NetIQ® AppManager® for Citrix XenApp

Management Guide

March 2014

Contents

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.

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.

1 Introducing AppManager for Citrix XenApp

AppManager lets you manage the performance and availability of servers running Citrix XenApp.

The XenApp category of Knowledge Scripts tracks vital aspects of system performance, including:

- The number of sessions on a server
- The number of sessions opened by each user
- The number of bytes transferred from each session by each user
- The average and high latency of server sessions
- Session states
- The number of processes on a server across all sessions
- The memory and CPU resources used by server processes
- The number of users running an application across all sessions
- The percentage of all licenses that are in use on the License Server, on clustered and non-clustered environments
- Whether a specific application is on the list of published applications, and which servers are assigned to that published application
- Whether a server is the default data collector, also called the master browser in previous versions of the Citrix Server, along with zone information, for a specific Citrix Server

By monitoring your Citrix XenApp implementation at the component level, you can quickly identify areas that are or will have a negative impact on overall performance. Then you can modify how you have configured the system.

For example, if you notice an excessive or steadily increasing number of server sessions, you can add additional processors and RAM to accommodate the load. Additionally, if you are maintaining a load-balanced environment, you can investigate whether another server is suffering problems.

After you have the system tuned to your satisfaction, you can collect new data to use as a benchmark for setting performance thresholds. When you establish these thresholds in the Knowledge Scripts that monitor your system, you are setting the boundaries for optimum performance. Any time the system begins operating outside those boundaries, you can be alerted immediately, and take steps to correct or prevent problems.

In addition to providing an ongoing evaluation of system performance, the Knowledge Scripts in the XenApp category collect data you can use for reporting. Reports can be configured for any time frame during which you have collected data, and allow you to study data in increments from one minute to one month. For example, you can study the minute-by-minute use of CPU time by Citrix XenApp server processes for the last hour, or the average number of sessions per user per month over the past year. The flexibility of the AppManager reporting architecture lets you perform close scrutiny of your system, as well as illustrate historical trends.

2 Installing AppManager for Citrix XenApp

This chapter provides installation instructions and describes system requirements for AppManager for Citrix XenApp.

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.

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 Citrix XenApp has the following system requirements:

Software/Hardware	Version
NetIQ AppManager installed on the AppManager repository (QDB) computers, on the Citrix computers you want to monitor (agents), and on all console computers	7.0 or later Support for Windows Server 2008 on AppManager 7.x requires AppManager Windows Agent hotfix 71704 or later. For more information, see the AppManager Suite Hotfixes page.
Microsoft Windows operating system on the agent computers	One of the following: ◆ Windows Server 2008 R2 ◆ Windows Server 2008 (32- and 64-bit)
AppManager for Microsoft Windows module installed on repository, agent, and console computers	7.6.276.0 or later.
Citrix XenApp or Citrix License Server, or both, on the agent computers	One of the following: ◆ Citrix XenApp Server or License Server 6.5 ◆ Citrix XenApp Server or License Server 6.0 ◆ Citrix XenApp Server (Platinum or Enterprise edition) or License Server 5.0
Microsoft Windows PowerShell on all Citrix XenApp Server computers	2.0 or later
Citrix XenApp Server SDK on all Citrix XenApp Server computers	6.0 or later

If you encounter problems using this module with a later version of your application, contact NetIQ Technical Support.

2.2 Installing the Module

Run the module installer on the XenApp servers 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-XenApp-8.x.x.0.msi` module installer from the `AM70-XenApp-8.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 `.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 `.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 now installs Knowledge Scripts for each module directly into the QDB instead of to 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 13. 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.

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 installs 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, run the module installer only for the primary QDB. Control Center automatically replicates 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 XenApp servers you want to monitor (agents) to install the agent components.

9 (Conditional) If you have not discovered XenApp resources, run the Discovery_XenApp Knowledge Script on all agent computers where you installed the module. For more information, see Section 2.5, "Discovering Citrix XenApp Resources," on page 14.

10 To get the updates provided in this release, upgrade any running Knowledge Script jobs. For more information, see Section 2.6, "Upgrading Knowledge Script Jobs," on page 15.

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

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.

2.3.1   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 Section 2.3.2, "Checking In the Installation Package," on page 13.

3 Configure an e-mail 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.

2.3.2   Checking In the Installation Package

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

To check in a module installation package:

1 Log in 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 **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 **Check in Packages** (for AppManager 7.x).

5 Navigate to the folder where you saved `AM70-XenApp-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.

2.4 Silently Installing the Module

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

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

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

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

```
/L* "AM70-XenApp-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 to a remote AppManager repository, you can use Windows authentication or SQL authentication.

Windows authentication:

```
AM70-XenApp-7.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-XenApp-7.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 Discovering Citrix XenApp Resources

Use this Knowledge Script to discover Citrix XenApp resources and configuration information. The TreeView for this module now includes a reorganized set of objects that include Citrix XenApp, License Servers, Licenses, Farms, and Servers, along with several additional services: MFCom, Citrix Licensing, Citrix XTE Server, Citrix Service Manager, Citrix Encryption Service, and Citrix IMA Service.

AppManager for Citrix XenApp supports cluster discovery on all cluster nodes for the Citrix License Server component. If you run the Discovery Knowledge Script on both nodes of a cluster added to the Operator Console, the Discovery script discovers the license server on both nodes, but the license types available on the license server object are not discovered. The TreeView for cluster discovery displays the license types available on the License Server object for non-cluster servers only.

Set the parameters on the Values tab as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the discovery job fails. The default is 5.
Discovery details	
Discover applications (yes, no)?	Select **Yes** to discover applications on XenApp Application Servers in addition to servers. The default is yes.
Event Notification	
Raise event when discovery succeeds?	Select **Yes** to raise an event if the discovery process is successful. The default is unselected.
Event severity when discovery succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the discovery process is successful. The default is 21.
Event severity level when discovery partially succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a discovery returns some data but also generates warning messages. The default is 11.

2.6 Upgrading Knowledge Script Jobs

This release of AppManager for Citrix XenApp contains updated Knowledge Scripts. 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.

2.6.1 Running AMAdmin_UpgradeJobs

If you are using AppManager 8.x or later, the module upgrade process now *retains* any changes you might have made to the parameter settings for the Knowledge Scripts in the previous version of this module. Before AppManager 8.x, the module upgrade process *overwrote* any settings you might have made, changing the settings back to the module defaults.

As a result, if this module includes any changes to the default values for any Knowledge Script parameter, the module upgrade process ignores those changes and retains all parameter values that you updated. 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.

2.6.2 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. Customized script parameters may have reverted to default parameters during the installation of the module. 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

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.

To propagate changes to ad hoc Knowledge Script jobs:

1   In the Knowledge Script view, select the Knowledge Script for which you want to propagate changes.

2   Click **Properties Propagation** > **Ad Hoc Jobs**.

3   Select the components of the Knowledge Script that you want to propagate to associated ad hoc jobs:

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.

Propagating Changes to Knowledge Script Groups

You can propagate the properties and logic (script) 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. For more information, see "Propagating Changes to Ad Hoc Jobs" on page 16.

To propagate Knowledge Script changes to Knowledge Script Groups:

1   In the Knowledge Script view, select the Knowledge Script Group for which you want to propagate changes.

2   On the KS menu, select **Properties propagation** > **Ad Hoc Jobs**.

3   *If you want to exclude a Knowledge Script member from properties propagation*, deselect that member from the list in the Properties Propagation dialog box.

4 Select the components of the Knowledge Script that you want to propagate to associated Knowledge Script Groups:

Select	To propagate
Script	The logic of the Knowledge Script.
Properties	Values from the Knowledge Script Schedule and Values tabs, including the schedule, actions, and Advanced properties.

5 Click **OK**. Any monitoring jobs started by a Knowledge Script Group member are restarted with the job properties of the Knowledge Script Group member.

2.7 Upgrading Knowledge Script Jobs

If you are using AppManager 8.x or later, the module upgrade process now *retains* any changes you might have made to the parameter settings for the Knowledge Scripts in the previous version of this module. Before AppManager 8.x, the module upgrade process *overwrote* any settings you might have made, changing the settings back to the module defaults.

As a result, if this module includes any changes to the default values for any Knowledge Script parameter, the module upgrade process ignores those changes and retains all parameter values that you updated. 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.

2.7.1 Running AMAdmin_UpgradeJobs

The AMAdmin_UpgradeJobs Knowledge Script can push changes to running Knowledge Script jobs. Your AppManager repository (QDB) must be at version 7.0 or later. 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.

2.7.2 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. You might need to appropriately set new parameters for your environment or application.

If you are not using AppManager 8.x or later, customized script parameters might have reverted to default parameters during the installation of the module.

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 the properties and the script.

For more information about propagating Knowledge Script changes, see the "Running Monitoring Jobs" chapter of the *Control Center User Guide for AppManager*.

2.7.3 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. If you are using AppManager 8.x or later, the module upgrade process now *retains* any changes you might have made to the parameter settings for the Knowledge Scripts in the previous version of this module.

2.8 Permissions for Running XenApp Knowledge Scripts

To run the XenApp Knowledge Scripts, the AppManager agent needs certain secure permissions. The account under which the agent services, NetIQmc and NetIQccm, run must be able to log in as a service on each monitored Citrix XenApp computer. The agent must run in one of the following ways:

◆ Run as a Windows user that is configured as a XenApp Administrator.

◆ Run as the local system account and configure AppManager Security Manager so that PowerShellHost runs as a WIndows user configured as a XenApp Administrator. For information about how to configure the PowerShellHost credentials, see Section 2.8.1, "Configuring Security Manager," on page 19.

If you run the NetIQmc agent service as a Windows user that is configured as a XenApp administrator, ensure your user account has *Full Administration* or *View Only* privileges, not *Custom* privileges.

2.8.1 Configuring Security Manager

To run the PowerShell cmdlets, the MCPSHostServer must run under the XenApp Administrator user account. If you are using the local system account, you must configure Security Manager for the correct credentials.

To configure AppManager Security Manager to ensure that MCPSHostServer runs under the XenApp Administrator user account:

1 On the **Extensions** menu in the Operator Console, click **Security Manager**.

2 Select the XenApp server.

3 On the Custom tab, click **Add**.

4 In the Label field, type **XenApp**.

5 In the Sub-label field, type **XAAdminUser**

6 In the Value 1 field, specify the user name for the XenApp Administrator.

7 In the Value 2 field, specify the password for the account.

8 In the Value 2 field, specify the domain for the account.

9 Select **Extended application support** to encrypt the password when it is stored in the repository.

10 Click **OK**.

11 Click **Apply** to save the Security Manager settings.

2.9 Configuring the PowerShell Execution Policy

This section describes the procedure for configuring the Microsoft PowerShell Execution Policy for environments that use Powershell. The PowerShell Execution Policy determines whether PowerShell scripts are allowed to run.

2.9.1 Understanding PowerShell Cmdlets

Citrix XenApp uses the Microsoft scripting and command environment known as PowerShell. PowerShell is made up of hundreds of executable objects called **cmdlets**, pronounced **command-lets**. When running the XenApp category of Knowledge Scripts, AppManager makes a series of calls to PowerShell and the XenApp cmdlets. The combination of cmdlets depends on the version of XenApp Server. AppManager executes the cmdlets to manipulate XenApp objects.

For more information about using Powershell, see your Microsoft PowerShell documentation.

2.9.2 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.

The Execution Policy directly affects the XenApp Knowledge Scripts. Although the scripts that ship with AppManager for XenApp 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 XenApp Knowledge Scripts, but with a `.ps1` extension.

NOTE: The digital signature encoded in a XenApp 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 XenApp 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 make with the `Set-ExecutionPolicy` cmdlet. For example, if the group policy forbids script execution, you cannot change the policy by running `Set-ExecutionPolicy`. 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 scripts, you must change the Execution Policy from `Restricted` to one of the following policy options:

- ◆ **AllSigned**, which 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 2.9.3, "Trusting XenApp PowerShell Scripts," on page 20.
- ◆ **RemoteSigned**, which allows local scripts to run regardless of signature, and requires trusted digital signatures only for remote scripts. XenApp Knowledge Scripts are local scripts.
- ◆ **Unrestricted**, which allows both local and remote scripts to run, regardless of signature.

To change the PowerShell Execution Policy:

1 Open the XenApp Command Shell on the agent computer.

2 Run the following cmdlet:

`Set-ExecutionPolicy <policy>`

where `<policy>` is the name of the Execution Policy you choose.

3 Repeat Steps **1** and **2** on all agent computers, including Server role computers.

2.9.3 Trusting XenApp PowerShell 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 XenApp 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* XenApp server in your environment.

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

To import the digital certificate:

1 Open the XenApp SDK Command Shell on the agent computer.

2 Set the ExecutionPolicy to AllSigned or RemoteSigned.

3 Change to the `AppManager\bin\PowerShell\Scripts` directory.

4 Type `.\XenApp_ServerFarmHealth.ps1` to establish trust.

5 Click Enter.

6 Type `A` at the prompt asking whether the script should be allowed to run.

7 Click Enter.

These steps allow the NetIQ Corporation digital certificate to be imported into the certificate store for the user running the script. Although ServerFarmHealth is provided in the instructions, you can run any script to establish trust.

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 XenApp PowerShell scripts.

To extend trust to all other user accounts, see Section 2.9.4, "Extending Trust to All User Accounts," on page 21.

2.9.4 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 in the Microsoft Management Console window, 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**. The **Certificates-Current User** node is displayed in the tree view of the Console window.

7 Expand **Certificates - Current User**.

8 Expand **Trusted Publishers** and select **Certificates**.

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 all users to execute the PowerShell scripts.

2.10 Changing Configuration Settings

AppManager for XenApp 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 XenApp scripts are executed.

Both components have associated configuration files that define certain operational parameters. You can modify 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, you should not change the value of that setting.

2.10.1 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.

2.10.2 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.

- **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. Because XenApp Knowledge Script jobs use XenApp cmdlets, which require a large amount of memory, server memory usage can grow excessively.

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 PowerShell Errors

XenApp Knowledge Scripts might 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 some 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 can use, modify the value of the maxMemoryUsage setting. For more information, see Section 2.10, "Changing Configuration Settings," on page 22.

◆ **Increase the number of PowerShell execution environments, or runspaces that MCPSHostServer.exe can host**. The default number of runspaces is eight, which means no more than eight Knowledge Script jobs can be running simultaneously on 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 in this manner 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 Configuration Settings," on page 22.

Increasing this value will be beneficial if you are running more than eight XenApp Knowledge Script jobs, but even then the benefit may not be significant.

NOTE: The client's maxActiveServers configuration option specifies the maximum number of servers that can be active at any time (the default is five). The maxActiveServers configuration value and the UpperMaxRunspaceHosts server configuration value determine the total number of jobs that can be serviced at any one time. You can have more than this number of jobs in the "Running" state in AppManager, but only if some of the jobs are between iterations, and not actually running at the same time.

---

3 XenApp Knowledge Scripts

AppManager provides the following Knowledge Scripts for monitoring servers that are running Citrix MetaFrame.

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
ApplicationUsersHigh	Monitors the number of users running one or more applications across all sessions on a specific XenApp server.
ApplicationUsersHighAll	Monitors the number of users running one or more applications across all sessions in a server farm.
BytesTransferredPerUser	Monitors the number of bytes per user transferred between client computers and a XenApp server.
DataCollectorChanged	Monitors whether a zone's data collector has changed since the last monitoring interval.
DefaultDataCollector	Identifies the default data collector for a XenApp server.
FarmUserLoad	Monitors the number of users connected to each XenApp server in a server farm.
ICAAvgLatencyHigh	Monitors the average latency of ICA sessions on a XenApp server.
ICALatencyHigh	Monitors the most-recent measure of latency for ICA sessions on a XenApp server.
LicenseInUseHigh	Monitors the percentage of licenses in use.
PublishedApplicationDetails	Searches for specified applications that are on the list of published applications for XenApp server farms.
ServerFarmHealth	Monitors the health and availability of XenApp Server services in a designated server farm and monitors the farm for servers that are not responding.
ServerProcessesHigh	Monitors the number of processes on a XenApp server across all sessions.
ServerProcessesResourceHigh	Monitors the use of CPU and memory resources by processes on a XenApp server.
ServerSessionHigh	Monitors the number of sessions on a XenApp server.
SessionPerUser	Monitors the number of sessions on a XenApp server that are open for each user.
SessionState	Monitors the number of sessions matching specified states.

Knowledge Script	What It Does
UserResourcesHigh	Monitors the use of CPU and memory resources by users connected to a XenApp server.

3.1 ApplicationUsersHigh

Use this Knowledge Script to monitor the number of users across all sessions running applications published on a XenApp server. If the number of users falls below the minimum threshold or exceeds the maximum threshold, an event is raised.

NOTE: To gather data about all sessions on servers in a XenApp farm, run the ApplicationUsersHighAll Knowledge Script instead of this Knowledge Script.

If you are monitoring multiple applications, separate events are raised for each application. The same thresholds apply to all applications.

3.1.1 Resource Objects

Citrix XenApp Applications object or individual applications

3.1.2 Default Schedule

The default schedule is **Every 30 minutes**.

3.1.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the ApplicationUsersHigh job fails. The default is 5.
Event Notification	
Raise event if number of users exceeds or falls below threshold?	Select **Yes** to raise an event when the number of users running an application falls below the minimum threshold or exceeds the maximum threshold you set. The default is Yes.
Event severity when number of users exceeds or falls below threshold	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5.
Data Collection	
Collect data for number of users?	Select **Yes** to collect data for charts and reports. If enabled, returns information about the number of users running an application. The default is not selected.
Monitoring	

Description	How to Set It
Threshold -- Minimum number of users	Specify the minimum number of users across all sessions that can be running a published application before an event is raised. The value can range from 0 to 99998 users, and must be lower than the threshold for the maximum number of users. The default is 5.
Threshold -- Maximum number of users	Specify the maximum number of users across all sessions that can be running a published application before an event is raised. The value can range from 1 to 99999 users, and must be higher than the threshold for the minimum number of users. The default is 50.

3.2 ApplicationUsersHighAll

Use this Knowledge Script to monitor the number of users across all sessions running applications published in a XenApp farm. If the number of users falls below the minimum threshold or exceeds the maximum threshold, an event is raised.

NOTE: To monitor users on an individual server instead of a XenApp farm, use the ApplicationUsersHigh Knowledge Script instead of this Knowledge Script.

If you are monitoring multiple applications, separate events are raised for each application. The same thresholds apply to all applications.

3.2.1 Resource Objects

Citrix XenApp Applications object or individual applications

3.2.2 Default Schedule

The default schedule is **Every 30 minutes**.

3.2.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event where the ApplicationUsersHighAll job fails. The default is 5.
Event Notification	
Raise event if number of users exceeds or falls below threshold?	Select **Yes** to raise an event when the number of users running an application falls below the minimum threshold or exceeds the maximum threshold you set. The default is Yes.

Description	How to Set It
Event severity when number of users exceeds or falls below threshold	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5.
Data Collection	
Collect data for number of users?	Select **Yes** to collect data for charts and reports. If enabled, returns information about the number of users running an application. The default is not selected.
Monitoring	
Threshold -- Minimum number of users	Specify the minimum number of users across all sessions that can be running a published application before an event is raised. The value can range from 0 to 99998 users, and must be lower than the threshold for the maximum number of users. The default is 5.
Threshold -- Maximum number of users	Specify the maximum number of users across all sessions that can be running a published application before an event is raised. The value can range from 1 to 99999 users, and must be higher than the threshold for the minimum number of users. The default is 50.

3.3 BytesTransferredPerUser

Use this Knowledge Script to monitor the number of bytes per user transferred between client computers and the XenApp server.

The number of bytes is calculated by taking the total of all bytes for all Independent Computing Architecture (ICA) sessions currently active for a user. For each user with one or more ICA protocol sessions on XenApp, the sum of bytes transferred by all sessions associated with that user is compared to the threshold you set. If the number of bytes exceeds the threshold, an event is raised.

3.3.1 Resource Objects

Citrix XenApp object

3.3.2 Default Schedule

The default schedule is **Every 5 minutes**.

3.3.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the BytesTransferredPerUser job fails. The default is 5.

Description	How to Set It
Event Notification	
Raise event if the total number of bytes transferred for a user exceeds threshold?	Select **Yes** to raise an event if the total bytes per user exceeds the threshold. The default is Yes.
Event severity when the total number of bytes transferred for a user exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the total bytes per user exceeds the threshold you set. The default is 5.
Data Collection	
Collect data for bytes transferred per user?	Select **Yes** to collect data for charts and reports. If enabled, returns information about the number of bytes per user transferred between ICA clients and XenApp. The default is unselected.
Monitoring	
Threshold -- Maximum bytes transferred per user	Specify the maximum number of bytes that can be transferred per user before an event is raised. The default is 10485760 bytes.

3.4   DataCollectorChanged

Use this Knowledge Script to determine whether the data collector for a XenApp server zone has changed since the last time the script was run. If a change to the data collector for the selected zone is detected, an event is raised.

3.4.1   Resource Objects

Citrix XenApp Zones object or individual zones

3.4.2   Default Schedule

The default schedule is **Every 30 minutes**.

3.4.3   Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the DataCollectorChanged job fails. The default is 5.
Event Notification	
Raise event if a change to the data collector is detected?	Select **Yes** to raise an event if a change to the data collector for this server zone has occurred since the last monitoring interval. The default is Yes.

Description	How to Set It
Event severity when a change is detected	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a change to the data collector occurs. The default is 5.
Data Collection	
Collect data for data collector changes?	Select **Yes** to collect data for charts and reports.
	If enabled, data collection returns one of the following values:
	◆ **100** if the data collector has changed
	◆ **0** if the data collector has not changed
	The default is unselected.

3.5   DefaultDataCollector

Use this Knowledge Script to identify the default data collector for a specific XenApp server under a XenApp farm, or to identify *all* available XenApp servers under a XenApp farm.

This script raises an event if the default data collector information is found, and the event message includes default data collector and zone information for the selected XenApp server.

If you run this script on the server object, the event returns the zone name and the default data collector for all the servers that are discovered under server object. If you run this script on a particular server or set of servers, the event returns the zone name and default data collector for those servers only.

3.5.1   Resource Object

XenApp Servers object or individual servers

3.5.2   Default Schedule

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

3.5.3   Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the DefaultDataCollector job fails. The default is 5.
Event Notification	

Description	How to Set It
Event severity when default data collector information is found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the default data collector information is found. The default is 15.
Event severity when user is not a Citrix XenApp farm administrator	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the user is not a XenApp farm administrator. The default is 11.

3.6 FarmUserLoad

Use this Knowledge Script to monitor the number of users connected to each XenApp server in a server farm. You can set thresholds for the minimum and maximum number of users. An event is raised if the maximum threshold is exceeded or the minimum threshold is not met.

In addition, you can set thresholds based on a standard deviation, calculated from the number of users connected to each server in the farm since the first job iteration. The maximum and minimum thresholds for individual servers are defined by the number of standard deviations above or below the average number of users connected to all servers since the first iteration of the job.

If you use the standard deviation thresholds, the thresholds for the minimum and maximum numbers of users are ignored.

You can also specify servers in a farm that are to be excluded from monitoring by this Knowledge Script.

3.6.1 Resource Object

XenApp Farm object

3.6.2 Default Schedule

The default schedule is **Every 30 minutes**.

3.6.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the FarmUserLoad job fails. The default is 5.
Event Notification	
Raise event if any threshold exceeded or not met?	Select **Yes** to raise an event if the number of standard deviations or the number of users exceeds or falls below one of the thresholds you set. The default is Yes.

Description	How to Set It
Event severity when any threshold exceeded or not met	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of standard deviations or the number of users exceeds or falls below a threshold. The default is 5.
Data Collection	
Collect data for number of users?	Select **Yes** to collect data for charts and reports. If enabled, returns the numbers of users connected to XenApp. The default is unselected.
Monitoring	
Type of threshold to use?	Select the type of threshold to use: ◆ Standard Deviation ◆ Minimum/Maximum The default is `Minimum/Maximum`.
Standard Deviation Settings	
Threshold -- Number of standard deviations below average	Specify the number of standard deviations below the average number of users connected to all servers in the farm. If the number of users of a particular server falls below this threshold, an event is raised. The default is 1.
Threshold -- Number of standard deviations above average	Specify the number of standard deviations above the average number of users connected to all servers in the farm. If the number of users of a particular server exceeds this threshold, an event is raised. The default is 1.
Minimum/Maximum Settings	
Threshold -- Minimum number of users	Specify the minimum number of users who must be connected to a server before an event is raised. The value must be lower than the threshold for the maximum number of users. The default is 10 users.
Threshold -- Maximum number of users	Specify the maximum number of users who can be connected to a server before an event is raised. The value must be higher than the threshold for the maximum number of users. The default is 50 users.
Servers to exclude (comma-separated, no spaces)	Provide a list of server names, separated by commas and no spaces (for example, `MFServer1,MFServer2,MFServer3`). Servers specified in this parameter are not monitored by this Knowledge Script.

3.7 ICAAvgLatencyHigh

Use this Knowledge Script to monitor the average latency, in milliseconds, for Independent Computing Architecture (ICA) sessions on a XenApp server. Latency refers to the delay between user input, such as mouse movement or keyboard strokes, and screen refresh.

Each time this Knowledge Script runs, it checks the average latency of each ICA session for the length of time the session has been open. If the average latency of any session exceeds the threshold you set, an event is raised.

Use the ICALatencyHigh Knowledge Script to monitor the most recently measured latency for each ICA session. If latency consistently exceeds the threshold you set, you can use the Citrix SpeedScreen Latency Reduction Manager to adjust your SpeedScreen settings.

3.7.1 Resource Objects

Citrix XenApp object

3.7.2 Default Schedule

The default schedule is **Every 30 minutes**.

3.7.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the ICAAvgLatencyHigh job fails. The default is 5.
Event Notification	
Raise event if average latency exceeds threshold?	Select **Yes** to raise an event if the average latency for ICA sessions exceeds the threshold. The default is Yes.
Event severity when average latency exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which average latency exceeds the threshold you set. The default is 5.
Data Collection	
Collect data for average latency?	Select **Yes** to collect data for charts and reports. If enabled, returns the average latency of each ICA session for the length of time the session has been open. The default is unselected.
Monitoring	
Threshold -- Maximum average latency of an ICA session	Specify a maximum threshold, in milliseconds, for the average latency for any ICA session. The default is 30 milliseconds.

3.8 ICALatencyHigh

Use this Knowledge Script to monitor the most recent or current measure of latency for each Independent Computing Architecture (ICA) session on a XenApp server. Latency refers to the delay between user input, such as mouse movement or keyboard strokes, and screen refresh.

If the most recent measure of latency for any ICA session exceeds the threshold you set, an event is raised.

Use the ICAAvgLatencyHigh Knowledge Script to monitor the average latency of all ICA sessions over time. If latency consistently exceeds the threshold you set, you can use the SpeedScreen Latency Reduction Manager to adjust your SpeedScreen settings.

3.8.1 Resource Objects

Citrix XenApp object

3.8.2 Default Schedule

The default schedule is **Every 30 minutes**.

3.8.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the ICALatencyHigh job fails. The default is 5.
Event Notification	
Raise event if current latency exceeds threshold?	Select **Yes** to raise an event if the current latency for any ICA session exceeds the threshold. The default is Yes.
Event severity when current latency exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which latency exceeds the threshold you set. The default is 5.
Data Collection	
Collect data for current latency of ICA sessions?	Select **Yes** to collect data for charts and reports. If enabled, returns the most recent measure of latency for each ICA session. The default is unselected.
Monitoring	
Threshold -- Maximum current latency of an ICA session	Specify the maximum latency amount (in milliseconds) any ICA session can have before an event is raised. The default is 30.

3.9 LicenseInUseHigh

Use this Knowledge Script to monitor the percentage of licenses in use for Citrix XenApp. If the percentage of licenses in use exceeds the threshold you set, an event is raised.

Citrix XenApp use a license server with license files that grant connection rights to a client. When a client connects to the server, one license is allocated. License servers can be shared by multiple server farms, and in such a case, a client can connect to either farm and consume only one license.

LicenseInUseHigh is cluster-aware. It monitors and collects data for active nodes, for all the available license types on the server. Even if you have two child jobs for LicenseInUseHigh, the script monitors and collects data for active nodes only. The LicenseInUseHigh job does not stop if the state of the

cluster node changes, such as when the passive node of the cluster becomes active, or the active node becomes passive. In the event of a failover, LicenseInUseHigh monitors all the license types available on the server.

If data collection is enabled, this Knowledge Script returns the percentage of licenses in use compared to the total number of licenses available on the license server.

3.9.1 Resource Object

For clustered environments, Citrix XenApp License object

For non-clustered environments. Citrix XenApp License object or individual license files

3.9.2 Default Schedule

The default schedule is **Every 30 minutes**.

3.9.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the LicenseInUseHigh job fails. The default is 5.
Event Notification	
Raise event if percentage of licenses in use exceeds threshold?	Select **Yes** to raise an event if the percentage of licenses in use exceeds the threshold. The default is Yes.
Event severity when percentage of licenses in use exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the percentage of licenses in use exceeds the threshold. The default is 5.
Data Collection	
Collect data for percentage of licenses in use?	Select **Yes** to collect data for charts and reports. If enabled, returns the percentage of licenses in use. The default is unselected.
Monitoring	
Threshold -- Maximum percentage of licenses in use	Specify the maximum percentage of licenses that can be in use before an event is raised. The default is 80%.

3.10 PublishedApplicationDetails

This Knowledge Script searches for specified applications that are on the list of published applications for Citrix Server farms. This script raises an event that lists details about the published application or the list of applications, including the name of the farms and servers on which the application has been published.

3.10.1 Resource Objects

Citrix XenApp Farm object

3.10.2 Default Schedule

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

3.10.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the PublishedApplicationDetails job fails. The default is 5.
Applications to be verified in the published application list (comma-separated)	Type the name of the application or applications you want to determine is in the published application list.
	For more than one application, separate the application names with a comma, no space. This parameter supports the wildcard characters "*" and "?" for published applications.
Event Notification	
Event severity when specified application details are found	Set the event severity level, from 1 to 40, to indicate the importance of the event raised when specific application details are found. The default is 15.
Event severity when user is not a Citrix farm administrator	Set the event severity level, from 1 to 40, to indicate the importance of the event raised when the user is not a XenApp farm administrator. The default is 11.

3.11 ServerFarmHealth

Use this Knowledge Script to monitor a XenApp server farm for unresponsive servers. You can set two thresholds for non-responding servers:

- The maximum number of servers that are unresponsive before a **warning** event is raised
- The maximum number of servers that are unresponsive before an **error** event is raised

This script raises an event if either threshold is exceeded. You can set severity levels for each event type.

You can also use this script to monitor the health and availability of the following services in a designated farm. The services in a designated farm must be running before you can collect data.

- Client Network
- Encryption
- Independent Management Architecture

- ◆ MFCOM (XenApp Management SDK)
- ◆ Licensing
- ◆ Services Manager
- ◆ XTE Server
- ◆ XML Server

Each service can display one of the following statuses:

- ◆ **Running** — The service is running.
- ◆ **Not running** — The service is not running.

3.11.1   Resource Objects

XenApp Farm object

3.11.2   Default Schedule

The default schedule is **Every 10 minutes**.

3.11.3   Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the ServerFarmHealth job fails. The default is 5.
Event Notification	
Raise event if number of servers not responding exceeds threshold?	Select **Yes** to raise an event if the number of unresponsive servers exceeds the thresholds you set. The default is Yes.
Raise event to display the status of XenApp Server services in a farm?	Select **Yes** to raise an event to display the status of XenApp server services in a designated farm. The default is Yes.
Warning event severity when the threshold is exceeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the warning threshold is exceeded. The default is 11.
Error event severity when the threshold is exceeded	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the error threshold is exceeded. The default is 5.
Event severity when the service is down	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the XenApp server service is down. The default is 5.
Data Collection	

Description	How to Set It
Collect data for XenApp servers not responding?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the percentage of servers in the server farm that are down. If any servers are down, the data details include the names of servers that are unresponsive. The default is unselected.
Collect data for XenApp services in a farm?	Select **Yes** to collect data for charts and reports. If enabled, data collection returns the percentage of XenApp server services in the farm that are down. The default is unselected.
Monitoring	
Servers to ignore	Provide a list of servers you do not want to monitor. Use commas with no spaces to separate server names in a list. For example, MFServer1,MFServer2,MFServer3.
	You can also click **Browse [...]** to use a network browser to select computer names.
Services to Ignore	
Ignore Client Network Service?	Select **Yes** to allow the script to ignore the Client Network Service during monitoring of the selected XenApp server. The default is unselected.
	This option is useful when the Client Network Service is on a different server than the one you are monitoring. When this option is enabled, the ServerFarmHealth job does not raise an event if it cannot locate the Client Network Service.
Ignore Encryption Service?	Select **Yes** to allow the script to ignore the Encryption Service during monitoring of the selected XenApp server. The default is unselected.
	This option is useful when the Encryption Service is on a different server than the one you are monitoring. When this option is enabled, the ServerFarmHealth job does not raise an event if it cannot locate the Encryption Service.
Ignore Independent Management Architecture Service?	Select **Yes** to allow the script to ignore the Independent Management Architecture Service during monitoring of the selected XenApp server. The default is unselected.
	This option is useful when the Independent Management Architecture Service is on a different server than the one you are monitoring. When this option is enabled, the ServerFarmHealth job does not raise an event if it cannot locate the Independent Management Architecture Service.
Ignore MFCom Service?	Select **Yes** to allow the script to ignore the MFCom Service during monitoring of the selected XenApp server. The default is unselected.
	This option is useful when the MFCom Service is on a different server than the one you are monitoring. When this option is enabled, the ServerFarmHealth job does not raise an event if it cannot locate the MFCom Service.

Description	How to Set It
Ignore Citrix Licensing Service?	Select **Yes** to allow the script to ignore the Licensing Service during monitoring of the selected XenApp server. The default is unselected.
	This option is useful when the Licensing Service is on a different server than the one you are monitoring. When this option is enabled, the ServerFarmHealth job does not raise an event if it cannot locate the Licensing Service.
Ignore Citrix Services Manager?	Select **Yes** to allow the script to ignore the Services Manager during monitoring of the selected XenApp server. The default is unselected.
	This option is useful when the Services Manager is on a different server than the one you are monitoring. When this option is enabled, the ServerFarmHealth job does not raise an event if it cannot locate the Services Manager.
Ignore Citrix XTE Server?	Select **Yes** to allow the script to ignore the XTE Server service during monitoring of the selected XenApp server. The default is unselected.
	This option is useful when the XTE Server is on a different computer than the one you are monitoring. When this option is enabled, the ServerFarmHealth job does not raise an event if it cannot locate the Citrix XTE Server service.
Ignore Citrix XML Server?	Select **Yes** to allow the script to ignore the XML Server service during monitoring of the selected XenApp server. The default is unselected.
	This option is useful when the XML Server is on a different computer than the one you are monitoring. When this option is enabled, the ServerFarmHealth job does not raise an event if it cannot locate the XML Server service.
Warning event threshold -- Maximum number of servers not responding	Specify the maximum number of servers that can be detected as not responding before a warning event is raised. The default is 3 servers.
Error event threshold -- Maximum number of servers not responding	Specify the maximum number of servers that can be detected as not responding before an error event is raised. The default is 10 servers.

3.12 ServerProcessesHigh

Use this Knowledge Script to monitor the number of XenApp processes across all sessions. If the number of server processes exceeds the specified threshold, an event is raised.

NOTE: To gather data about all sessions on a specific server in a XenApp farm, run this Knowledge Script on that individual server in the farm.

This script returns the number of processes generated by all sessions on XenApp server. The event detail message includes information about each process, such as process name, process state, process ID, and username.

3.12.1 Resource Object

Citrix XenApp object

3.12.2 Default Schedule

The default schedule is **Every 30 minutes**.

3.12.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the ServerProcessesHigh job fails. The default is 5.
Event Notification	
Raise event if number of processes exceeds the threshold?	Select **Yes** to raise an event if the number of XenApp processes across all sessions exceeds the specified threshold. The default is Yes.
Event severity when number of processes exceeds the threshold	Set the event severity level, from 1 to 40, to indicate the importance of the event. The default is 5.
Data Collection	
Collect data for number of processes?	Select **Yes** to collect data for charts and reports. If enabled, returns the number of XenApp processes across all sessions. The default is unselected.
Monitoring	
Threshold -- Maximum processes on a server	Specify the maximum number of processes allowed on a server across all sessions before an event is raised. The default is 50 processes.

3.13 ServerProcessesResourceHigh

Use this Knowledge Script to monitor the use of CPU and memory resources by processes on XenApp servers.

You can set thresholds for physical and virtual memory utilization and CPU utilization. If the use of resources by a process exceeds a threshold you set, an event is raised.

You can also configure the Knowledge Script to automatically terminate processes that exceed usage thresholds.

3.13.1 Resource Object

Citrix XenApp object

3.13.2   Default Schedule

The default schedule is **Every 30 minutes**.

3.13.3   Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the ServerProcessesResourceHigh job fails. The default is 5.
Event Notification	
Raise event if memory or CPU utilization exceeds threshold?	Select **Yes** to raise an event when the use of physical or virtual memory or CPU time exceeds the threshold you set. By default, events are enabled.
Event severity when memory or CPU utilization exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which memory or CPU utilization exceeds the threshold you set. The default is 5.
Data Collection	
Collect data for memory and CPU utilization?	Select **Yes** to collect data for charts and reports. If enabled, returns information about the use of physical and virtual memory (in KB) and CPU time (as a percentage). The default is unselected.
Monitoring	
Threshold -- Maximum physical memory utilization	Specify the maximum amount of physical memory that can be used by any single XenApp process before an event is raised. The default is 30720 KB.
Threshold -- Maximum virtual memory utilization	Specify the maximum amount of virtual memory that can be used by any single XenApp process before an event is raised. The default is 61440 KB.
Threshold -- Maximum CPU utilization	Specify the maximum percentage of CPU time that can be used by any single XenApp process before an event is raised. The default is 90%.
Processes to monitor (comma-separated, no spaces)	Provide the names of the XenApp processes you want to monitor. Separate multiple process names with commas and no spaces. For example, `Process1,Process2,Process3`. If no process names are entered, all processes are monitored. By default, all processes are monitored.
Terminate processes that exceed a threshold?	Select **Yes** to terminate any listed processes whose use of memory or CPU time exceeds the thresholds you set. The default is unselected.

3.14 ServerSessionHigh

Use this Knowledge Script to monitor the number of sessions on a XenApp server. If the number of sessions exceeds the threshold you set, an event is raised.

If data collection is enabled, this script returns the number of server sessions. The event detail message includes information about each session, such as session name, session ID, and username.

3.14.1 Resource Object

Citrix XenApp object

3.14.2 Default Schedule

The default schedule is **Every 30 minutes**.

3.14.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the ServerSessionHigh job fails. The default is 5.
Event Notification	
Raise event if number of sessions exceeds threshold?	Select **Yes** to raise an event if the number of server sessions exceeds the threshold. The default is Yes.
Event severity when number of sessions exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of sessions exceeds threshold. The default is 5.
Data Collection	
Collect data for number of sessions?	Select **Yes** to collect data for charts and reports. If enabled, returns the number of sessions, and information about each session. The default is unselected.
Monitoring	
Threshold -- Maximum number of sessions on a server	Specify the maximum number of sessions allowed on a server before an event is raised. The default is 20 sessions.

3.15 SessionPerUser

Use this Knowledge Script to monitor the number of sessions open for each user. You can monitor individual servers or entire server farms. If the number of sessions per user exceeds the threshold you specify, an event is raised.

3.15.1 Resource Object

Citrix XenApp object

3.15.2 Default Schedule

The default schedule is **Every 30 minutes**.

3.15.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the SessionPerUse job fails. The default is 5.
Event Notification	
Raise event if number of sessions exceeds threshold?	Select **Yes** to raise an event if the number of user sessions exceeds the threshold you set. The default is Yes.
Event severity when number of sessions exceeds threshold	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 you set. The default is 5.
Data Collection	
Collect data for number of sessions?	Select **Yes** to collect data for charts and reports. If enabled, returns the number of sessions on XenApp open for each user. The default is unselected.
Monitoring	
Threshold -- Maximum number of sessions	Specify the maximum number of sessions on XenApp that can be open for each user before an event is raised. The default is 5 sessions.
Monitor all servers in the farm?	Select **Yes** to monitor the number of sessions for all servers in a farm. The default is unselected.

3.16 SessionState

Use this Knowledge Script to monitor for Independent Computing Architecture (ICA) sessions that are in certain states. SessionState Knowledge Script can now monitor XenApp server sessions per farm, generating event messages by farm name instead of server name.

If the number of sessions matching the states you select for monitoring falls below the minimum threshold or exceeds the maximum threshold you set, an event is raised.

SessionState obtains a list of all sessions from the XenApp API and loops through that list, looking at the state of each session. As an example, set the **Minimum threshold** to 2 and the **Maximum threshold** to 4. If this Knowledge Script finds two sessions in LISTENING state, and one in ACTIVE

state, the number of sessions in LISTENING state is between the minimum and maximum thresholds, so the Knowledge Script will not raise an event for that state. The number of ACTIVE sessions has fallen below the minimum threshold, so the script raises an event for the ACTIVE state.

In a case like the one cited above, the Knowledge Script would not raise an event for any other session state, even if other states had fallen below the minimum threshold. It only raises events for a state if at least one session is in that particular state.

One use for this script is to track the number of active or idle XenApp sessions.

3.16.1   Resource Object

Citrix XenApp object

3.16.2   Default Schedule

The default schedule is **Every 30 minutes**.

3.16.3   Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the SessionState job fails. The default is 5.
Event Notification	
Raise event when threshold exceeded or not met?	Select **Yes** to raise an event if the number of sessions matching a specified state exceeds or falls below the maximum or minimum threshold. The default is Yes.
Event severity when threshold exceeded or not met	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the number of sessions exceeds or falls below the threshold you set. The default is 5.
Data Collection	
Collect data for number of server sessions in specified state?	Select **Yes** to collect data for charts and reports. If enabled, returns the number of sessions matching specified states. The default is unselected.
Monitoring	
Threshold -- Minimum number of sessions matching specified states	Specify the minimum number of sessions whose states must match the states you selected for monitoring before an event is raised. The default is 0 sessions (disabled).
Threshold -- Maximum number of sessions matching specified states	Specify the maximum number of sessions whose states can match the states you selected for monitoring before an event is raised. The default is 5 sessions.
Session States to Monitor	

Description	How to Set It
All session states	Select **Yes** for each type of session state you want to monitor. By default, only **All session states** is set to Yes.
Active	
Connected	
Connecting	
Disconnected	
Down	
Idle	
Initializing	
Licensed	
Listening	
Reconnected	
Resetting	
Shadowing	
Stale	
Unlicensed	

3.17 UserResourcesHigh

Use this Knowledge Script to monitor the utilization of CPU time and memory resources by users connected to XenApp. You can select which users to monitor and set thresholds for physical or virtual memory utilization or CPU utilization.

Monitoring a user's processes occurs on a per-process basis. Resource utilization is only measured for the processes being used by the user selected for monitoring. However, the utilization metrics of different processes are not aggregated per user. All users on the server where you dropped the Knowledge Script are monitored by default. When user runs multiple instances of a process, each process instance has a pound sign (#) then a number after the process name so you can easily see how many instances of that process are running for that user.

If the percentage of CPU time or the amount of physical or virtual memory used by a process exceeds a threshold you set, an event is raised.

3.17.1 Resource Object

Citrix XenApp object

3.17.2 Default Schedule

The default schedule is **Every 30 minutes**.

3.17.3 Setting Parameter Values

Set the following parameters as needed:

Description	How to Set It
General Settings	
Job failure event notification	
Event severity when job fails	Set the severity level, from 1 to 40, to indicate the importance of an event in which the UserResourcesHigh job fails. The default is 5.
Event Notification	
Raise event when CPU or memory utilization exceeds threshold?	Select **Yes** to raise an event when the use of CPU or memory resources by users connected to XenApp exceeds any threshold you set. The default is Yes.
Event severity when CPU or memory utilization exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the CPU or memory utilization exceeds the threshold you set. The default is 5.
Data Collection	
Collect data for CPU and memory utilization?	Select **Yes** to collect data for charts and reports. If enabled, returns information about the use of CPU and memory resources by users connected to XenApp. The default is unselected.
Monitoring	
Threshold -- Maximum physical memory utilization	Specify the maximum amount of physical memory that can be consumed by users connected to XenApp before an event is raised. The default is 30720 KB.
Threshold -- Maximum virtual memory utilization	Specify the maximum amount of virtual memory that can be consumed by users connected to XenApp before an event is raised. The default is 61440 KB.
Threshold -- Maximum CPU utilization	Specify the maximum percentage of CPU time that can be consumed by users connected to XenApp before an event is raised. The default is 80%.
Users to monitor (comma-separated, no spaces)	Provide the names of the users you want to monitor. Separate names in a list with commas and no spaces (for example, User1,User2,User3). If no names are entered, all users are monitored. By default, all users are monitored.