicator). **... communications**. The default is 15 (yellow event indicator). **... job**. The default is 15 (yellow event indicator). **... security**. The default is 15 (yellow event indicator). **... upgrade**. The default is 15 (yellow event indicator).
Filter events by event description	The script will look for matching entries in the event log Description field. Multiple strings can be entered separated by commas. The filter string can contain criteria used to include entries, exclude entries, or both. Separate the include and exclude criteria with a colon (:). If you are specifying only include criteria, the colon is not necessary. For example, to include “communication” and “cold start” event types but exclude them when associated with the managed client or CCM agent service, type: communication,cold start:mc,ccm
Use case-sensitive description filter?	Set to **y** to make all filter statements for this script case-sensitive. The default value is n (not case-sensitive).

13.23 RemovePrimaryMS

Use this Knowledge Script to remove the agent’s designated primary and secondary management servers. This script removes the designations for the current site. To remove designations for another site, you must run this script from that site.

For performance reasons, you should always designate an agent’s primary management server and if there is one, a secondary management server, within a site. To change the management server designations for an agent, use the SetPrimaryMS script.

This script does not change the authorized list of management servers with which the agent can communicate.

Before using this script, make sure the agent is configured to authorize communication with a management server by running the AgentConfigMSRestrictions Knowledge Script. If the agent is not authorized to communicate with a management server and you remove the agent’s management server designations, the client computer cannot communicate with AppManager. To resolve this problem, you must manually edit the registry on the managed client computer to specify an authorized management server list in the `\HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\4.0\NetIQMC\Security\AllowMS`.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event when primary and backup management server setting removed?	Set to **y** to raise an event when the managed client's primary and secondary (backup) management servers have been successfully removed. An event is always raised if the job fails. The default is n.
Event severity when primary and backup management server setting removed	Set the event severity level, from 1 to 40, to indicate the importance of the event when the job completes successfully. The default severity level is 25 (blue event indicator).

13.24 SchedMaint

Use this Knowledge Script to specify a period of scheduled maintenance for an application resource (such as WMI) or all resources on a managed client computer on Windows. During the maintenance period, regularly scheduled AppManager jobs for the application resource do not run. You can specify the application resources you want to block by script category, or prevent all jobs from running on a server (for example, because of expected downtime).

This icon, , indicates that a Windows computer is in maintenance mode, or all application resources on a computer are in scheduled maintenance mode. It indicates that AppManager has temporarily stopped monitoring the computer.

- The icon is displayed next to all resources when all application resources for a computer are in scheduled maintenance mode or when a computer is in machine maintenance mode.
- The icon is displayed next to all resource objects on a computer when a particular application resource is in scheduled maintenance mode. Only jobs for the specified application resource are blocked.

You define the start and end time for the scheduled maintenance period under the **Schedule** properties tab. Jobs resume running on the managed computer when the maintenance period expires.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Daily**. However, you should use the Schedule tab to set a schedule appropriate to your environment and maintenance needs.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Knowledge Script category to block	Specify the script category for the jobs you do not want to run during a maintenance period (for example: sql). You can specify either a single category or an asterisk (*) for all jobs. The default is *.
Raise event when schedule implementation succeeds?	Set to **y** to generate an event indicating the success or failure of the operation. The default is n.
Event severity when schedule implementation succeeds	Set the event severity level, from 1 to 40, to indicate the importance of a successful registration of the management server. The default severity level is 25 (blue event indicator).

Example of How this Script Is Used

In many environments, specific application servers have regularly scheduled periods when they are brought down by administrators so administrative tasks can be performed.

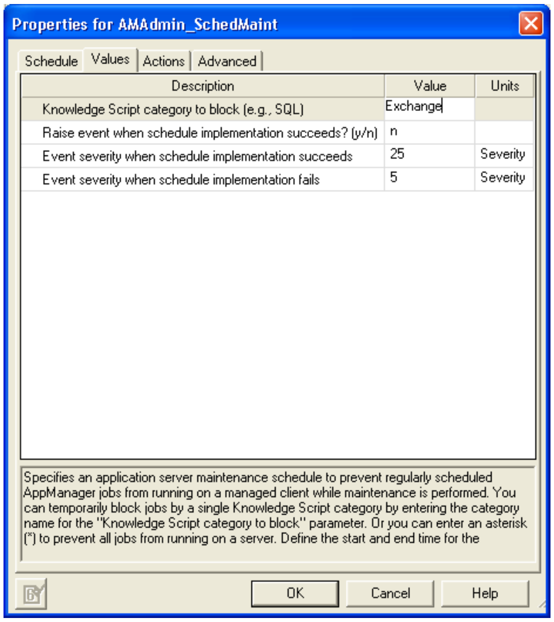
For example, an organization may have 20 Exchange servers that are shut down every Friday at 9 P.M. This interruption causes all of the AppManager Exchange jobs that are not explicitly stopped to error out and forces the administrator to restart the jobs manually when the servers are brought back online.

With this script, administrators can define a specific schedule for temporarily blocking jobs during a planned maintenance period.

Using the Exchange example above, you might set a start time of 8:55 p.m. and an end time of 2:55 a.m. on the Schedule tab. Click **Every** in the Frequency section to set an **End** time. The frequency interval (such as 5 Minutes) is ignored.

On the Values tab, you might identify **exch** (if only Exchange is going to be off-line) or ***** (if the computers are going to be physically shut down) as the script category to block on the Values tab.

For example, to block Exchange Knowledge Script jobs, you might set the parameters on the Values tab similar to the following example:



Properties for AMAdmin_SchedMaint

Description	Value	Units
Knowledge Script category to block (e.g., SQL)	Exchange	
Raise event when schedule implementation succeeds? (y/n)	n	
Event severity when schedule implementation succeeds	25	Severity
Event severity when schedule implementation fails	5	Severity

Specifies an application server maintenance schedule to prevent regularly scheduled AppManager jobs from running on a managed client while maintenance is performed. You can temporarily block jobs by a single Knowledge Script category by entering the category name for the "Knowledge Script category to block" parameter. Or you can enter an asterisk (*) to prevent all jobs from running on a server. Define the start and end time for the

At 8:55 p.m. local time (where the job is running), all Exchange Knowledge Script jobs running on the target computers are stopped. At 2:55 a.m. local time, the maintenance period expires and the Exchange jobs resume running at their regularly scheduled intervals.

13.25 SetAllowMS

Use this Knowledge Script in sites with multiple management servers or multiple repositories to restrict the management servers that can control the agent.

This script sets a registry entry on the agent computer to explicitly allow a managed client to communicate with specified management servers from other management sites. The list of management servers with which the agent communicates is stored in the following registry key:

```
\HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\4.0\NetIQMC\Security\AllowMS
```

An asterisk (*) as a value for the `AllowMS` registry key authorizes all management servers to communicate with the agent. With this setting, “anonymous” management servers, servers with which the agent has not explicitly authorized communication, can communicate with the agent. This represents the lowest-security setting. It is the default if you do not choose to designate a primary management server during agent installation.

This script should not be used to enforce security or control communication between the management server and the managed client within a single site. Within a site, you should designate a primary and, if desired, a secondary management server for each agent. A separate registry key is involved in those designations; you can use the SetPrimaryMS Knowledge Script to identify the primary and secondary management server for each managed client within sites where more than one management server is installed.

You can specify the hostnames of allowed management servers for the *New hostname(s) for AllowMS* parameter. The computers you specify here will not become the agent’s primary or secondary management server, but those computers can communicate with the agent and instruct it to run monitoring jobs.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
New hostname(s) for AllowMS	Specify a comma-separated list of computer hostnames to designate the management servers that are allowed to communicate with this agent. The `AllowMS` registry key will be set with this list as the value. **NOTE:** it is a good idea to use this script to allow management servers from other management sites to use this agent. For management server-to-agent communications within a single site, use the SetPrimaryMS Knowledge Script.
Raise event if attempt to set AllowMS succeeds?	Set to **y** to raise an event if the job succeeds. The default is n.
Event severity when attempt to set AllowMS succeeds	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue event indicator).

13.26 SetDataTimeStamp

Use this Knowledge Script to set the timestamp for data as it is referenced for reports. This setting affects all reports, but it does not affect areas other than reporting.

You can set one of three timestamps:

- **AppManager Repository** uses the local date/time of the AppManager repository computer.
- **Agent** uses the local date/time of the AppManager agent computer.
- **Custom** uses UTC (Coordinated Universal Time) plus or minus *N* hours.

By default, AppManager reports use the local time of the AppManager repository from which the reports are generated. Under circumstances where you want to have an AppManager repository-centric view of your data, you can leave these settings at their defaults.

Under circumstances where the accuracy of your reports depends on data being understood in the context of the local times during which it was collected, you would want to use the **Agent** timestamp. For example, if you are collecting data in four different time zones and want your report to include only data collected between 8 AM and 5 PM, you need that time frame to be relative to each time zone.

If you need to see all your data in the context of a specific time zone, you can use the **Custom** setting, and set the time zone by specifying the number of hours in positive or negative relation to UTC.

Resource Object

Report agent

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if timestamp successfully set?	Select **Yes** to raise an event if a timestamp is set successfully. The default is Yes.
Set timestamp to	Select a timestamp to use in reports: ◆ **AppManager Repository** to set the timestamp to the local time of the AppManager repository. ◆ **Agent** to set the timestamp to the local time of the AppManager agent that collected the data. ◆ **Custom** to set the timestamp to a custom time (UTC plus or minus *N* hours). If you select this option, you must specify the number of hours in the following parameter.
Custom time bias	Specify the number of hours by which UTC is modified. For example, if you enter 8, the time bias is Select UTC plus 8 hours. If you enter -8, the time bias is Select UTC minus 8 hours. Enter 0 to use UTC time.
Event severity when timestamp successfully set	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

13.27 SetDeploymentWebService

Use this Knowledge Script to set or change the hostname of the Deployment Web Service with which the managed client should communicate to install the agents remotely.

NOTE: For details on installing the agents remotely, see the *User Guide for Control Center*.

The Deployment Web Service is normally set during agent installation. If for any reason you did not supply the hostname of the Deployment Web Service during agent installation, or if you need to change the Deployment Web Service that was set for an agent, use this script to set or change it.

The computer that hosts the Deployment Web Service must be accessible over the network via Port 80 to all managed clients.

This script sets the following registry key on the target computer:

```
HKLM\SOFTWARE\NetIQ\AppManager\4.0\AgtShared\DeploymentEndpoint
```

To disable communication with the Deployment Web Service, leave the *Name of Deployment Web Service* parameter blank.

NOTE: After you run this script, the AppManager agent may take up to six hours to report its software inventory to the deployment Web service. Restart the NetIQ AppManager Client Resource Monitor and NetIQ AppManager Client Communication Manager agent services to ensure that the agent reports its software inventory immediately.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event if job succeeds?	Select **Yes** to raise an event if the job succeeds. This script always raises an event if the job fails. the default is unselected.
Event severity when job succeeds	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue event indicator).
Event severity when job fails	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 10 (red event indicator).
Name of Deployment Web Service	Specify the hostname of the Deployment Web Service. To disable remote installation of agents, run the job with this parameter left blank.

13.28 SetKSStandby

Use this Knowledge Script to designate a selected managed client as a standby managed client for specified script categories and for the master managed client.

A *standby* managed client runs jobs only when the master managed client is down, or when jobs from the specified script category are blocked.

If a master managed client is currently configured, leave the *Hostname for master managed client* parameter blank to change it.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Every hour**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Knowledge Script categories	Specify the script categories for which the selected managed client will serve as a standby. Whenever a job from one of these categories cannot run because the agent that is supposed to run it cannot be reached, the job defaults to the standby. Separate the names of multiple script groups with commas and no spaces. The category name is shown on the script view tab for that category. For example, for Microsoft IIS, the category name is "IIS." An asterisk (*) indicates all categories. The default is *.
Hostname for master managed client	Designate the master managed client for the script category specified in the previous parameter by supplying its hostname.
Raise event if job succeeds?	Set to **y** to raise an event if the job succeeds in designating a master managed client to serve as a standby for the specified script categories. The default is n.
Event severity when job succeeds	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue event indicator).

13.29 SetLocalRPSize

Use this Knowledge Script to modify the maximum number of events or data points that can be stored in the managed client's local repository. If the managed client is not able to communicate with the management server for any reason, the local repository for the managed client stores the most recent events and data points up to this limit until communication with the management server is restored.

If the number of events or data points exceeds the limit you have set (for example because of an extended network interruption), the oldest events or data records are lost as new events or data points are recorded.

Setting this registry key to 0 may affect the performance on the managed computer when a large number of records are inserted into the local repository (for example, because the management server is down, communication is disabled, or the managed computer is between scheduled uploads). If you are using ODBC, no changes are required.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Change maximum number of data points to store?	Set to **y** to change the maximum number of data points that can be added to the local repository. The default is y.
Maximum number of data points to store	Specify the maximum number of data points that can be added to the local repository. Enter 0 if you do not want to set a limit on the maximum number of data points. The default is 10,000 data points.
Change maximum number of events to store?	Set to **y** to change the maximum number of events that can be added to the local repository. The default is y.
Maximum number of events to store (0 for no limit)	Specify the maximum number of events that can be added to the local repository. Enter 0 if you do not want to set a limit on the number of events. The default is 10,000 events.
Raise event if repository configuration succeeds?	Set to **y** to raise an event indicating the success or failure of the operation. The default is n.
Event severity when repository configuration...	Set the event severity level, from 1 to 40, to reflect the importance when the job: **... succeeds**. If you set this script to raise an event when the job succeeds, set the event severity level for a successful discovery. The default is 25 (blue event indicator). **... fails**. The default is 5 (red event indicator).

13.30 SetPrimaryMS

Use this Knowledge Script to designate the primary and secondary management server for an agent. If the primary management server fails, the secondary, or backup, management server takes over communication with the managed client until communication with the primary management server resumes.

Within an AppManager management site, the agent only accepts job requests and sends events to its designated management server. During installation, you can designate the agent’s primary and optionally, a secondary management server. For performance reasons, you should always designate the primary and, if there is one, a secondary management server, within a management site.

After installation, use this script to add a secondary management server, or change the agent’s designated primary management server. If you are managing a computer from more than one AppManager site, run this script from another site to designate the primary and secondary management server for that site.

If you are managing a client from more than one management site, run this script from each site to designate the primary and secondary management server for that site. To authorize an agent to communicate with an additional management server, use the AgentConfigMSRestrictions Knowledge Script.

The list of authorized management servers is updated to include the designated primary and secondary management servers. If the agent was configured to not allow anonymous management server communication, that communication restriction goes into effect after you designate the primary and secondary management server.

To improve repository performance, you should always designate a primary management server for each managed client computer in your site. If you cannot designate a primary management server during installation—for example, if the installation program cannot communicate with the management server—you must manually designate the primary management server using this script.

To configure a primary and backup management server for a managed Windows client, run this script and set the *Primary management server hostname* and *Backup management server hostname* parameters. After establishing a primary and backup management server for a managed client, you can also use this script to change the primary management server hostname, the backup management server hostname, or both using the *Select management server operation to perform* parameter:

Designation to Change	How to Change It
The managed client's **primary** management server	Enter a new hostname for the primary management server. Leave the *Backup management server hostname* parameter blank. Set the *Management server operation to perform* parameter to 1.
The managed client's **backup** management server	Enter the hostname of the existing primary management server. Enter a new hostname for the backup management server. Set the *Management server operation to perform* parameter to 2.
Both the **primary** and **backup** management servers	Enter a new hostname for the primary management server. Enter a new hostname for the backup management server. Set the *Management server operation to perform* parameter to 3.

Set the *Management server operation to perform* parameter properly to avoid unexpected behavior. For example, assume you want to establish the computer `BOSTON` as the primary management server, but do not want to make any change to the backup management server. If you run this script with the *Backup management server hostname* blank but inadvertently set the *Management server operation to perform* parameter to 3, the empty *Backup management server hostname* parameter is not ignored. Because you have indicated you want to change both the primary and backup management servers, the blank entry for *Backup management server hostname* is interpreted as authorization for any available management server to act as a backup management server for the target managed client.

For more information about multiple management server configurations, see the *Administrator Guide for AppManager*.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

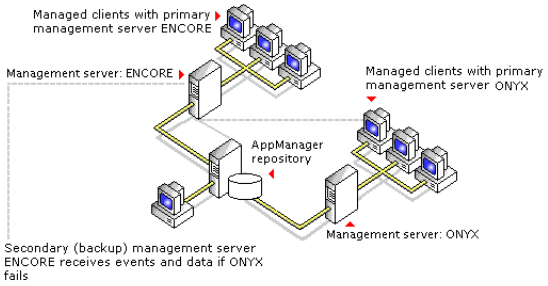
Set the following parameters as needed:

Parameter	How to Set It
Raise event if set operation succeeds or fails?	Set to **y** to raise an informational event when the managed client is successfully updated with the new management server information or if the update fails. The default is n.
Event severity when set operation succeeds	Set the event severity level, from 1 to 40, to indicate the importance of a successful registration of the management server. The default severity level is 25.
Event severity when set operation fails	Set the event severity level, from 1 to 40, to indicate the importance of a event in which the registration of the management server fails. The default severity level is 10.
Primary management server hostname	Specify the name of the management server you want to use as the primary management server. **NOTE:** The value for this parameter cannot be blank, even if you are only setting the backup management server.
Backup management server hostname	Specify the name of the management server you want to use as the backup management server.
Management server operation to perform	Specify which management server configuration you want to update for the target managed client. Type: ◆ **1** to change only the primary management server ◆ **2** to change only the backup management server ◆ **3** to change both the primary and backup management servers The default is 1.

Example of How this Script Is Used

When you install the AppManager agent, you automatically designate a primary management server, and that management server becomes the only management server that the managed client communicates with for a single repository/management server configuration. A secondary or backup management server can also be defined at installation for each managed client in case the primary management server fails. The secondary management server only communicates with the managed client when the primary management server is unavailable. When communication with the primary management server resumes, the managed client resumes exclusive communication with the primary management server.

Because a multiple management server environment is chiefly intended for failover functionality (to provide an alternative management server if the primary management server fails), each managed client can have one primary management server and one backup management server for each repository.


Managed clients with primary management server ENCORE
Management server: ENCORE
Managed clients with primary management server ONYX
AppManager repository
Management server: ONYX
Secondary (backup) management server ENCORE receives events and data if ONYX fails

In addition, identifying a specific management server for specific groups of managed clients gives you greater control over the distribution of communication load and network bandwidth usage.

13.31 SetReportPaths

Use this Knowledge Script to change the default output path used by the report agent.

You can also use this script to instruct the report agent to display the locations of reports as hyperlinks in events. By default, the absolute path to a report is displayed as text on the **Message** tab of the Event Properties dialog box. Use this script to display a hyperlink to the report in addition to the default text. Clicking the hyperlink opens an instance of Internet Explorer to display the contents of the report.

Resource Object

AM Repositories object under the Report agent

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Raise event when change to report path succeeds?	Set to **y** to raise an event when one or both of these paths are successfully changed. The default is y.

Parameter	How to Set It

Base output path	Use this parameter to set a new base output path for the Report Agent.
	Type the path using the following format:
	`\\<server>\<share>\<path to folder>`
	where:
	`<server>` is the computer where you want to write reports.
	`<share>` is the share name of the drive where you want to write reports.
	`<path to folder>` is the absolute path to the folder where you want to write reports.
	For example, the base output path might be:
	`\\RptServer\C$\Program Files\NetIQ\ReportCenter\Web\Report`
	You can also use an absolute path for the value of this parameter. For example:
	`C:\Program Files\NetIQ\ReportCenter\Web\Report`
	NOTE: Leave this parameter blank if you are only using this script to set the *URL mapping* parameter.

URL mapping	Use this parameter to set the URL mapping registry value. Type the URL to the AppManager Web Management Server using the following format: `<protocol name>://<web server name>` where: `<protocol>` is the Internet Protocol, either `HTTP` or `HTTPS`. `<web server name>` is the computer where you have installed the AppManager Web management server. **NOTE:** This parameter should be left blank if you are only using this script to set the *Base output path* parameter. This parameter must be set in order to successfully use Action_SMTPMailRpt.
Event severity when change to report path succeeds	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Event severity when change to report path fails	Set the event severity level, from 1 to 40, to indicate the importance of the event when the job fails. The default is 5 (red level indicator).

13.32 SetResDependency

Use this Knowledge Script to define the resources required to run script jobs on a Windows computer. Resources can include physical file-system related resources such as logical disk drives or directories, or the availability of specific services. You can specify the dependency list by script category or define resources that apply to all script jobs.

The resources and services you specify must be active and available for jobs in the specified category to run. If any resource or service is not available, the jobs in the specified category are temporarily suspended until the specified resource or service becomes available.

Typically, this script is used to define shared cluster resources for physical cluster nodes. For information about running this script in active/passive and active/active cluster environments, see the chapter on cluster support in the *AppManager Administrator Guide*.

This script is used to ensure that jobs do not run when required resources are not available. For example, you may want to check that the Oracle Database services are running before running an Oracle job.

If you are monitoring a cluster environment, you use this script to identify the cluster resources for the active physical node. For example, assume you have an active/passive Exchange 2003 cluster with two physical nodes, `SHASTA` and `VENICE` and that this cluster uses the logical drive `M:` as its shared cluster resource. This shared cluster resource is only available to the active physical node. You use this script to ensure that the Exchange 2003 Knowledge Script jobs only run on the active node by setting the *Knowledge Script category* parameter to `exch` and the *Required available resources* parameter to specify the `M:` drive.

To remove dependencies, you must cold start the AppManager Client Resource Monitor and AppManager Client Communication Manager services using the `-o` start parameter.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Knowledge Script job categories	Indicate the script category for which you want to specify resource dependencies. To specify multiple categories, separate the names with commas and no spaces. The default is all (*) job categories.
Required available resources	Specify the physical resources required for running jobs in the specified category. Physical resources are file-system based and can include logical disk drives, directories, or specific files. You can enter multiple resources, separated by commas with no spaces. For example: `J:,K:,K:\temp\test.log`
Required active services	Specify the Windows services required for running jobs in the specified category. Active services are services that are running when checked. To specify multiple services, separate service names with commas and no spaces: `MSSQLServer,SQLExecutive`
Raise event when update to required resources succeeds?	Set to **y** to raise an event indicating the success of the operation. The default is n.
Event severity when resource update succeeds	Set the event notification level to give you the desired visibility for a successful operation. By default, the severity level is 25 (blue event indicator).

13.33 SiteSchedUpload

Use this Knowledge Script to specify a schedule for uploading data and/or events from the managed client’s local repository on Windows to the current management server. You can set up specific schedules for data, events, or both, as needed.

Depending on your selection, the Client Communication Manager agent service (`NetIQCCM`) stores the events or data points in the local repository until the scheduled upload time. At upload time, the `NetIQCCM` service reads the events and/or data points from the local repository and sends them to the management server. The upload time starts when the job is scheduled to start and ends when the job is scheduled to stop, as specified on the Schedule tab.

The size of message batches delivered in the upload is configured through the ConfigSiteNetFlowCtrl Knowledge Script. You can also configure the maximum number of data points or events to store in the local repository with the SetLocalRPSize Knowledge Script.

Resource Objects

Windows 2003 Server or later

Default Schedule

The default interval for this script is **Run once**. However, you should use the Schedule tab to set a schedule appropriate to your environment.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Schedule upload time for data?	Set to **y** to upload any data points stored on the managed computer to the central repository. The default is y.
Schedule upload time for events?	Set to **y** to upload any events stored on the managed computer to the central repository. The default is n.
Raise event when attempt to set schedule succeeds?	Set to **y** to raise an event indicating the success or failure of the operation. The default is n.
Event severity when attempt to set schedule...	Set the event severity level, from 1 to 40, to reflect the importance when the job: **... succeeds**. If you set this script to raise an event when the job succeeds, set the event severity level for a successful discovery. The default is 25 (blue event indicator). **... fails**. The default is 5 (red event indicator).

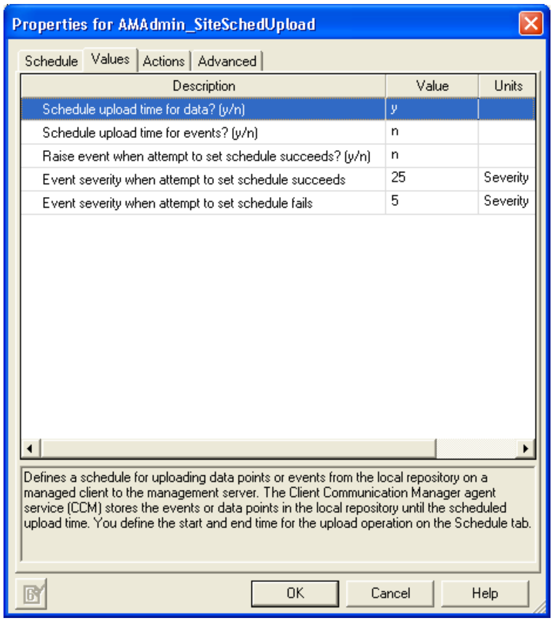
Example of How this Script Is Used

This script allows you to store performance and event data in the local repository until you are ready to upload it to the management server. By giving you the flexibility to transfer events and data during off-peak hours or when network traffic is light, the AppManager management server and repository can handle data from more servers and you can better manage network bandwidth.

For example, if you are collecting a significant amount of data on a few key managed clients, you may want to store the data locally on those managed clients while the network is busy, then transfer it to the management server at a time you know network traffic is light. In addition, you can schedule data from different managed clients to be uploaded at staggered times, further reducing the load on the management server and repository.

To use this script, set a schedule interval, start time, and end time on the **Schedule** properties tab. Click **Every** in the Frequency section to set an **End** time. The frequency interval (such as 5 Minutes) is ignored.

On the **Values** tab, you indicate whether this schedule applies to data, events, or both, and the event visibility. For example:



Description	Value	Units
Schedule upload time for data? (y/n)	y	
Schedule upload time for events? (y/n)	n	
Raise event when attempt to set schedule succeeds? (y/n)	n	
Event severity when attempt to set schedule succeeds	25	Severity
Event severity when attempt to set schedule fails	5	Severity

Defines a schedule for uploading data points or events from the local repository on a managed client to the management server. The Client Communication Manager agent service (CCM) stores the events or data points in the local repository until the scheduled upload time. You define the start and end time for the upload operation on the Schedule tab.

When you run this script on a target, the `NetIQccm` service immediately begins storing the specified information (in this case, data points) from all jobs running on the managed client in the managed client's local repository.

At the scheduled upload Start time (in this case, 1:00 a.m.), the information is transferred to the management server. If all the information in the local repository cannot be transferred to the management server, for example because the upload time is too short, any information not transferred remains in the local repository, up to the maximum number of events or data points that can be stored in the local repository. You can configure the maximum number of events or data points that can be stored in the local repository with SetLocalRPSize).

You can further control the flow of network traffic and the transfer of data from the managed client to the management server using the ConfigSiteNetFlowCtrl Knowledge Script.

13.34 UpgradeJobs

After you upgrade the AppManager agent, existing jobs on the managed client computer are not automatically upgraded to use the latest script functionality. Use this Knowledge Script to upgrade all child jobs for one or more parent jobs.

NOTE: The functionality provided in the latest version of the script may not be supported by older agents with older managed objects. For this reason, you should upgrade your managed clients to the latest version of the AppManager agent before you upgrade jobs running on those agents.

Upgrading jobs to use the latest script version allows the jobs to take advantage of the latest script logic while maintaining existing parameter values for the job, along with the associated graph data and event information. If the latest version of a script has been modified to have new parameters, for example, to create different events or datastreams, the default values in the latest script for the new parameters are used.

This script upgrades all child jobs for one or more parent jobs. You can select the parent jobs you want to upgrade based on the following:

- **Knowledge Script** — Select this option to upgrade all ad hoc jobs started by the specified script. This option upgrades ad hoc jobs started by a particular script and ad hoc jobs started by a Knowledge Script Group member. This option does not upgrade policy-based jobs.
- **Knowledge Script category** — Select this option to upgrade all ad hoc jobs started by the specified script category. This option does not upgrade policy-based jobs.
- **Parent job identifier** — Select this option to upgrade all ad hoc child jobs that belong to the specified Parent Job ID. This option does not upgrade policy-based jobs.
- **Monitoring policy** — All policy-based jobs started by the specified Knowledge Script Group are upgraded. If you are using a Knowledge Script Group in one or more monitoring policies, all affected monitoring policies are updated. This option does not upgrade ad hoc jobs started by a Knowledge Script Group.

NOTE: This script does not upgrade AppManager report Knowledge Script jobs, nor does it return a list of report Knowledge Scripts in an instant check query.

Version Compatibility

This script upgrades the following:

- AppManager jobs on a version 7.0 (or later) Windows agent
- Version 7.0 (or later) AppManager jobs on a version 8.0 (or later) UNIX agent

Performing an Instant Check Query before Running this Knowledge Script

Before you attempt to upgrade jobs using this script, you should identify jobs that have not yet been upgraded by performing an **instant check query**.

The instant check query provides a list of jobs to upgrade and jobs that have already been upgraded. You should use the instant check query to identify the jobs to upgrade and to develop a strategy for upgrading existing jobs.

The instant check query identifies jobs by AppManager version and displays both Windows and UNIX jobs. Use the name of the script category to identify Windows or UNIX jobs.

The query results for each job also include the version of the AppManager agent.

To perform an instant check query, use the **Instant Check Query** parameters on the Values tab. Use the *Select query* parameter to select the type of query you want. Then click **Browse** (**...**) in the *Display query* parameter to see the results of the selected query. To save the query results to a file, click **Finish**. You can run the following query types:

Query Type	Description
Out-of-date parent jobs	This query returns a list of parent IDs that you should upgrade. Note that some parent jobs may contain two different versions of a script. If that is the case, and either one of them is not the latest, the **KS Build ID** field reads “multiple build IDs.”

Query Type	Description
Up-to-date parent jobs	This query returns a list of parent job IDs that are presently using the latest script in the repository and cannot be updated.
Old parent jobs with no upgrade	This query returns a list of jobs with an old script but for which there is no newer version in the repository. If this query returns any parent job IDs, it means the script has either been discontinued in later versions of AppManager, or it is a script you created or customized under a new name and for which you have yet to create a new version in the repository. When this query returns no values, then there are no parent jobs using out-of-date scripts. No further upgrading is required.
Child jobs on v6.x agents	This query returns a list of child jobs running on AppManager version 6.x agents.
Agent build IDs	This query returns a list of the agent build number on each computer. You can use this list to identify agents that you may want to upgrade.
Monitoring-policy jobs	This query returns a list of the jobs that are currently part of a monitoring policy. The jobs are listed according to the view or server group associated with the monitoring policy and then sorted by script group. Note that the Knowledge Script Group names (**KSGName** field) all have the prefix "KSG_." If you want to upgrade a Knowledge Script Group, add this prefix to the group name. You cannot upgrade any UNIX jobs that are policy-based. After you upgrade the backlevel UNIX agent to the latest version, remove the existing backlevel policy-based jobs and recreate them.

After you run an instant check query to identify the jobs you want to upgrade, you can generate a report that previews the jobs that would be upgraded. For more information, see "Viewing Job Upgrade Reports" on page 505.

Upgrading Jobs Created by a Custom Knowledge Script

If you have written a custom script, you do not need to upgrade existing jobs created by that script unless you have made changes to the script. In most cases, existing custom scripts can be run successfully on AppManager 7.0 (and later) agents.

Upgrading Jobs Created by a Copy of a Standard AppManager Knowledge Script

Before you can upgrade jobs created by a copy of a script, you must update the copy of the script in the AppManager repository:

To update a copy of a script:

1. On the repository computer, use Windows Explorer to open the `\netiq\appmanager\qdb\kp` folder and click the folder that contains the new version of the original script upon which the copy is based.
2. Copy the script and rename it to use the same name as the script copy.
3. Check the updated script copy into the repository. You are now ready to upgrade existing jobs.

Verifying Upgraded Jobs

To verify that a job has been upgraded, view the job properties.

To verify a job upgrade:

1. In the List pane in the Operator Console, double-click a child job on the Jobs tab.
2. In the Properties dialog box, click **View KS**.
3. In the Script for Job dialog box, verify the **AppManID** is **Select 7.0**.

Resetting Password Information for Upgraded Jobs

In some rare cases, running the AMAdmin_UpgradeJobs script replaces the existing password for your environment with the default password specified in the original script properties. After these jobs are upgraded, they no longer run because the password is incorrect. This problem occurs for the following scripts:

- NTADMIN_AddUser
- NTADMIN_ChangePassword
- SQL_Bcp

If you upgrade any of these script jobs, update the job properties to restore the correct password information.

Resource Objects

Run this script on a managed Windows computer with the AppManager 7.0 (or later) agent where the “Log On As” account for the AppManager agent Client Resource Monitor (`NetIQmc`) service is a valid domain user account that belongs to the AppManager **Administrator** role.

To verify that the Windows user account that the AppManager agent uses belongs to the AppManager **Administrator** role, in AppManager Security Manager expand **AppManager Roles** in the Navigation pane or the TreeView and click **Administrator** to see a list of valid AppManager administrators.

Default Schedule

The default interval for this script is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
Instant Check Query	
Select query	The instant-check query provides a list of jobs to upgrade and jobs that have already been upgraded. You should use the instant check query to identify the jobs to upgrade and to develop a strategy for upgrading existing jobs. For more information, see “Performing an Instant Check Query before Running this Knowledge Script” on page 500.

Parameter	How to Set It
Display query	Click **Browse [...]** to see a list of Knowledge Script jobs found by the instant check query.
Job Options	After you use *Instant check query* to identify the jobs you want to upgrade, use these parameters to generate a preview of the changes to the script that will be applied to existing jobs and to actually apply those changes and upgrade the jobs.
Upgrade jobs, or generate report?	Select one of the following options: ◆ **Generate report** To preview detailed information about which jobs would be upgraded based on your selection criteria, select this option. If you select this option, no jobs are upgraded. This option provides detailed information about the changes to the actual script, including a list of new or changed parameters. If the latest script has new or changed parameters, you can preview the default values for these parameters before they are applied when you upgrade. ◆ **Upgrade jobs** This option upgrades jobs based on your selection criteria. The default is Generate report. For more information, see "Viewing Job Upgrade Reports" on page 505.
Force, or restricted upgrade?	Select an option for upgrading parent jobs: ◆ **Restricted** This option only upgrades a parent job if all of its child jobs are running on AppManager agents that have been upgraded to the latest version. If one of the child jobs for the specified parent job is running on an older agent, none of the child jobs are upgraded. ◆ **Warning** When using the **Restricted** option, this script does **not** raise an event after unsuccessfully attempting to upgrade jobs on an older agent. Before you select this option, be sure to use the Instant Check Query to verify that there are no jobs running on older agents. ◆ **Force** This option upgrades all of the child jobs for a parent, including child jobs that are running on an agent that has not been upgraded to the latest version. ◆ **Warning** When using the **Force** option, this script does **not** raise an event after unsuccessfully attempting to upgrade jobs on a version 4.3 or 5.0 UNIX agent. If you are upgrading UNIX jobs, be sure to use the Instant Check Query to verify that there are no jobs on version 4.3 and 5.0 UNIX agents. Note that in some cases, the functionality provided in the latest version of the script logic may not be supported by older agents with older managed objects. The default is Restricted.

Parameter	How to Set It
Override job build version?	Set to **y** to upgrade jobs regardless of the job build version (force job upgrade). This option is required to upgrade jobs that have a build version that is the same or earlier than the script build version. The job upgrade process uses **all numbers** of the build version to compare versions. For example, if the build version for an AppManager job is **7.0.2** and the build version of the newer script is **7.0.112**, the job upgrade mechanism would **not** upgrade the job unless you enabled this option. The default is y. **Tip** To view the build version: ◆ For a job, click the **View KS Script** button on the **Values** tab of the Job Properties dialog box. In the Script dialog box, the **AppManID** value specifies the build version. ◆ For a script, in the Operator Console, right-click the script and click **Version History**. In the **Version** dialog box, the build version appears in the **Build ID** column.
Job selection criterion	Specify how to select the jobs you want to upgrade. You can select jobs by: ◆ **Knowledge Script** Select this option to upgrade all ad hoc jobs started by the specified script. This option upgrades ad hoc jobs started by a particular script and ad hoc jobs started by a Knowledge Script Group member. This option does not upgrade policy-based jobs. ◆ **Knowledge Script Category** Select this option to upgrade all ad hoc jobs started by the specified script category. This option does not upgrade policy-based jobs. ◆ **Parent Job Identifier** Select this option to upgrade all ad hoc child jobs that belong to the specified parent job. This option does not upgrade policy-based jobs. ◆ **Monitoring policy** All policy-based jobs started by the specified Knowledge Script Group are upgraded. This option does not upgrade ad hoc jobs started by a Knowledge Script Group. **Warning** If you are using a Knowledge Script Group in more than one monitoring policy, all affected monitoring policies are updated.
Job selection specification	Select the jobs you want to upgrade from a list, based on a job selection criterion. Click **Browse** (**...**) to see the list. If your job selection criterion is: ◆ **Knowledge Script**, this list allows you to select the scripts you want to upgrade. This list only displays scripts that have a corresponding ad hoc job. ◆ **Knowledge Script Category**, this list allows you to select the script categories you want to upgrade. This list only displays script categories that have a corresponding ad hoc job. ◆ **Parent Job Identifier**, this list allows you to select one or more parent jobs. This list only displays parent job identifiers for ad hoc parent jobs. ◆ **Monitoring policy**, this list allows you to select all policy-based jobs that belong to the specified Knowledge Script Group. This list only displays Knowledge Script Groups that belong to a monitoring policy. **Warning** If you are using a Knowledge Script Group in one or more monitoring policies, all monitoring policies are updated.

Viewing Job Upgrade Reports

Each time you run this script, job upgrade reports are created under:

```
\netiq\temp\netiq_debug\computer\jobupgrade
```

where *computer* is the name of the computer where you ran the report. The following reports are always generated regardless of whether you configure this job to generate a report or upgrade jobs:

- `Upgradejob_`*id*`.txt, where` *id* is the UpgradeJobs ID, provides information about which jobs are upgraded.
- `Upgradejob_`*id*`.rpt`, where *id* is the UpgradeJobs job ID, provides detailed information about each job.

TIP: `Upgradejob_`*id*`.log`, `where` *id* is the UpgradeJobs ID, lists the Job IDs that are upgraded and references the corresponding `.rpt` file and `.log` files for more information.

If the child of a specified parent job is running on an agent that has not been upgraded to the latest version, and you specified the **Restricted** upgrade option, the `UpgradeJob_<`*id*`>.txt` file displays information similar to the following:

```
Connected to SQL Server : RACKR14 repository QDB.
Time stamp: 03/03/07 14:20:47
  [Child Job] [Parent Job] [Build ID]  [Computer\KS]
2 4.3 agent(s) found.
2 5.0 agent(s) found.
1 5.0.1 agent(s) found.
Parent job 436 is skipped because under restricted mode, there cannot be any
non-7.0 agents.
Upgrade is finished.
Please check upgradejob_1343.rpt and upgradejob_1343.log located in
D:\NetIQ\Temp\NetIQ_Debug\RACKR14\jobupgrade.
Time stamp: 03/03/07 14:20:47
```

If the child of a specified parent job can be upgraded with parameter changes, the `UpgradeJob_<`*id*`>.rpt` file displays information similar to the following:

```
Connected to SQL Server : RACKR14 repository QDB.
Time stamp: 03/03/07 15:14:30
***************************************************************
Parent job 54 can be upgraded under force mode.
2 4.3 agent(s) found.
2 5.0 agent(s) found.
2 5.0.1 agent(s) found.
1)
Child job ID = 55
Parent job ID = 54
KS name = NT_CpuLoaded
Machine name = RACKN08
Version = 4.6
Job 55 can be upgraded.
The following parameters in the existing job are not found in the new version
of the KS:
1) Event? (y/n)
Existing value is y.
2) Collect Data? (y/n)
Existing value is y.
3) Overall Load? (y/n)
Existing value is y.
```

```
4) Cpu Threshold >
Existing value is 0
5) Cpu Queue Length >
Existing value is 0
6) Event Severity
Existing value is 5
7) Severity for an unexpected KS error
Existing value is 35
The following parameters in the new version of the KS are not found in the
existing job:
1) Event Notification
Default value is NULL.
2) Create event if total system CPU is high?
Default value is y.
3) Severity - Total system CPU
Default value is 5
4) Create event if any individual CPU is high?
Default value is n.
5) Severity - Individual CPU
Default value is 15
6) Severity - Job failure
Default value is 35
7) Data Collection
Default value is NULL.
8) Collect total system utilization data?
Default value is y.
9) Collect individual processor utilization data?
Default value is n.
10) Collect processor queue data?
Default value is y.
11) Monitoring
Default value is NULL.
12) Threshold - Total system CPU
Default value is 0
13) Threshold - Individual CPU
Default value is 98
14) Threshold - Processor queue length
Default value is 0
Check for OldParameter tag
1) Create event if total system CPU is high?
Default value is y
OldParameter tag value = ?DO_EVENT="y" ((AND)) DO_OVERALL="y":"y":"n".
New StringValue = "y"
2) Severity - Total system CPU
Default value is 5
OldParameter tag value = ?DO_EVENT="y" ((AND))
DO_OVERALL="y":Severity:$default$.
New IntValue = "5"
3) Create event if any individual CPU is high?
Default value is n
OldParameter tag value = ?DO_EVENT="y" ((AND)) DO_OVERALL="n":"y":"n".
New StringValue = "n"
4) Severity - Individual CPU
Default value is 15
OldParameter tag value = ?DO_EVENT="y" ((AND))
DO_OVERALL="n":Severity:$default$.
No matching value, will keep original.
5) Severity - Job failure
Default value is 35
OldParameter tag value = PRM_KSERR.
```

```
New IntValue = "35"
6) Collect total system utilization data?
Default value is y
OldParameter tag value = ?DO_DATA="y" ((AND)) DO_OVERALL="y":"y":"n".
New StringValue = "y"
7) Collect individual processor utilization data?
Default value is n
OldParameter tag value = ?DO_DATA="y" ((AND)) DO_OVERALL="n":"y":"n".
New StringValue = "n"
8) Collect processor queue data?
Default value is y
OldParameter tag value = DO_DATA.
New StringValue = "y"
9) Threshold - Total system CPU
Default value is 0
OldParameter tag value = ?DO_OVERALL="y":TH_UTIL:$default$.
New IntValue = "0"
10) Threshold - Individual CPU
Default value is 98
OldParameter tag value = ?DO_OVERALL="n":TH_UTIL:$default$.
No matching value, will keep original.
11) Threshold - Processor queue length
Default value is 0
OldParameter tag value = TH_QLEN.
New IntValue = "0"
```

If the child of a specified parent job cannot be upgraded because the UNIX agent on which it is running is version 6.5, the entry looks like this:

```
Parent job 1536 cannot be upgraded under restricted mode.
29 6.5 agents are found.
Please upgrade these agents and restart the upgrade process.
```

In this case, upgrade the agent or use the *Force upgrade* parameter to upgrade the jobs on the older agent.

14 ReportAM Knowledge Scripts

The ReportAM category provides the following AppManager Knowledge Scripts for generating reports based on data collected by Knowledge Script jobs.

From the Knowledge Script view of Control Center, you can access more information about any NetIQ-supported Knowledge Script by selecting it and clicking **Help**. In the Operator Console, click any Knowledge Script in the Knowledge Script pane and press **F1**.

Knowledge Script	Description
AgentMaintenance	The maintenance history of any computer on which you installed an AppManager agent.
AggValueHistory	Average, minimum, or maximum values aggregated by hour, day, week, or month over a specified time period.
ApplicationInfo	The monitored applications on computers in your AppManager environment.
AvgMaxMinValue	The average, maximum and minimum values of datastreams collected by Knowledge Script jobs.
AvgValueByDay	The average daily value of datastreams collected by Knowledge Script jobs.
AvgValueByHr	The average values per hour of datastreams collected by Knowledge Script jobs.
AvgValueByMin	The average values per minute of datastreams collected by Knowledge Script jobs.
Chart2HTML	Converts the contents of an XML file to an HTML page containing a chart and/or table of AppManager repository data.
Compare24Hours	Twenty-four average, minimum, or maximum hourly values for today, last week, last month, and the last three months.
Compare24HoursLD	Twenty-four average, minimum, or maximum hourly values for today, last week, last month, and the last three months. Use this report for data sets over 1.5 GB.
CompDeploy	The total number of instances of each AppManager component installed on computers in an AppManager site.
CompLic	Summary of AppManager license compliance.
CompVersion	The version number of AppManager components installed on all computers in an AppManager site.
CurrentDiskSpaceUsage	The used and free space on logical disks.
DataStream	High-level information about datastreams collected by Knowledge Script jobs.
DataSummary	Statistical analysis of selected datastreams.

Knowledge Script	Description
DeletedObjects	Objects that have been permanently deleted from the Navigation pane or the TreeView.
DetailData	Information returned by Knowledge Scripts that prepare data for presentation in an XML format.
DFSSummary	Details of the Distributed File System service on specified computers.
EventArchiveSummary	Summary of events from the ArchiveEvent table in the AppManager repository.
EventSeveritySummary	Number and severity of events raised by Knowledge Script jobs on specified computers.
EventStatisticsSummary	Summary of events per computer, listed by monitored application.
EventSummary	Summary of events per computer.
FRSSummary	Details of the File Replication service on specified computers.
GeneralCounter	Average, maximum or minimum value of selected datastreams.
GeneralMachineDown	Computers that were detected as down during a specified time period.
GroupPolicySummary	Summary of Group Policy settings for specified computers.
Inventory	Details of a specified application on a computer, or all components of a computer.
JobInfo	Details of Knowledge Script jobs.
JobSummary	Knowledge Script job, monitoring policy, action, and event information for each computer in an AppManager repository.
LastDataPoint	Value of the most recently collected data point in a datastream.
ModuleUsage	The number of different AppManager modules in use.
NetworkInterface	Details of the network interface components on specified computers.
NTLogicalDisk	Details of the logical disks on specified computers.
NTPhysicalDisk	Details of the physical disks on specified computers.
PerfOverview	A separate chart for each selected datastream.
PerfOverviewLD	A separate chart for each selected datastream. Use this report for data sets over 1.5 GB.
PlainDataInfo	Details of data points in specified datastreams.
PrinterSummary	Details of the printers and printer drivers installed on specified computers.
SerLevAvailability	Percentage of time specified services were up or down.
SQLDBInfo	Details of the databases managed by SQL Servers on specified computers.
SystemUpTime	Up- and downtime (by percent) of monitored computers.
SystemUpTimePie	Up- and downtime (by percent) of monitored computers. This report uses only a pie chart.

Knowledge Script	Description
WatchList	Top or bottom N computers (by number or percent) generating the selected datastreams.

14.1 AgentMaintenance

Use this Knowledge Script to generate a report about the maintenance history of any computer on which you installed an AppManager agent. The report lists the type of maintenance (scheduled or ad hoc), and the beginning and ending dates and times of the maintenance period.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer name.
Select time range	Filter data in your report by a specific or sliding time range. The default is Sliding
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	

Parameter	How To Set It
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.2 AggValueHistory

Use this Knowledge Script to generate a report from data in the archive and aggregate tables.

If you choose to move data from the archive table to the aggregate tables, that data is then aggregated by hour, day, and month.

Briefly, the aggregation process works as follows:

- A preference is specified for the number of months' worth of data to keep in the archive table (for example, the three most recent months' worth).

  For all older data, an hourly average, minimum, maximum, sum, and count value are calculated. Those hourly values are then moved to the hourly aggregate table: ArcAvgHourlyData. Each hour's worth of data is then represented by five data points: average, minimum, maximum, sum, and count.
- The hourly aggregate table keeps three months' worth of data.

  For all older data, a daily average, minimum, maximum, sum, and count value are calculated. Those daily values are then moved to the daily aggregate table (ArcAvgDailyData). Each day's worth of data is then represented by five data points (average, minimum, maximum, sum, and count).
- The daily aggregate table keeps six months' worth of data.

  For all older data, a monthly average, minimum, maximum, sum, and count value are calculated. Those monthly values are then moved to the monthly aggregate table (ArcAvgMonthlyData). Each month's worth of data is then represented by five data points (average, minimum, maximum, sum, and count).
- The monthly aggregate table keeps data indefinitely.

Once information is moved from the source table to the destination table, it is deleted from the source table.

This script lets you generate a report that gives the hourly, daily, weekly, or monthly average, minimum or maximum value for selected datastreams over the time range you specify (for example, the monthly average of memory used by SQL Server processes over the last year).

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.
Select the style	Select the style for the first page of the report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **By Knowledge Script** provides links to pages showing all datastreams collected by a Knowledge Script (each page shows all datastreams collected from all computers on which the script has run) ◆ **All datastreams on one page** generates a report with all data on a single page
Select time range	Filter the data in your report by a specific or sliding time range. The default is Sliding.
Select average, minimum, or maximum	Select the type of value you want to represent in your report.
Aggregation interval	Select the time period by which the data in your report is aggregated: ◆ Hourly ◆ Daily ◆ Weekly ◆ Monthly
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.

Parameter	How To Set It
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.3 ApplicationInfo

Use this Knowledge Script to generate a report detailing the monitored applications on computers in your AppManager environment. Details include application and build numbers, installation directories, and path and log information.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	

Parameter	How To Set It
Select computers	Filter the data in your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.4 AvgMaxMinValue

Use this Knowledge Script to generate a report detailing the average, maximum and minimum values of datastreams collected by Knowledge Script jobs.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Select peak weekday(s)	Filter the data for your report by the days of the week.
Aggregation interval	Select the number of hours by which the data in your report is aggregated.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Series style (average)	Select a graphical style for the average value series in the charts in your report.
Chart title	Provide a title for the charts in your report.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).

Parameter	How To Set It
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.5 AvgValueByDay

Use this Knowledge Script to generate a report detailing the average daily value of datastreams collected by Knowledge Script jobs.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.
Select the style	Select the style for the first page of the report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **By Knowledge Script** provides links to pages showing all datastreams collected by a Knowledge Script (each page shows all datastreams collected from all computers on which the script has run) ◆ **All datastreams on one page** generates a report with all data on a single page
Select time range	Filter the data for your report by a specific or sliding time range.
Select peak weekday(s)	Filter the data for your report by the days of the week.
Aggregation interval	Select the number of days by which the data in your report is aggregated.

Parameter	How To Set It
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.6 AvgValueByHr

Use this Knowledge Script to generate a report detailing the average values per hour of datastreams collected by Knowledge Script jobs.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.
Select the style	Select the style for the first page of the report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **By Knowledge Script** provides links to pages showing all datastreams collected by a Knowledge Script (each page shows all datastreams collected from all computers on which the script has run) ◆ **All datastreams on one page** generates a report with all data on a single page
Select time range	Filter the data in your report by a specific or sliding time range. The default is Sliding.
Select peak weekday(s)	Filter the data in your report by days of the week.
Aggregation interval	Select the number of hours by which the data in your report is aggregated.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Select output folder	Set parameters for the output folder.

Parameter	How To Set It
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.7 AvgValueByMin

Use this Knowledge Script to generate a report detailing the average values per minute of datastreams collected by Knowledge Script jobs.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.

Parameter	How To Set It
Select the style	Select the style for the first page of the report: - **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) - **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) - **By computer and datastream** provides links to pages showing a single datastream collected from a computer - **By Knowledge Script** provides links to pages showing all datastreams collected by a Knowledge Script (each page shows all datastreams collected from all computers on which the script has run) - **All datastreams on one page** generates a report with all data on a single page
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Select peak weekday(s)	Filter the data for your report by days of the week.
Aggregation interval	Select the number of minutes by which the data in your report is aggregated.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: - **Table** (table only) - **Chart** (chart only) - **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	

Parameter	How To Set It
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.8 Chart2HTML

Use this Knowledge Script to generate an HTML page containing a chart and/or table of the data referenced in an XML file. Export data from a report to an XML file using the **Report to XML** command on the Export menu in the AppManager Chart Console.

To use this script successfully, the XML file must be saved to a folder accessible by the report agent.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
XML input file	Specify the full path to the .XML file generated from the AppManager Chart Console.
Select output folder	Set parameters for the output folder.
Chart type	Select the type of data display you want in the report: ◆ Chart (chart only) ◆ Data (table only) ◆ Chart and Data (chart and table)

Parameter	How To Set It
Time frame	Select the time frame for data in the report: - All (all data referenced by the XML file) - Today (any data from the day the report script is run -- 12 A.M. to 11:59:59 P.M.) - Yesterday (any data from the day before the report script is run -- 12 A.M. to 11:59:59 P.M.) - This Week (any data from the week during which the report script is run -- 12 A.M. Sunday to the time of the report) - This Month (any data from the month during which the report script is run -- 12 A.M. of the first day to the time of the report) - This Year (any data from the year during which the report script is run -- 12 A.M. January 1 to the time of the report)
Fit into one graph?	Set to **y** to fit all data into a single graph. The default is y.
Report title	Provide a title for the report.
Severity for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.9 Compare24Hours

Use this Knowledge Script to generate a report that compares the average, minimum or maximum values per hour of selected datastreams for periods of 1 day, 1 week, 1 month, and 3 months. You can limit the scope of the report by selecting data from specific days of the week (for example Monday through Friday only).

If you expect the data sets from which you are deriving values to exceed 1.5 GB, use the Compare24HoursLD Knowledge Script.

The time periods illustrated in the report are:

- Today
- Last Week
- Last Month
- Last 3 Months

For example:

- Today = 4-1-06, 12 A.M. to 11:59:59 P.M.
- Last Week = 3-24-06, 12 A.M. to 3-30-06, 11:59:59 P.M. (the previous seven days)
- Last Month = 3-1-06, 12 A.M. to 3-31-06, 11:59:59 P.M. (all days of the previous month; for example, if Today is any day in April, then Last Month = all days in March)
- Last 3 Months = 1-1-06, 12 A.M. to 3-31-06, 11:59:59 P.M. (all days of the previous three months; for example, if Today is any day in April, then Last 3 Months = all days in January, February, and March)

Twenty-four values are given for each time period, from 12 A.M. to 11 P.M. For each time period, the values are based on different quantities of data. For example:

- 12 A.M. Today is an average of one hour's worth of data (12 A.M. to 12:59:59 A.M. for one day)
- 12 A.M. Last Week is an average of seven hours' worth of data (12 A.M. to 12:59:59 A.M. for each day of the week)
- 12 A.M. Last Month is an average of 31 hours' worth of data (12 A.M. to 12:59:59 A.M. for each day of the month)
- 12 A.M. Last 3 Months is an average of 90 hours' worth of data (12 A.M. to 12:59:59 A.M. for each day of the three months)

This report always compares these four time periods, and so there is no option to select the time range.

All time periods are relative to the day you start the report.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.
Select peak weekday(s)	Filter the data for your report by days of the week.
Select average, minimum, or maximum	Select the type of value you want to represent in your report.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: - **Table** (table only) - **Chart** (chart only) - **Both** (table and chart)
Select chart style (Today)	Define the graphic properties of the chart in your report, and for the chart series representing values for the Today time period.

Parameter	How To Set It
Series style (Last Time Periods)	Select the series style for the last three time periods represented in the report. This parameter lets you select a series style different from the Today series in order to easily differentiate the time periods.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.10 Compare24HoursLD

Use this Knowledge Script instead of Compare24Hours when the data sets from which you are deriving values exceed the ADO limit of 1.5 GB. When data sets exceed this size limit, the report agent cannot process the data. Use this script to process data on the SQL Server managing the AppManager repository and return the aggregated value to the report agent.

Use this script to generate a report that compares the average, minimum or maximum values per hour of selected datastreams for periods of 1 day, 1 week, 1 month, and 3 months. You can limit the scope of the report by selecting data from specific days of the week (for example Monday through Friday only).

The time periods illustrated in the report are:

- Today
- Last Week
- Last Month
- Last 3 Months

For example:

- Today = 4-1-06, 12 A.M. to 11:59:59 P.M.
- Last Week = 3-24-06, 12 A.M. to 3-30-06, 11:59:59 P.M. (the previous seven days)
- Last Month = 3-1-06, 12 A.M. to 3-31-06, 11:59:59 P.M. (all days of the previous month; for example, if Today is any day in April, then Last Month = all days in March)
- Last 3 Months = 1-1-06, 12 A.M. to 3-31-06, 11:59:59 P.M. (all days of the previous three months; for example, if Today is any day in April, then Last 3 Months = all days in January, February, and March)

Twenty-four values are given for each time period, from 12 A.M. to 11 P.M. For each time period, the values are based on different quantities of data. For example:

- 12 A.M. Today is an average of one hour's worth of data (12 A.M. to 12:59:59 A.M. for one day)
- 12 A.M. Last Week is an average of seven hours' worth of data (12 A.M. to 12:59:59 A.M. for each day of the week)
- 12 A.M. Last Month is an average of 31 hours' worth of data (12 A.M. to 12:59:59 A.M. for each day of the month)
- 12 A.M. Last 3 Months is an average of 90 hours' worth of data (12 A.M. to 12:59:59 A.M. for each day of the three months)

This report always compares these four time periods, and so there is no option to select the time range.

All time periods are relative to the day you start the report.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.
Select peak weekday(s)	Filter the data for your report by days of the week.
Select average, minimum, or maximum	Select the type of value you want to represent in your report.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.

Parameter	How To Set It
Include table/chart/both?	Select the type of datastream values you want to include in the report: - **Table** (table only) - **Chart** (chart only) - **Both** (table and chart)
Select chart style (Today)	Define the graphic properties of the chart in your report, and for the chart series representing values for the Today time period.
Series style (Last Time Periods)	Select the series style for the last three time periods represented in the report. This parameter lets you select a series style different from the Today series in order to easily differentiate the time periods.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.11 CompDeploy

Use this Knowledge Script to generate a report detailing the total number of instances of each AppManager component installed on computers in an AppManager site (for example, the number of IIS and Exchange managed objects).

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.12 CompLic

Use this Knowledge Script to generate a report summarizing AppManager license compliance.

A separate license is required for each managed application on each computer. For example, if you are using AppManager to manage Microsoft SQL Server on ten different computers, then you are required to have ten licenses for AppManager for Microsoft SQL Server. If you are managing multiple instances of an application running on a single computer, such as multiple instances of SQL Server, only one license is required for that computer.

This report lists the number of AppManager licenses you have for a particular application (Number of Permanent Licenses) and the number of different computers on which you have discovered that application (Number of Permanent Licenses in Use). You are out of compliance if the number of permanent licenses in use exceeds the number of permanent licenses.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.13 CompVersion

Use this Knowledge Script to generate a report detailing the version number of AppManager components installed on all computers in an AppManager site.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).

Parameter	How To Set It
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.14 CurrentDiskSpaceUsage

Use this Knowledge Script to generate a report detailing the used space on logical disks.

This report uses data collected by any Knowledge Script that monitors the available MB or the percentage of used space on a logical disk, such as NT_DiskSpace or UNIX_FileSystemSpace. Ensure you archive data detail when running the Knowledge Scripts to collect data. You must disable the *Do not archive data detail* option in the Advanced tab of the Knowledge Script properties dialog box to allow automatic data archiving.

NOTE: In the *Filter Settings* parameters, if you set the *Filter column* parameter to **Drive** and the *Filter operator* parameter to either **Greater than** or **Less than**, the script will not run but will raise an event indicating you made an invalid configuration of the script. For example, a drive can be equal to/not equal to C:, but not greater than/less than C:.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.

Parameter	How To Set It
Select type	Select the method by which report content is ordered. Possible values are: ◆ By computer ◆ By drive ◆ By total space ◆ By space used ◆ By percent used ◆ By space free ◆ By percent free For example, if you select By computer, then data is ordered by computer name. If you select By drive, data is ordered by drive letter.
Sort order	Select whether values are displayed in an ascending or descending order.
Filter Settings	
Filter column	Use this parameter to set the first variable of the filtering equation. Use this parameter in conjunction with the *Select type* parameter. Make sure the two parameters have comparable values (for example, if *Select type* is set to *By drive*, set this parameter to *Drive*). If *Select type* is set to *By computer*, set this parameter to <*None*>. Possible values are: ◆ <None> ◆ Drive ◆ Total space ◆ Space used ◆ Percent used ◆ Space free ◆ Percent free
Filter operator	Select an operator for the filtering equation. Possible values are: ◆ Greater than ◆ Less than ◆ Equal to ◆ Not equal to
Filter value	Specify a value for the second variable of the filtering equation. For example, if the first variable is *Drive*, and the operator is *Equal to*, set this parameter to *C:* to return data for all C: drives.
Report settings	
Disk space units	Select whether you would like the values in the report expressed in MB or GB.

Parameter	How To Set It
Include parameter help card?	Select **Yes** to include a table in the report that lists parameter settings for the report script. The default is Yes.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Select **Yes** to append the job ID to the name of the output folder. A job ID allows you to correlate a specific instance of a Report Script with the corresponding report. The default is unselected.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Select **Yes** to append a time stamp to the title of the report, making each title unique. The time stamp is made up of the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is Yes.
Event notification	
Event for report success?	Select **Yes** to raise an event when the report is successfully generated. The default is Yes.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.15 DataStream

Use this Knowledge Script to generate a report containing high-level information about datastreams collected by Knowledge Script jobs. High-level information includes the script names, datastream legends, and job IDs.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Select Knowledge Script(s)	Filter the data for your report by Knowledge Script.
Select job(s)	Filter the data for your report by specific Knowledge Script jobs.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID allows you to correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.16 DataSummary

Use this Knowledge Script to generate a report containing a statistical summary of selected datastreams.

The data wizard allows you to select any combination of datastreams for a report. Depending on which datastreams you select and which style you select, your report can contain meaningful, easily-understood information, or it can contain information that is of no value.

The following examples of selecting datastreams and report styles show the different types of information reports can contain in each case.

By computer style

If you select different datastreams that use *different* units of measure, the report contains a separate value for each datastream from each computer.

For example, if you select the following datastreams:

- `Ldsk: D:USED %`
- `Ldsk: D:AVAIL MB`

the report contains one value for each computer for the percentage of used space on the D: drive, and one value for each computer for the available megabytes on the D: drive.

If you select different datastreams that use the *same* unit of measure, the report contains one value for each computer.

For example, if you select the following datastreams:

- `Ldsk: D:USED %`
- `MemPhysUsage %`

the report contains a single percentage value for each computer. Each single percentage value is derived from all memory usage and disk usage values taken together. In this case, the values in the report are meaningless.

By legend style

If you select multiple datastreams with the same legend from two different computers, the report contains one value for each different legend.

For example, if you select the following datastreams:

- `Ldsk: D:USED %`
- `Ldsk: D:AVAIL MB`

the report contains one value for the percentage of used space on both D: drives, and one value for the available megabytes on both D: drives. This type of report would be useful, for example, if you wanted an overall statistic of disk space availability or memory use for something like a server farm.

If you select multiple datastreams with different legends, the report contains one value for each legend.

For example, if you select the following datastreams:

- `Ldsk: D:USED %`
- `MemPhysUsage %`

the report contains a single percentage value for each legend. One percentage value is derived from disk usage on both computers, the other from memory usage on both computers. As in the previous example, this type of report is useful for overall statistics.

By computer and legend style

The *by computer and legend* style lets you get individual values for each datastream from each computer.

Using this style, if you select the following datastreams:

- `Ldsk: D:USED %`
- `Ldsk: D:AVAIL MB`

the report contains one value for each datastream for each computer.

If you select the following datastreams:

- `MemPhysUsage %`
- `Ldsk: D:USED %`

the report contains one value for each datastream for each computer.

Regardless of the style you select, the table in the report always shows the average, minimum, maximum, and count values for a datastream. It may show additional values, as well, depending on how you configure the report.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Select peak weekday(s)	Filter the data for your report by days of the week.
Select the style	Select the style for the report: - **By computer** shows values for each computer you selected. - **By legend** shows one value for each different legend. - **By computer and legend** shows one value for each unique legend from each computer.
Data settings	

Parameter	How To Set It
Statistics to show	Select a statistical method by which to display data in the report: ◆ **Average**: The average value of data points for the aggregation interval (for example, the average value for 1 Hour) ◆ **Minimum**: The minimum value of data points for the aggregation interval ◆ **Maximum**: The maximum value of data points for the aggregation interval ◆ **Min/Avg/Max**: The minimum, average, and maximum values of data points for the aggregation interval ◆ **Range**: The range of values in the datastream (maximum - minimum = range) ◆ **StandardDeviation**: The measure of how widely values are dispersed from the mean ◆ **Sum**: The total value of data points for the aggregation interval ◆ **Close**: The last value for the aggregation interval ◆ **Change**: The difference between the first and last values for the aggregation interval (close - open = change) ◆ **Count**: The number of data points for the aggregation interval
Select sorting/display option	Select whether data is sorted, or the method of display: ◆ **No sort**: Data is not sorted ◆ **Sort**: Data is sorted by value (lowest to highest from front to back; highest to lowest from left to right) ◆ **Top %**: Chart only the top N % of selected data (sorted by default) ◆ **Top N**: Chart only the top N of selected data (sorted by default) ◆ **Bottom %**: Chart only the bottom N % of data (sorted by default) ◆ **Bottom N**: Chart only the bottom N of selected data (sorted by default)
Percentage/count for top/bottom	Specify a number for either the percent or count defined in the previous parameter (for example, Top 10%, or Top 10). The default is 25.
Truncate top/bottom?	If set to yes, then the data table shows only the top or bottom N or % (for example, only the top 10%). Otherwise, the table shows all data. The default is no.
Show totals on the table?	If set to yes, then additional calculations are made for each column of numbers in a table, and the following values are listed at the end of the table: ◆ **Report Average**: An average of all values in a column ◆ **Report Minimum**: The minimum value in a column ◆ **Report Maximum**: The maximum value in a column ◆ **Report Total**: The total of all values in a column The default is no.

Parameter	How To Set It
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table/chart/both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to yes to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is no.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **yes** to append a time stamp to the title of the report, making each title unique. The time stamp is made up of the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is no.
Event notification	
Event for report success?	Set to yes to raise an event when the report is successfully generated. The default is yes.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.17 DeletedObjects

Use this Knowledge Script to generate a report about objects that have been permanently deleted from the Navigation pane or the TreeView. These are objects that have been deleted with the **Do not rediscover** option.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.18 DetailData

Use this Knowledge Script to generate a report containing information returned by Knowledge Scripts that prepare data for presentation in an XML format. If the data is not in an XML format, use the PlainDataInfo script.

You can use this report for any script that collects and displays data details in an XML format.

In order to have detail data available for this report, the *Collect data details with data point* option must be set in one of the following ways:

- for each relevant AppManager repository (**File > Preferences > Repository tab > Knowledge Script options > Advanced Properties**)
- on the Advanced properties tab of any individual Knowledge Script for which you want to generate a report.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select Knowledge Script	Filter the data for your report by Knowledge Script. If you have not collected data using a supported Knowledge Scripts, the browser is blank. If you change the name of a Knowledge Script, the new name appears in the browser.
Select computer(s)	Filter the data for your report by computer.
Select time range	Filter the data for your for report by a specific or sliding time range. The default is Sliding.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. The default is n. A job ID helps you correlate a specific instance of a Report Script with the corresponding report.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. The default is n. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output.
Event notification	

Parameter	How To Set It
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.19 DFSSummary

Use this Knowledge Script to generate a report containing details of the Distributed File System (DFS) service on specified computers. Details include the image path and services upon which DFS depends.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data in your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.

Parameter	How To Set It
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.20 EventArchiveSummary

Use this Knowledge Script to generate a report containing a summary of events per computer. The summary includes event IDs and statuses, Knowledge Scripts that raised the events, and event messages. The data for this report is taken from the ArchiveEvent table in the AppManager repository, which contains any archived event information you saved according to your AppManager repository preferences.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computers	Filter the data for your report by computer.
Select Knowledge Scripts	Filter the data for your report by Knowledge Script. The default is All.
Select jobs	Filter the data for your report by a specific Knowledge Script job. The default is All.
Event status	Filter the data for your report by event status. The default is All.

Parameter	How To Set It
Select time range for last occurrence	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Report settings	
Limit the size of event detail	Specify the maximum number of characters to display in the event detail message. The default is 200.
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder. The default folder name is ArchivedEventSummary.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired. The default report name is Archived Event Summary.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.21 EventSeveritySummary

Use this Knowledge Script to generate a report containing the number and severity of events raised by Knowledge Script jobs on specified computers.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.22 EventStatisticsSummary

Use this Knowledge Script to generate a report summarizing events per computer. Events are listed by monitored application. The total event count for each application is listed, as well as the count for each event status: open, acknowledged, closed.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.23 EventSummary

Use this Knowledge Script to generate a report containing a summary of events per computer. The summary includes event IDs and statuses, names of the Knowledge Scripts that raised the events, and event messages.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Select Knowledge Script(s)	Filter the data for your report by Knowledge Script.
Select job(s)	Filter the data for your report by specific Knowledge Script jobs.
Event status	Filter the data for your report by event status.
Select time range for last occurrence	Filter the data for your report by a specific or sliding time range. If the last occurrence of the event does not fall within this range, the report will have no data. The default is Sliding.
Report settings	
Limits the size of event detail	Specify the maximum number of characters to display in the event detail message. The default is 200.
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).

Parameter	How To Set It
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.24 FRSSummary

Use this Knowledge Script to generate a report containing details of the File Replication Service (FRS) on specified computers. Details include image path and services upon which FRS depends.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	

Parameter	How To Set It
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.25 GeneralCounter

Use this Knowledge Script to generate a report containing a chart and table showing the average, maximum or minimum value of each selected datastream.

This report allows you to set the maximum number of data points illustrated in the chart, regardless of the number of data points that have been collected. For example, you may be collecting data every five minutes, but you want to report on the daily maximum value for the last week. Set the time range to 7 days, and set the maximum number of points per chart to 7. The report aggregates the data in increments of one day, and the chart illustrates seven maximum values for each selected datastream. The table in the report mirrors the chart settings: in this case, the table has seven rows of data.

The number of points in the chart do not have to correspond exactly to the time period on which you are reporting. You can illustrate a year's worth of data using 50 or 100 points, and you can illustrate a day's worth of data using 50 or 100 points. This script aggregates the data in the report according to the time range and points per chart settings.

This feature is useful for illustrating any time period in a single chart.

Resource Objects

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.

Parameter	How To Set It
Select the style	Select the style for the first page of the report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **By Knowledge Script** provides links to pages showing all datastreams collected by a Knowledge Script (each page shows all datastreams collected from all computers on which the script has run) ◆ **All datastreams on one page** generates a report with all data on a single page
Select time range	Filter the data for your report by a specific or sliding time range.
Select peak weekday(s)	Filter the data for your report by days of the week.
Maximum number of points per chart	Specify the maximum number of data points illustrated in the chart. The default is 200. Possible values range from 5 to 1000.
Select average, minimum or maximum	Select the type of value you want in the report.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.

Parameter	How To Set It
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.26 GeneralMachineDown

Use this Knowledge Script to generate a report about computers that were detected as down during a specified time period.

This report uses data collected by the UNIX_PingMachine and General_MachineDown Knowledge Scripts.

Resource Objects

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer. **NOTE:** You must select the computers on which the UNIX_PingMachine or General_MachineDown Knowledge Scripts were run.
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.

Parameter	How To Set It
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.27 GroupPolicySummary

Use this Knowledge Script to summarize Group Policy settings for specified computers.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Report settings	

Parameter	How To Set It
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.28 Inventory

Use this Knowledge Script to generate a report containing details of a specified application on a computer, including version number, root directory and security settings, or all components of a computer, including memory and disk configuration, and details of any applications monitored by AppManager.

This report contains information from a single AppManager repository.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select application	Select the application that is the subject of your report. Select **All Components** to report on all applications on specified computers.
Select computer(s)	Filter the data for your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.29 JobInfo

Use this Knowledge Script to generate a report containing the details of Knowledge Script jobs.

The first page of the report lists the specified jobs by computer, and gives the job ID and status of each job. The job IDs are links to pages containing further details about each job. Details include the schedule for each job and the parameter settings.

NOTE: Run this script only on a management server computer.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Select job status	Filter the data for your report by job status.
Select job(s) (optional)	Filter the data for your report by specific Knowledge Script jobs. **NOTE:** If you enable this parameter, do not enable the next parameter. You cannot successfully implement both optional parameters.
Select Knowledge Script(s) (optional)	Filter the data for your report by Knowledge Script. **NOTE:** If you enable this parameter. You cannot successfully implement both optional parameters.
Sort by	Select a sorting method for the contents of the first page of the report: ◆ **Computer** sorts the contents alphabetically by computer name ◆ **Job ID** sorts the information for each computer numerically by job ID ◆ **Knowledge Script Name** sorts the information for each computer alphabetically by Knowledge Script name ◆ **Job Status** sorts the information for each computer alphabetically by job status
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.

Parameter	How To Set It
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.30 JobSummary

Use this Knowledge Script to generate a report about the Knowledge Script jobs, monitoring policies, actions, and events associated with each computer you are monitoring. The report includes all computers in a given AppManager repository.

The first page of the report lists each computer followed by the number of Knowledge Script jobs, monitoring policies, actions, and events associated with that computer. The computer names are links to pages containing further details about each Knowledge Script job on that computer. These details include:

- The job ID
- The Knowledge Script name
- The names of Knowledge Script Groups that include the Knowledge Script
- The names of any Action Knowledge Scripts initiated by the Knowledge Script job
- The number of events raised by the Knowledge Script job

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Report settings	
Include parameter help card?	Set to **yes** to include a table in the report that lists parameter settings for the report script. The default is yes.
Include table?	Set to **yes** to include a table of datastream values in the report. The default is yes.
Include chart?	Set to **yes** to include a chart of datastream values in the report. The default is yes.
Select chart style	Define the graphic properties of the charts in your report.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **yes** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is no.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **yes** to append a time stamp to the title of the report, making each title unique. The time stamp is made up of the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is no.
Event notification	
Event for report success?	Set to **yes** to raise an event when the report is successfully generated. The default is yes.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.31 LastDataPoint

Use this Knowledge Script to generate a report about the value of the most recently collected data point in a datastream.

This report is useful for monitoring conditions such as database size or disk space; conditions where the current state of affairs is of most interest.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.
Select the style	Select the style for the first page of the report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By Knowledge Script** provides links to pages showing all datastreams collected by a Knowledge Script (each page shows all datastreams collected from all computers on which the script has run) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer)
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Select output folder	Click the **Browse [...]** button to set parameters for the output folder.

Parameter	How To Set It
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.32 ModuleUsage

Use this Knowledge Script to generate a report about the number of different AppManager modules in use (for example, the number of different SQL Servers on which you are running Knowledge Script jobs). The report contains:

- The module name
- The number of module licenses required
- The number of modules in use

You can use this report in conjunction with the CompLic Knowledge Script to identify differences between the number of module licenses and the number of modules deployed.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.33 NetworkInterface

Use this Knowledge Script to generate a report detailing the network interface components on specified computers. Details include the manufacturer, IP address, and subnet mask.

Resource Objects

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.34 NTLogicalDisk

Use this Knowledge Script to generate a report detailing the logical disks on specified computers. Details include file systems, and drive sizes (in MB).

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.35 NTPhysicalDisk

Use this Knowledge Script to generate a report detailing physical disks on specified computers. Details include disk sizes (in MB), cylinders per disk, and tracks per cylinder.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.36 PerfOverview

Use this Knowledge Script to generate a report containing a single average value for each selected datastream for a specified increment of time, for example, an average value derived from one hour's worth of data or 30 days' worth of data.

If you expect the data sets from which you are deriving values to exceed 1.5 GB, use PerfOverviewLD instead of this script.

A separate chart is generated for each selected datastream.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.
Select the style	Select the style for the first page of the report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By Knowledge Script** provides links to pages showing all datastreams collected by a Knowledge Script (each page shows all datastreams collected from all computers on which the script has run)
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Select peak weekday(s)	Filter the data for your report by days of the week.

Parameter	How To Set It
Aggregation by	Select the method by which data in the report is aggregated. The options are: - Minute - Hour - Day This parameter is used in conjunction with the following parameter. For example, if you choose to aggregate by day, then the following parameter determines how many days' worth of data are aggregated.
Aggregation interval	Select the interval at which the data in your report is aggregated. This parameter is used in conjunction with the previous parameter. For example, if the **Aggregation by** parameter is set to **Day**, use this parameter to set the number of days by which data is aggregated (1 gives you a single value for one day's worth of data, 2 gives you a single value for two days' worth of data, and so on).
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: - **Table** (table only) - **Chart** (chart only) - **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).

Parameter	How To Set It
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.37 PerfOverviewLD

Use this Knowledge Script in place of PerfOverview when the data sets from which you are deriving values exceed the ADO (ActiveX Data Objects) limit of 1.5 GB. When data sets exceed this size limit, the report agent cannot process the data. Use this script to process data on the SQL Server managing the AppManager repository and return the aggregated value to the report agent.

The report contains a single average value for each selected datastream for a specified increment of time, for example, an average value derived from one month’s worth of data or one year’s worth of data.

A separate chart is generated for each selected datastream.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.
Select the style	Select the style for the first page of the report: ◆ **By computer** provides links to pages showing the data collected from individual computers (each page shows all the datastreams collected from a single computer) ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By Knowledge Script** provides links to pages showing all datastreams collected by a Knowledge Script (each page shows all datastreams collected from all computers on which the script has run)
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Select peak weekday(s)	Filter the data for your report by the days of the week.

Parameter	How To Set It
Aggregation by	Select the method by which data in the report is aggregated. The options are: ◆ Hour ◆ Day This parameter is used in conjunction with the following parameter. For example, if you choose to aggregate by day, then the following parameter determines how many days' worth of data are aggregated.
Aggregation interval	Select the interval at which the data in your report is aggregated. This parameter is used in conjunction with the previous parameter. For example, if the **Aggregation by** parameter is set to **Day**, use this parameter to set the number of days by which data is aggregated (1 gives you a single value for one day's worth of data, 2 gives you a single value for two days' worth of data, and so on).
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.38 PlainDataInfo

Use this Knowledge Script to generate a report listing the details of data points in specified datastreams. Details include the data point value, and the time at which the data point was collected. If the data is presented using an XML format, use the DetailData Knowledge Script.

In order to have detail data available for this report, the *Collect data details with data point* parameter must be set in one of the following ways:

- for each relevant AppManager repository (**File > Preferences > Repository tab > Knowledge Script options > Advanced Properties**)
- on the Advanced properties tab of any individual Knowledge Script for which you want to generate a report.

Resource Objects

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select Knowledge Script	Filter the data for your report by Knowledge Script.
Datastream	Specify the legend of the datastreams that you want to include in your report. To return all datastreams, enter ALL.
Select computer(s)	Filter the data for your report by computer.
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.

Parameter	How To Set It
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.39 PrinterSummary

Use this Knowledge Script to generate a report detailing the printers and printer drivers installed on specified computers.

Resource Objects

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.

Parameter	How To Set It
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.40 SerLevAvailability

Use this Knowledge Script to generate a report detailing the percentage of time specified services were up or down.

This report uses data collected by the NT_ServiceDown and General_ServiceDown Knowledge Scripts. In order to have accurate data for this report, schedule these Knowledge Scripts to run every five minutes.

If you are using NT_ServiceDown, set the *Collect data?* parameter to **y**, and the *Collect data only on down?* parameter to **n**, so that you are always collecting data, rather than collecting data only when a service is down.

Uptime and downtime are calculated during scheduled maintenance. Ad hoc maintenance is considered as downtime, and is included in all calculations.

NOTE: This script expects a certain number of data points per time period based on the parameter settings of the Knowledge Script collecting data. If any data points are missing, the corresponding times are considered as downtime. For example, if a Knowledge Script is configured to collect 12 data points per hour, but only collects six, then one half hour is considered downtime. Data points may be missing, for example, if the Knowledge Script job was stopped and restarted, or if the agent was not running for that period.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select data wizard	Select the data for your report by Knowledge Script or by datastream.
Select the style	Select the style for the first page of the report: ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** generates a report with all data on a single page
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Select peak weekday(s)	Filter the data for your report by the days of the week.
Aggregation interval	Select the time period by which the data in your report is aggregated: ◆ Hourly ◆ Daily ◆ Weekly
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.

Parameter	How To Set It
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.41 SQLDBInfo

Use this Knowledge Script to generate a report detailing the databases managed by SQL Servers on specified computers. Details include the database names and owners, and database and log sizes (in MB).

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Select output folder	Set parameters for the output folder.

Parameter	How To Set It
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.42 SystemUpTime

Use this Knowledge Script to generate a report detailing the uptime and downtime of monitored computers. Uptime and downtime are illustrated in hours and minutes, as well as the percentage of the monitoring interval during which a computer is running or not. For example, if during a 24-hour monitoring interval, the computer is running for 18 hours and not running for six hours, the uptime and downtimes are represented as:

- Uptime: 18 hours 0 minutes
- Downtime: 6 hours 0 minutes
- Uptime: 75%
- Downtime: 25%

This report uses data collected by the NT_SystemUpTime and UNIX_SystemUpTime Knowledge Scripts. In order to have accurate data for this report, schedule these Knowledge Scripts to run every 5 minutes.

Uptime and downtime are calculated during scheduled maintenance. Ad hoc maintenance is considered as downtime, and is included in all calculations.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Select the style	Select the style for the first page of the report: ◆ **By computer and datastream** provides links to pages showing a single datastream collected from a computer ◆ **All datastreams on one page** generates a report with all data on a single page
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Select peak weekday(s)	Filter the data for your report by the days of the week.
Aggregation interval	Select the time period by which the data in your report is aggregated: ◆ Hourly ◆ Daily ◆ Weekly
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include Table/Chart/Both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.

Parameter	How To Set It
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.43 SystemUpTimePie

Use this Knowledge Script to generate a report detailing the uptime and downtime of monitored computers. This report illustrates uptime and downtime using a pie chart. You can enter a minimum threshold for uptime. Any values below the threshold are colored red in the table in the report.

Uptime and downtime are illustrated in hours and minutes, as well as the percentage of the monitoring interval during which a computer is running or not. For example, if during a 24-hour monitoring interval, the computer is running for 18 hours and not running for six hours, the uptime and downtime are represented as:

- Uptime: 18 hours 0 minutes
- Downtime: 6 hours 0 minutes
- Uptime: 75%
- Downtime: 25%

This report uses data collected by the NT_SystemUpTime and UNIX_SystemUpTime Knowledge Scripts. In order to have accurate data for this report, schedule these scripts to run every 5 minutes.

Uptime and downtime are calculated during scheduled maintenance. Ad hoc maintenance is considered as downtime, and is included in all calculations.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Select computer(s)	Filter the data for your report by computer.
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Select peak weekday(s)	Filter the data for your report by the days of the week.
Uptime threshold %	Specify a value for the minimum uptime threshold. Any value below this threshold is colored red in the table. Use any value less than 100. You can use decimals.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Chart width	Provide a value for the width in pixels of the pie chart image.
Chart height	Provide a value for the height in pixels of the pie chart image.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).

Parameter	How To Set It
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

14.44 WatchList

Use this Knowledge Script to generate a report detailing the top or bottom N computers, by number or percent, that generated the selected datastreams.

Resource Object

Report agent

Default Schedule

The default schedule is **Run once**.

Setting Parameter Values

Set the following parameters as needed:

Parameter	How To Set It
Data source	
Top or bottom	Select either **Top** or **Bottom** as a filtering criterion.
Number N	Set the value of the *Top or bottom* parameter (for example, top 5 or bottom 5).
Number or percent	Select whether the report defines the top or bottom N by number or percent (for example, top 5 or bottom 5 percent).
Top or bottom by	Select which type of datastream values take precedence in the report: ◆ Average ◆ Minimum ◆ Maximum For example, if you are reporting on the Top 5 computers and you set this parameter to Average, then the top 5 computers with the highest average values for the selected datastreams are included in the report. In this case, the maximum and minimum values are also included, but the average values are listed first in the table, and the average values are by default given a unique graphic style in the chart. The report always includes all three values for the sake of comparison, but the type of value you select for this parameter is given precedence in the report.
Select data wizard	Select the data for your report by Knowledge Script or by datastream.

Parameter	How To Set It
Select the style	Select the style for the first page of the report: ◆ **By datastream** provides links to pages showing a side-by-side comparison of values for the same datastream collected from different computers (each page shows, for example, the value of the *NT_CpuResource-All Threads(#)* datastream from each computer) ◆ **By Knowledge Script** provides links to pages showing all datastreams collected by a Knowledge Script (each page shows all datastreams collected from all computers on which the script has run)
Select time range	Filter the data for your report by a specific or sliding time range. The default is Sliding.
Select peak weekday(s)	Filter the data for your report by the days of the week.
Report settings	
Include parameter help card?	Set to **y** to include a table in the report that lists parameter settings for the report script. The default is y.
Include table/chart/both?	Select the type of datastream values you want to include in the report: ◆ **Table** (table only) ◆ **Chart** (chart only) ◆ **Both** (table and chart)
Select chart style	Define the graphic properties of the charts in your report.
Top/bottom N series style	Select the graphical style of the data series representing the top or bottom N datastreams.
Select output folder	Set parameters for the output folder.
Add job ID to output folder name?	Set to **y** to append the job ID to the name of the output folder. A job ID helps you correlate a specific instance of a Report Script with the corresponding report. The default is n.
Select properties	Set miscellaneous report properties as desired.
Add time stamp to title?	Set to **y** to append a time stamp to the title of the report, making each title unique. The time stamp contains the date and time the report was generated. A time stamp allows you to run consecutive iterations of the same report without overwriting previous output. The default is n.
Event notification	
Event for report success?	Set to **y** to raise an event when the report is successfully generated. The default is y.
Severity level for report success	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 35 (magenta level indicator).
Severity level for report with no data	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 25 (blue level indicator).

Parameter	How To Set It
Severity level for report failure.	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5 (red level indicator).

A Using PowerShell Callback and Helper Functions

The PowerShell_RunCommand and Action_RunPowerShell Knowledge Scripts make a number of callback and helper functions available to the PowerShell commands or scripts being run.

These functions are implemented in a file called `Agent.ps1`, which is installed in the `NetIQ\AppManager\bin\PowerShell\Scripts\Library` folder on agent computers. The `PowerShell_RunCommand.ps1` script automatically loads the `Agent.ps1` file.

This appendix explains how to use four key PowerShell functions:

- Agent_CreateEvent, which lets you create an AppManager event.
- Agent_CreateData, which lets you create an AppManager data point.
- Agent_BuildDetailTableXML, which lets you build an XML string defining a detail or event message formatted as a table.
- Agent_BuildDynamicLegendXML, which lets you build an XML string defining a Analysis Center-compatible datastream legend.

For details on the additional functions not listed here, review the `Agent.ps1` file.

A.1 Agent_CreateEvent

This function raises an AppManager event. The arguments to this function are listed below in the order they should be passed to the function.

Required arguments:

- **$severity** defines the severity of the event, from 1 to 40. The lower the severity value, the more severe the event.
- **$event_msg** defines the text of the short event message. The short event message should be identical for all instances of the same event. Include details that might differ between instances in the detail message, such as the value of a measurement that exceeded a threshold.
- **$detail_msg** defines the text of the event detail message. This can be either plain text or an XML string that describes how the message should be formatted into a table (see Agent_BuildDetailTableXML). To create an event without a detail message, pass `$null` or an empty string ("").
- **$resource** defines the type and name of the resource object with which this event is associated. For scripts or commands executed using the PowerShell_RunCommand script, specify this value as `"NT_PowerShellFolder = $NT_PowerShellFolder"`.

Optional arguments:

- **$action** defines the IDs of the actions you want to execute when the event is raised. To trigger the actions that are associated with the PowerShell_RunCommand job, specify `$AKPID` for this argument. To disable action triggering, omit this argument or pass `$null` as the argument value.

Returns:

- This function returns no value.

Example usage:

- To create an informational event with an event detail message without triggering actions:

  ```
  Agent_CreateEvent 40 "This is an informational event" "These are the event
  details" "NT_PowerShellFolder = $NT_PowerShellFolder"
  ```

- To create an error event with no detail message, which triggers the actions associated with the job:

  ```
  Agent_CreateEvent 1 "This is a severe error event" $null "NT_PowerShellFolder =
  $NT_PowerShellFolder" $AKPID
  ```

A.2 Agent_CreateData

This function creates an AppManager data point. If this is the first data point for this datastream, AppManager automatically creates the datastream header. The arguments to this function are listed below in the order they should be passed to the function.

Required arguments:

- **$stream_name** identifies the datastream. This name must be unique among all of the datastreams generated by the job. The stream name is not exposed to users, so it does not need to be easily readable. The stream name is used solely to tie all data points representing the same measurement (one during each iteration of the job) together into a single stream.
- **$am_legend** describes what this datastream measures. AppManager uses this legend, but not Analysis Center. The legend should be a single line of text that describes the resource with which the datastream is associated, as well as the measurement the datastream represents, such as *Number of restarts for service X*. For data points that have associated units, add "`^^<units>`" to the end of the string, such as "`CPU usage^^%`".
- **$ac_legend** defines the Analysis Center (AC) dynamic legend XML, or `$null` if no AC legend is included. Analysis Center uses this legend, but AppManager does not. If Analysis Center will not be used to report on this datastream, pass `$null` for this argument. Otherwise pass a legend produced by the Agent_BuildDynamicLegendXML function.
- **$resource** defines the type and name of the resource object with which this event is associated. For scripts or commands executed using the PowerShell_RunCommand job, specify this value as "`NT_PowerShellFolder = $NT_PowerShellFolder`".
- **$value** defines the numeric (integer or floating point) value of the data point to be created.
- **$detail_msg** defines the text of the data detail message. This can be either plain text or an XML string that describes how the message should be formatted into a table (see Agent_BuildDetailTableXML). Pass `$null` if no detail message is to be associated with the data point.

Returns:

- This function returns no value.

Example usage:

- To create a data point for CPU usage, with no Analysis Center legend and no detail message:

  ```
  Agent_CreateData "CPU usage" "CPU usage^^%" $null "NT_PowerShellFolder =
  $NT_PowerShellFolder" 93.5 $null
  ```

A.3 Agent_BuildDetailTableXML

This function builds an XML string describing a table, suitable for passing as the detail message to the Agent_CreateEvent and Agent_CreateData functions.

Required arguments:

- **$table_desc** defines a hashtable that described the structure and content of the table, as shown below:

  ```
   @{
       Title              = <table title>
       Description        = <table description>
       TableGUID          = <table globally unique ID (GUID)>
       IsTransposed       = <$true if column-based, else $false>
       DetailType         = <either "Event" or "Data">
       ColumnDefs         = <array of hashtables defining columns>
       Rows               = <array or hashtables holding table data>
  }
  ```

  Each entry in the `ColumnDefs` array should have the following format:

  ```
   @{
   Name                = <name identifying the column>
   Title               = <the string used as the column title>
   ACType              = <"ACName" or "ACValue">
  }
  ```

  NOTE: The `ACType` is no longer used, but you still need to specify it. The need to specify this value will be removed in a future release of the product.

  Each entry in the `Rows` array should have the following format:

  ```
   @{
   <column name>       =
   @{
       Text            = <text to be displayed in this table cell>
       Style           = <optional: CSS style to apply to the cell>
    }
  }
  ```

Returns:

- This function returns an XML string suitable for passing to Agent_CreateEvent or Agent_CreateData as the detail message, resulting in the details being rendered as a table rather than as free-flowing text.

Example usage:

- To define rows (first method):

```
$rows = New-Object Collections.Hashtable[] 2


 $rows[0] = New-Object Collections.Hashtable

 $rows[0]["Column1"] = New-Object Collections.Hashtable

 $rows[0]["Column1"]["Text"] = "Row 1 Column 1 Text"

 $rows[0]["Column1"]["Style"] = "font-weight: bold; color: black;"

 $rows[0]["Column2"] = New-Object Collections.Hashtable

 $rows[0]["Column2"]["Text"] = "Row 1 Column 2 Text"

 $rows[0]["Column2"]["Style"] = "font-weight: normal; color: black;"


 $rows[1] = New-Object Collections.Hashtable

 $rows[1]["Column1"] = New-Object Collections.Hashtable

 $rows[1]["Column1"]["Text"] = "Row 2 Column 1 Text"

 $rows[1]["Column1"]["Style"] = "font-weight: bold; color: red;"

 $rows[1]["Column2"] = New-Object Collections.Hashtable

 $rows[1]["Column2"]["Text"] = "Row 2 Column 2 Text"

 $rows[1]["Column2"]["Style"] = "font-weight: normal; color: red;"
```

- To define rows (second method):

```
$rows =
@(
    @{
        Column1 =
        @{
            Text  = "Row 1 Column 1 Text"
            Style = "font-weight: bold; color: black;"
        }
        Column2 =
        @{
            Text  = "Row 1 Column 2 Text"
            Style = "font-weight: normal; color: black;"
        }
    }
    @{
        Column1 =
        @{
            Text  = "Row 2 Column 1 Text"
            Style = "font-weight: bold; color: red;"
        }
        Column2 =
        @{
            Text  = "Row 2 Column 2 Text"
            Style = "font-weight: normal; color: red;"
        }
    }
)
```

- To define columns (first method):

```
$column_def_1 = New-Object Collections.Hashtable
$column_def_1["Name"]   = "Column1"
$column_def_1["Title"]  = "Column 1 Title"
$column_def_1["ACType"] = "ACName"

$column_def_2 = New-Object Collections.Hashtable
$column_def_2["Name"]   = "Column2"
$column_def_2["Title"]  = "Column 2 Title"
$column_def_2["ACType"] = "ACName"

$column_defs = @($column_def_1, $column_def_2)
```

- To define columns (second method):

```
$column_defs =
@(
    @{
        Name    = "Column1"
        Title   = "Column 1 Title"
        ACType  = "ACName"
    }
    @{
        Name    = "Column2"
        Title   = "Column 2 Title"
        ACType  = "ACValue"
    }
)
```

- To define the table:

```
    $table_info =
@{
        Title               = "My Table"
        Description         = "My Table Description"
        TableGUID           = "3cd66ae9-741a-447e-a8fc-7c67cedb066a"
        IsTransposed        = $false
        DetailType          = "Event"
        ColumnDefs          = $column_defs
        Rows                = $rows
    }
```

- To build the XML string to be passed to the Agent_CreateEvent function:

  ```
  $table_xml = Agent_BuildDetailTableXML $table_info
  ```

A.4 Agent_BuildDynamicLegendXML

This function builds a dynamic legend (or AC legend) string suitable for passing to the Agent_CreateData function as the AC legend argument. Dynamic legends built with this function are compatible with Analysis Center, and all datastreams to be consumed by Analysis Center should build their dynamic legends using this function.

The arguments to this function are listed below in the order they should be passed to the function.

Required arguments:

- **$measurement_desc** defines the measurement, such as "Current Session Count" or "Time Since Last Restart."
- **$measurement_units** defines the units associated with the measurement, such as "Sessions" for the "Current Session Count" measurement or "Seconds" for the "Time Since Last Restart" measurement. In the case of "Current Session Count," you can also specify the units as the empty string, because the units are evident from the name of the measurement.
- **$measured_resource** defines a string that indicates the specific resource to which the datastream is related, such as "Application Server Two" for the "Current Session Count" measurement or "NetIQ Agent Service" for the "Time Since Last Restart" measurement.

Returns:

- This function returns an XML string suitable for passing to the Agent_CreateData function as the AC-compatible legend argument, for use by the NetIQ Analysis Center reporting application.

Example Usage:

- To build an AC legend:

  ```
  $ac_legend = Agent_BuildDynamicLegendXML "Current Session Count" "Sessions"
  "Application Server Two"
  ```