

---

# Management Guide

## NetIQ® AppManager® for Self Monitoring (AM Health)

December 2018

## Legal Notice

For information about NetIQ legal notices, disclaimers, warranties, export and other use restrictions, U.S. Government restricted rights, patent policy, and FIPS compliance, see <https://www.netiq.com/company/legal/>.

© 2018 NetIQ Corporation. All Rights Reserved.

For information about NetIQ trademarks, see <https://www.netiq.com/company/legal/>. All third-party trademarks are the property of their respective owners.

---

# Contents

<b>About this Book and the Library</b>	<b>5</b>
<b>About NetIQ Corporation</b>	<b>7</b>
<b>1 Introducing AppManager for Self Monitoring</b>	<b>9</b>
<b>2 Installing AppManager for Self Monitoring</b>	<b>11</b>
2.1 System Requirements . . . . .	11
2.2 Installing the Module . . . . .	12
2.3 Silently Installing the Module . . . . .	12
2.4 Configuring Security Manager for AM Health. . . . .	13
2.5 Discovering AppManager Resources for Windows . . . . .	14
2.6 Discovering AppManager Resources for UNIX and Linux Servers . . . . .	15
2.7 Upgrading Knowledge Script Jobs . . . . .	16
<b>3 AM Health Knowledge Scripts</b>	<b>19</b>
3.1 AgentDown. . . . .	19
3.2 CCComponentsHealth . . . . .	22
3.3 HeartbeatUNIX. . . . .	30
3.4 HeartbeatWin . . . . .	34
3.5 QDBComponentsHealth. . . . .	37
3.6 TaskSchedulerHealth . . . . .	45
3.7 Recommended Knowledge Script Groups . . . . .	47



# 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 Self Monitoring provides system administrators with a central, easy-to-use console to view critical server and application resources across the enterprise. With AppManager Self Monitoring, 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 Self Monitoring library is available in Adobe Acrobat (PDF) format from the [AppManager Documentation](#) page of the NetIQ website.



# 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.

<b>Worldwide:</b>	<a href="http://www.netiq.com/about_netiq/officelocations.asp">www.netiq.com/about_netiq/officelocations.asp</a>
<b>United States and Canada:</b>	1-888-323-6768
<b>Email:</b>	<a href="mailto:info@netiq.com">info@netiq.com</a>
<b>Website:</b>	<a href="http://www.netiq.com">www.netiq.com</a>

## Contacting Technical Support

For specific product issues, contact our Technical Support team.

<b>Worldwide:</b>	<a href="http://www.netiq.com/support/contactinfo.asp">www.netiq.com/support/contactinfo.asp</a>
<b>North and South America:</b>	1-713-418-5555
<b>Europe, Middle East, and Africa:</b>	+353 (0) 91-782 677
<b>Email:</b>	<a href="mailto:support@netiq.com">support@netiq.com</a>
<b>Website:</b>	<a href="http://www.netiq.com/support">www.netiq.com/support</a>

## Contacting Documentation Support

Our goal is to provide documentation that meets your needs. The documentation for this product is available on the NetIQ website 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](http://www.netiq.com/documentation). You can also email [Documentation-Feedback@netiq.com](mailto: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](http://community.netiq.com).



# 1 Introducing AppManager for Self Monitoring

The AppManager Self Monitoring module, also known as AM Health, provides Knowledge Scripts for monitoring the health and availability of AppManager components.

You can use the AM Health module to monitor the following AppManager components and features:

- ◆ Health and availability of Microsoft SQL Server resources associated with the AppManager repository (QDB) and the management server. These resources include SQL Server services, management server services, and SQL Server queries.
- ◆ Health and availability of Microsoft SQL Server resources associated with AppManager Control Center components. These components include the Cache Manager, the Command Queue Service (CQS), deployment services, and the Control Center repository (NQCCDB).
- ◆ Heartbeat of AppManager Windows agent computers. A **heartbeat** is a periodic signal generated by an AppManager agent computer to indicate that it is still running. If an AppManager agent fails to send either data or an event to the QDB within the specified grace period, AppManager assumes that the agent is offline and generates a corresponding event.
- ◆ Heartbeat of AppManager UNIX agent computers.



# 2 Installing AppManager for Self Monitoring

This chapter provides installation instructions and describes system requirements for AppManager for Self Monitoring (AM Health).

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 Self Monitoring has the following system requirements:

Software/Hardware	Version
NetIQ AppManager installed on the AppManager repository (QDB) computer and on all console computers	8.0.3, 8.2, 9.1, 9.2, 9.5, or later  One of the following AppManager agents are required: <ul style="list-style-type: none"><li>◆ AppManager agent 7.0.4 with hotfix 72616 or later</li><li>◆ AppManager agent 8.0.3, 8.2, 9.1, 9.2, 9.5, or later</li></ul>
NetIQ AppManager for Microsoft Windows module installed on the AppManager agent computers	7.6.170.0 or later
Microsoft .NET Framework installed on the AppManager agent computers	2.0 or 3.5  <b>NOTE:</b> If you have only installed .NET Framework 4.0 or a later version, then you must also install AppManager for Microsoft Windows 8.1.82.0.
Microsoft SQL Server Native Client 11.0 <b>(for TLS 1.2 support)</b>	11.3.6538.0 or later  <b>NOTE:</b> The SQL Server Native client can be installed from this <a href="#">Microsoft download link</a> .

**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 only once on any Windows computer. The module installer automatically identifies and updates all relevant AppManager components on a computer.

Access the `AM70-AMHealth-8.2.x.0.msi` module installer from the `AM70_AMHealth_8.2.x.0` self-extracting installation package on the [AppManager Module Upgrades & Trials](#) page.

You can install the Knowledge Scripts and the Analysis Center reports 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.

### To install the module manually:

- 1 Double-click the module installer `.msi` file.
- 2 Accept the license agreement.
- 3 Run the module installer on all console computers to install the Help and console extensions.
- 4 To install the Knowledge Scripts into the QDB:
  - 4a Select **Install Knowledge Scripts** to install the repository components, including the Knowledge Scripts.
  - 4b Specify the SQL Server name of the server hosting the QDB, as well as the case-sensitive QDB name.
- 5 Run the module installer only for the primary QDB. Control Center automatically replicates this module to secondary QDBs.
- 6 (Conditional) If the service account does not have sufficient privileges to run `Discovery_AMHealth`, set up SQL authentication with AppManager Security Manager. For more information, see [Section 2.4, "Configuring Security Manager for AM Health,"](#) on page 13.
- 7 (Conditional) If you have not discovered AM Health Windows resources, run the `Discovery_AMHealth` Knowledge Script on all Windows agent computers where you installed the module. For more information, see [Section 2.5, "Discovering AppManager Resources for Windows,"](#) on page 14.
- 8 (Conditional) If you have not discovered AM Health UNIX or Linux resources, run the `Discovery_AMHealthUNIX` Knowledge Script on all UNIX or Linux agent computers where you installed the module. For more information, see [Section 2.6, "Discovering AppManager Resources for UNIX and Linux Servers,"](#) on page 15.
- 9 To get the updates provided in this release, upgrade any running Knowledge Script jobs. For more information, see [Section 2.7, "Upgrading Knowledge Script Jobs,"](#) on page 16.

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

## 2.3 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-AMHealth-8.2.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 installer, add the following flag to the command noted above:

```
/I* "AM70-AMHealth-8.2.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-AMHealth-8.2.x.0.msi /qn MO_B_QDBINSTALL=1 MO_B_SQLSVR_WINAUTH=1  
MO_SQLSVR_NAME=SQLServerName MO_QDBNAME=AM-RepositoryName
```

**SQL authentication:**

```
AM70-AMHealth-8.2.x.0.msi /qn MO_B_QDBINSTALL=1 MO_B_SQLSVR_WINAUTH=0  
MO_SQLSVR_USER=SQLLogin MO_SQLSVR_PWD=SQLLoginPassword  
MO_SQLSVR_NAME=SQLServerName MO_QDBNAME=AM-RepositoryName
```

## 2.4 Configuring Security Manager for AM Health

If the service account does not have sufficient privileges, before running the Knowledge Script, set up SQL authentication with AppManager Security Manager.

On the **Custom** tab in Security Manager, complete the following fields:

---

Field	Description
Label	sql\$<agent server name>  For example, if your agent server name is SERVER1TEST, you would type sql\$SERVER1TEST.  <b>IMPORTANT:</b> The server name must be specified in the NetBIOS format only.
Sub-label	SQL user name that exists in the QDB.
Value 1	Password for the user entered in the <b>Sub-Label</b> field.
Value 2	Leave this field blank.
Value 3	Leave this field blank.
Extended application support	Required field. Encrypts the user name and password in Security Manager. Do not leave this option unselected.

---

**NOTE:** To discover or monitor the QDB and CCDB roles on a cluster instance, you must configure Security Manager on all cluster nodes.

---

When you want to discover the management server, use the SQL user name used above for the *SQL Server login* parameter of the Discovery\_AMHealth Knowledge Script. For more information about discovering AM Health resources, see [Section 2.5, “Discovering AppManager Resources for Windows,” on page 14](#).

If an agent is not installed on the QDB, and the NetIQ Client Resource Monitor (`netiqmc`) service accounts on the management server do not have sufficient rights to access the QDB, configure the NetIQ services with a Windows account that has access to the QDB. Otherwise, configure the Discovery\_AMHealth Knowledge Script to use SQL authentication by typing a SQL user name in the *SQL Server login* parameter for Discovery\_AMHealth. For example, `sql$<management server name>`.

For example, if your management server name is SERVER1TEST, you would type `sql$SERVER1TEST`.

If an agent is not installed on the Control Center repository (NQCCDB), and the `netiqmc` service accounts on the NetIQ Command Queue Server do not have sufficient rights to access the NQCCDB, configure the NetIQ service accounts with a Windows account that has access to the NQCCDB. Otherwise, configure the Discovery\_AMHealth Knowledge Script to use SQL authentication. For example, `sql$<command queue server>`.

For example, if your command queue server name is SERVER1TEST, you would type `sql$SERVER1TEST`.

If an agent is installed on QDB or NQCCDB, and the `netiqmc` service accounts on the NetIQ Command Queue Server do not have sufficient rights to access QDB or NQCCDB, you can configure SQL user of the agent in AppManager Security Manager to monitor the health and availability of AppManager components. For example, `sql$<agent server name>`.

For example, if your agent server name is SERVER1TEST, you would type `sql$SERVER1TEST`.

You can also use the AppManager Security Manager configuration listed above to monitor the health of a management server that is in an untrusted domain from your AppManager installation. Use a SQL Server user to allow the AM Health Knowledge Scripts on the management server in the untrusted domain to communicate with the QDB on the SQL Server. You cannot use Windows authentication, because the SQL Server will not be aware of any users that belong to the untrusted domain.

## 2.5 Discovering AppManager Resources for Windows

Use the Discovery\_AMHealth Knowledge Script to discover AppManager and Control Center resources installed on Windows servers. This Knowledge Script returns information about successful and failed discoveries, and it raises events to notify you of errors.

Run the Discovery\_AMHealth script on computers with the one or more of the following AppManager components: AppManager repository (QDB), management server, Cache Manager, Command Queue Service (CQS), deployment service, AppManager managed client, Control Center repository (NQCCDB), and Task Scheduler service.

In a clustered environment, the Discovery\_AMHealth script discovers only QDB and CCDB roles. For more information about the setup of the AMHealth Module's AMHealth\_QDBComponentsHealth Knowledge Script in a clustered environment, contact [NetIQ Technical Support](#).

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

Set the following parameters as needed:

Parameter	How to Set It
<b>Event Notification</b>	
<b>Raise an event if discovery succeeds?</b>	Select <b>Yes</b> 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.
<b>Raise event if discovery fails?</b>	Select <b>Yes</b> 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 authentication.  <b>NOTE:</b> 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. For more information, see <a href="#">Section 2.4, "Configuring Security Manager for AM Health," on page 13.</a>

## 2.6 Discovering AppManager Resources for UNIX and Linux Servers

Use the Discovery\_AMHealthUNIX Knowledge Script to discover AppManager resources installed on UNIX and Linux servers. This Knowledge Script returns information about successful, failed, and partial discoveries, and it raises events to notify you of errors.

Run the Discovery\_AMHealthUNIX script on AppManager UNIX agents. By default, this script runs once for each computer.

Set the following parameters as needed:

Parameter	How to Set It
Raise event if discovery succeeds?	Set to <b>y</b> 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 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 partially succeeds	Set the event severity level, from 1 to 40, to reflect the importance when the discovery partially succeeds. This type of situation usually occurs when the target computer does not have all the prerequisites installed. The default is 15.

## 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 <i>retains</i> any changes you might have made to the parameter settings for the Knowledge Scripts in the previous version of this module.



# 3 AM Health Knowledge Scripts

The AppManager Self Monitoring module, also known as AM Health, provides Knowledge Scripts for monitoring the health and availability of AppManager components.

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
<a href="#">AgentDown</a>	Monitors the health of AppManager agents and raises events when agents are down or unavailable.
<a href="#">CCComponentsHealth</a>	Monitors the health and availability of SQL Server resources associated with Control Center components.
<a href="#">HeartbeatUNIX</a>	Monitors the AppManager agent heartbeat on a UNIX or Linux server. Also monitors Knowledge Script job run time against the job schedule and against a user-defined maximum run time.
<a href="#">HeartbeatWin</a>	Monitors the AppManager agent heartbeat in a Microsoft Windows environment.
<a href="#">TaskSchedulerHealth</a>	Monitors the health status of Task Scheduler that schedules SQL jobs for QDB and NQCCDB.
<a href="#">QDBComponentsHealth</a>	Monitors the health and availability of SQL Server resources associated with the AppManager repository (QDB) and the management server.
<a href="#">Recommended Knowledge Script Groups</a>	Perform essential monitoring of AppManager components. These groups include AMHealth_HealthCheckAMAgentComponents and AMHealth_HealthCheckAMCoreComponents

## 3.1 AgentDown

Use this Knowledge Script to monitor the health of AppManager agents, raise events when agents are down or unavailable, and generate actions as appropriate. This script does not raise events when AppManager agents are up (running) and available. You can set the event severity for an individual agent, or you can override the severity with this script to specify one severity for any agent with issues.

This script complements the AMHealth\_HeartbeatWin and AMHealth\_HeartbeatUNIX Knowledge Scripts, but it is different in the following important ways:

- ♦ Run the AMHealth\_AgentDown Knowledge Script on management servers. This script queries information from the AppManager repository (QDB) to obtain agent status, which is populated by the AMHealth\_Heartbeat jobs. As a result, you must run AMHealth\_Heartbeat jobs on your

agents as well as run an AMHealth\_AgentDown job on your management server. One AgentDown job can monitor all agents for which the heartbeat is running. The AMHealth\_AgentDown script supports the execution of actions.

- ♦ Run the relevant AMHealth\_HeartBeat Knowledge Scripts on the agents you wish to monitor. Select **Yes** for either the **Raise an event if the agent heartbeat fails?** parameter or the **Generate heartbeat data?** parameter, or select both parameters. The AMHealth\_Heartbeat scripts do not support the execution of actions.

You can run this script on multiple management servers for redundancy. In this situation, each AMHealth\_AgentDown job raises duplicate events and generates duplicate actions.

If you stop and restart AMHealth\_HeartbeatWin and AMHealth\_HeartbeatUNIX jobs, the AMHealth\_AgentDown job might raise duplicate events.

### 3.1.1 Prerequisites

- ♦ AppManager version 8.0.2 or later on the QDB and management server
- ♦ AppManager version 8.0 or later on the AppManager agent
- ♦ AMHealth version 8.0.113.0 or later on the QDB and all console computers

### 3.1.2 Configuring Security Manager for AMHealth\_AgentDown

The AMHealth\_AgentDown Knowledge Script requires credentials to connect to the AppManager Repository (QDB). If the QDB uses SQL authentication, create the following entry in Security Manager on the **Custom** tab:

Field	Description
Label	AMH\$SQL
Sub-label	SQL user name with authority to access the QDB.
Value 1	SQL password for the user name entered in the <b>Sub-Label</b> field.
Value 2	Leave this field blank.
Value 3	Leave this field blank.
Extended application support	Encrypts the user name and password in Security Manager. This option must be selected.

### 3.1.3 Resource Objects

Management server

### 3.1.4 Default Schedule

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

### 3.1.5 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
<b>General Settings</b>	
<b>Job Failure Notification</b>	
<b>Raise event if job fails unexpectedly?</b>	Select <b>Yes</b> to raise an event if the AMHealth_AgentDown job fails unexpectedly. The default is Yes.
Event severity when job fails unexpectedly	Set the event severity level, from 1 to 40, to indicate the importance of an event that is raised when the job fails unexpectedly. The default is 5.
<b>Additional Settings</b>	
<b>Event Details</b>	
Event detail format	<p>Select the format in which you want to display the event detail. You can select from <i>HTML Table</i> or <i>Plain Text</i>. The default is HTML Table.</p> <p>Regardless of what you enter for this parameter, if the Action_SMTPEmail action Knowledge Script is associated with an AMHealth_AgentDown job, the email displays in plain text.</p>
<b>Knowledge Script Settings</b>	
Always consolidate events?	<p>Select <b>Yes</b> to raise a single event for all down agents, regardless of how many agents are down. The default is Yes.</p> <p>Select <b>No</b> to use the <b>Raise consolidated event if X percent of agents are down</b> setting in Control Center for determining whether to raise a single event for all down agents, or one event per down agent. You can edit the Health Check settings in Control Center by clicking <b>Options</b> on the <b>Main</b> tab, and then clicking <b>Health Check</b>.</p>
SQL logon	<p>Specify the SQL Server user name required for access to the AppManager repository (QDB).</p> <p>This setting requires an entry in AppManager Security Manager, as described in <a href="#">“Configuring Security Manager for AMHealth_AgentDown” on page 20</a>.</p> <p>Leave this field blank for Windows Authentication.</p>
<b>Event Settings</b>	

Parameter	How to Set It
Override agent down severity?	<p>Select <b>Yes</b> to use this script to specify the event severity when an agent is down. If you select No, AppManager uses the event severity for a down agent that is specified by the AMHealth_HeartbeatWin and AMHealth_HeartbeatUNIX Knowledge Scripts. The default is Yes.</p> <p>To prevent the duplication of agent-down events generated by both the AMHealth_Heartbeat and AMHealth_AgentDown jobs, select <b>Yes</b> and set the <b>Severity for override</b> parameter in this script to 40. Next, choose <i>one</i> of the following options:</p> <ul style="list-style-type: none"> <li>◆ In the Operator Console, open the <b>File</b> menu and select <b>Preferences</b>. On the <b>Repository</b> tab, click <b>Event</b>. Select <b>Automatically close event when severity is greater than X</b> and set it to 39. Using this option, events raised by AgentDown immediately go to Closed state, but actions are still executed.</li> <li>◆ On the Management Server, open the registry and navigate to <code>\SOFTWARE\...\NetIQ\AppManager\4.0\NetIQmc\Config</code>, and set the <code>NoEventSev</code> registry key to 40. Then stop and restart the NetIQMC service. With this option, the AMHealth_AgentDown job does not raise events, but the script still executes actions.</li> </ul>
Severity for override	<p>Set the event severity level, from 1 to 40, to indicate the importance of an event when an agent is detected as being down or unavailable. This parameter is ignored if you do not select the <b>Override agent down severity?</b> parameter. The default is 40.</p>

## 3.2 CCComponentsHealth

Use this Knowledge Script to monitor the health and availability of Microsoft SQL Server resources associated with AppManager Control Center components. These components include the Cache Manager, the Command Queue Service (CQS), deployment services, and the Control Center repository (NQCCDB).

This script monitors the percentage of database space and log space used, the amount of time required for a SQL command or query to execute, and the status of AppManager scheduled tasks, which are SQL jobs operating on the NQCCDB. This script can restart a service or job that is down.

If the NQCCDB does not have an AppManager agent installed on it, Discovery\_AMHealth discovers the NQCCDB components on the server running the Command Queue Service. As a result, the service and database monitoring parameters for this script run remotely. In this situation, you must have sufficient privileges on the service account for the NetIQMC service on the server running the Command Queue Service so that the service account can remotely access the SQL Server service on the NQCCDB to obtain its status.

If the account does not have proper privileges, the script will be unable to access the service status and will report that the SQL Server service is down even when it is not. If you do not have sufficient access for the service account for the NetIQMC service, deselect the *Raise an event if SQL Server services are down* and *Restart SQL Server services that stop unexpectedly* parameters in this script to avoid raising unnecessary events.

This script raises events for the following situations:

- ◆ SQL Server services are down or have been restarted.
- ◆ SQL jobs are disabled, missing, or have failed.

- ♦ The CQS is down or is not connected to the Control Center repository.
- ♦ Control Center Cache Manager errors occur.
- ♦ SQL Server queries against the NQCCDB take too long to process.
- ♦ Deployment services are down.
- ♦ Database or log space is low, and there is insufficient disk space for further growth.
- ♦ Primary repository is changed for Control Center
- ♦ Management group rules are out of sync between the QDB and the CCDB
- ♦ CCDB processing threads executed by CQS have critical, major, or minor errors

If you do not have an agent installed on the NQCCDB server, the repository component gets discovered on the CQS. If you try to remotely monitor the NQCCDB from the CQS using the CComponentsHealth script, the script will not be able to obtain the disk information remotely.

As a result, if the repository component is monitored remotely from the CQS by the CComponentsHealth script, the following NQCCDB component monitoring parameters under the *SQL Server File Size and Growth Settings Monitoring* Event Notification Knowledge Script section will not be available:

- ♦ Raise an event if SQL Server maximum file size exceeds available disk space?
- ♦ Raise an event if insufficient space available for further file growth?

### 3.2.1 Resource Objects

- ♦ Cache Manager
- ♦ CQS
- ♦ Deployment services
- ♦ Control Center repository

### 3.2.2 Default Schedule

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

### 3.2.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
<b>General Settings</b>	
<b>Job Failure Notification</b>	
<b>Raise event if job fails unexpectedly?</b>	Select <b>Yes</b> to raise an event if the AMHealth_CComponentsHealth job fails unexpectedly. The default is Yes.
Event severity when job fails unexpectedly	Set the event severity level, from 1 to 40, to indicate the importance of an event that is raised when the job fails unexpectedly. The default is 5.
<b>Additional Settings</b>	
<b>Event Details</b>	

Parameter	How to Set It
Event detail format	Select the format in which you want to display the event detail. You can select from <i>HTML Table</i> or <i>Plain Text</i> . The default is HTML Table.
<b>Event Notification</b>	
<b>Raise an event if SQL Server services are down?</b> (Use MC based Action Script for notification)	Select <b>Yes</b> to raise an event if Control Center SQL Server services are down. The default is Yes.  <b>Tip</b> Use Action Script for notification, as the Control Center repository will not be available.
Event severity when SQL Server services are down	Set the event severity level, from 1 to 40, to indicate the importance of an event in which SQL Server services are down. The default is 10.
<b>Restart SQL Server services that stop unexpectedly?</b>	Select <b>Yes</b> to restart SQL Server services that stop unexpectedly, such as when not as part of scheduled maintenance. The default is Yes.
<b>Raise an event if SQL Server services cannot be restarted?</b>	Select <b>Yes</b> to raise an event if SQL Server services cannot be restarted. The default is Yes.  <b>Tip</b> Use Action Script for notification, as the Control Center repository will not be available.
Event severity when SQL Server services cannot be restarted	Set the event severity, from 1 to 40, to indicate the importance of an event in which SQL Server services cannot be restarted. The default is 5.
<b>Raise an event if SQL Server services are restarted?</b>	Select <b>Yes</b> to raise an event if SQL Server services are successfully restarted. The default is Yes.
Event severity when SQL Server services are restarted	Set the event severity level, from 1 to 40, to indicate the importance of an event in which SQL Server services are successfully restarted. The default is 25.
<b>Raise an event if Control Center Cache Manager errors occur?</b>	Select <b>Yes</b> to raise an event if the Control Center Cache Manager experiences one of the following: <ul style="list-style-type: none"> <li>◆ An error related to Microsoft Distributed Transaction Control (MSDTC).</li> <li>◆ An error with the synchronization of data between Control Center and the QDB repositories that are synchronized to Control Center.</li> <li>◆ An error from a Control Center SQL Server agent job that has failed.</li> </ul> <p>The default is Yes.</p> <p><b>NOTE:</b> Not all Control Center SQL jobs are monitored. Only those that are related to Cache Manager are monitored.</p> <p>The following jobs are monitored for errors that affect Cache Manager:</p> <ul style="list-style-type: none"> <li>◆ NetIQ CC Manage SQL Jobs NQCCDB</li> <li>◆ NetIQ CC Hourly Task NQCCDB</li> <li>◆ NetIQ CC Half-Hourly Task NQCCDB</li> <li>◆ NetIQ CC Daily Task NQCCDB</li> </ul> <p>Cache Manager is a child process of the CQS. Cache Manager runs the Control Center queries on AppManager repository computers to retrieve view information for Control Center s. After the first iteration of the Knowledge Script completes, only new errors generated since the previous iteration will be raised as events.</p>



Parameter	How to Set It
Event severity when Cache Manager errors occur	Set the event severity, from 1 to 40, to indicate the importance of an event in which Cache Manager experiences errors. The default is 10.
<b>Raise an event if Control Center SQL jobs are missing?</b>	Select <b>Yes</b> to raise an event if SQL jobs are missing. The default is Yes. A missing job is one that might have been deleted or renamed.
Event severity when Control Center SQL jobs are missing	Set the event severity, from 1 to 40, to indicate the importance of an event in which SQL jobs are missing. The default is 15.
<b>Raise an event if Control Center SQL jobs are disabled?</b>	Select <b>Yes</b> to raise an event if SQL jobs are disabled. The default is Yes.
Event severity when Control Center SQL jobs are disabled	Set the event severity, from 1 to 40, to indicate the importance of an event in which SQL jobs are disabled. The default is 15.
<b>Enable Control Center SQL jobs that are disabled?</b>	Select <b>Yes</b> to enable, or start, SQL jobs that are disabled. The default is No.
Event severity when SQL jobs cannot be enabled	Set the event severity, from 1 to 40, to indicate the importance of an event in which disabled SQL jobs cannot be enabled, or started. The default is 20.
<b>Raise an event if Control Center SQL jobs are successfully enabled?</b>	Select <b>Yes</b> to raise an event if disabled SQL jobs are successfully enabled, or started. The default is No.
Event severity when Control Center SQL jobs are successfully enabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which disabled SQL jobs are successfully enabled, or started. The default is 20.
<b>Raise an event if a Control Center SQL Server job fails?</b>	Select <b>Yes</b> to raise an event if any step in a SQL Server job fails. The default is Yes.
Event severity when a Control Center SQL Server job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a SQL Server job fails. The default is 20.
Raise an event if Control Center server disks are fragmented?	Select <b>Yes</b> to raise an event if disks on the Control Center server are fragmented. The default is Yes.  <b>Tip</b> To avoid errors when running this script on a Windows Server 2008 server, disable User Account Control (UAC).
Event severity when Control Center server disks are fragmented	Set the event severity level, from 1 to 40, to indicate the importance of an event in which disk fragmentation has occurred on the Control Center server. The default is 15.
<b>Raise an event if Control Center database size exceeds threshold?</b>	Select <b>Yes</b> to raise an event if the Control Center database size exceeds the threshold you set. The default is Yes.  <b>NOTE:</b> This parameter is applicable only on AppManager 9.1 running on SQL Server Express edition.
Threshold - Maximum Control Center database size	Specify how large the Control Center database file can grow before raising an event. If the database file size is too small, AppManager performance will not be optimal in larger environments. The default is 8 GBytes.
Event severity when Control Center database size exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the Control Center database file size exceeds the threshold. The default is 20.

Parameter	How to Set It
<b>Raise an event if query process time exceeds threshold?</b>	Select <b>Yes</b> to raise an event if the amount of time it takes to process a SQL query exceeds the threshold you set. The default is Yes.
Threshold -- Maximum process run time	Specify the maximum length of time it can take SQL Server processes to run before an event is raised. The default is 300 seconds.
Event severity when query process time exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which SQL query processing time exceeds the threshold you set. The default is 10.
Event severity when query process time cannot be retrieved	<p>Set the event severity level, from 1 to 40, to indicate the importance of an event in which the script cannot determine the processing time for SQL queries.</p> <p>Processing time could be prevented from being retrieved if the query fails due to blocking, or if a connection cannot be made to the Control Center QDB SQL Server, which would cause the query to fail. If the query fails, the event will be raised without exception, which would inform users if there was an issue with the Control Center QDB SQL Server.</p> <p>The default is 10.</p>
<b>Raise an event if Deployment services are down?</b>	<p>Select <b>Yes</b> to raise an event if Control Center deployment services are down. The default is Yes.</p> <p>The following situations can cause deployment services to be down:</p> <ul style="list-style-type: none"> <li>◆ The Control Center server is restarted.</li> <li>◆ Deployment services took too long to restart.</li> <li>◆ The connection to the Control Center repository is unavailable.</li> <li>◆ The proxy server is unavailable.</li> </ul>
Event severity when Deployment services are down	Set the event severity level, from 1 to 40, to indicate the importance of an event in which deployment services are down. The default is 10.
<b>Raise an event if unable to retrieve Control Center component information?</b>	Select <b>Yes</b> to raise an event if the script does not have the discovery details for the Control Center components, or if the discovery details are empty. The default is Yes.
Event severity when unable to retrieve Control Center component information	Set the event severity level, from 1 to 40, to indicate the importance of an event in which AppManager cannot retrieve Control Center Component information. The default is 10.
<b>Raise an event if primary repository for Control Center is changed?</b>	Select <b>Yes</b> to raise an event if the primary repository is changed for the Control Center. The default is Yes.
Event severity when primary repository for Control Center is changed	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the primary repository for Control Center is changed. The default is 25.
<b>Raise an event if rules are out of sync between QDB and CCDB</b>	<p>Select <b>Yes</b> to raise an event if the management group rules that are defined in the CCDB are not synchronized with the QDB. The default is Yes.</p> <p><b>NOTE:</b> This parameter runs every 24 hours irrespective of the default interval of the Knowledge Script.</p>

Parameter	How to Set It
Event severity when rules are out of sync between QDB and CCDB	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the management group rules in the QDB and CCDB are not synchronized. The default is 5.
Timeout	Specify the number of seconds that the Knowledge Script should wait to get the sync status of the management group rules between the CCDB and the QDB before raising a timeout exception. The default is 60 seconds.
<b>NetIQ Control Center Command Queue Service Monitoring</b>	
<b>Raise an event if Command Queue Service is not connected to the Control Center database?</b>	Select <b>Yes</b> to raise an event if the Command Queue Service (CQS) is not connected to the Control Center database (NQCCDB). The default is Yes.  The CQS polls the Command Queue table for queries to run. The Command Queue table stores queries that collect view information based on the criteria defined in Control Center.
Event severity when Command Queue Service is not connected to the Control Center database	Set the event severity, from 1 to 40, to indicate the importance of an event in which the CQS is not connected to the Control Center database (NQCCDB). The default is 10.
<b>Raise an event if Command Queue Service is down?</b>	Select <b>Yes</b> to raise an event if the CQS is down. The default is Yes.
Event severity when Command Queue Service is down	Set the event severity level, from 1 to 40, to indicate the importance of an event in which CQS is down. The default is 10.
<b>Restart Command Queue Service if the service is down?</b>	Select <b>Yes</b> to restart CQS if it is down. The default is Yes.
<b>Raise an event if Command Queue Service is restarted?</b>	Select <b>Yes</b> to raise an event if CQS is successfully restarted. The default is Yes.
Event severity when Command Queue Service is restarted	Set the event severity level, from 1 to 40, to indicate the importance of an event in which CQS is successfully restarted. The default is 25.
<b>Raise an event if Command Queue Service cannot be restarted?</b>	Select <b>Yes</b> to raise an event if CQS cannot be restarted. The default is Yes.
Event severity when Command Queue Service cannot be restarted	Set the event severity level, from 1 to 40, to indicate the importance of an event in which CQS cannot be restarted. The default is 5.
<b>SQL Server File Size and Growth Settings Monitoring</b>	
<b>Raise an event if insufficient space available for further file growth?</b>	Select <b>Yes</b> to raise an event if the amount of available disk space is not enough to allow the file to continue to grow.  SQL Server has growth settings on the Database Data and Log files. If the amount of free space on the disk is lower than this growth setting, an event will be raised to keep the files from attempting to grow and causing the SQL Server databases to become corrupted.  The default is Yes.
Event severity when available space is insufficient	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the available server space is insufficient. The default is 10.

Parameter	How to Set It
<b>Raise an event if file growth rate falls below the threshold?</b>	<p>Select <b>Yes</b> to raise an event if the growth rate of the database data and log files falls below the threshold you set. If the growth rate is low, the data and log files become fragmented, which negatively impacts performance.</p> <p><b>NOTE:</b> The two threshold parameters (MB and percentage) are evaluated individually depending on whether the database file growth setting is in MB or a percentage.</p> <p>The default is Yes.</p>
Threshold -- Minimum growth rate in MB	Specify in megabytes the minimum growth rate of SQL Server files before an event is raised. The default is 256 MB.
Threshold -- Minimum growth rate in percentage	Specify as a percentage the minimum growth rate of SQL Server files before an event is raised. The default is 9%.
Event severity when file growth rate falls below the threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the rate of growth of SQL Server files falls below the threshold you set. The default is 10.
<b>Raise an event if Autogrowth is not enabled and file usage exceeds the threshold?</b>	<p>Select <b>Yes</b> to raise an event if the file usage exceeds the threshold and Autogrowth is not enabled. The default is Yes.</p> <p>The Autogrowth feature allows a database to grow by the amount of space required by a file or a transaction.</p>
Threshold -- Maximum file usage with Autogrowth disabled	Specify the maximum amount of file usage with Autogrowth disabled that can occur before an event is raised. The default is 90%.
Event severity when Autogrowth disabled and usage exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the file usage exceeds the threshold and Autogrowth is not enabled. The default is 10.
<b>Raise an event if SQL Server maximum file size exceeds available disk space?</b>	<p>Select <b>Yes</b> to raise an event if the maximum SQL file size (<code>MAXSIZE</code>) is set to a value greater than the amount of available disk space. The default is No.</p> <p>The <code>MAXSIZE</code> value identifies the maximum size to which a SQL Server database can grow.</p>
Event severity when SQL maximum file size exceeds disk space	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the maximum SQL file size is set to a value greater than the amount of available disk space. The default is 15.
<b>CCDB processing threads executed by CQS Service Monitoring</b>	
<b>Raise an event if threads executed by CQS have critical errors?</b>	<p>Select <b>Yes</b> to raise an event if the CCDB processing threads that are executed by CQS have critical SQL Server Database Engine errors. Critical errors indicate system problems and fatal errors, which can affect all processes accessing data in the same database. Critical errors fall under the severity level 20-24 of the Database Engine Error Severities.</p> <p>For information on Database Engine Errors and Severities, see the <a href="#">Microsoft article</a>.</p>
Event severity when critical errors occur in threads executed by CQS service	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the errors in the threads are of critical severity. The default is 5.

Parameter	How to Set It
<b>Raise an event if threads executed by CQS have major errors?</b>	<p>Select <b>Yes</b> to raise an event if the CCDB processing threads that are executed by CQS have major SQL Server Database Engine errors. Major errors include deadlock errors, security- related errors, syntax errors, and software errors. Major errors fall under the severity level 11-19 of the Database Engine Error Severities.</p> <p>For information on Database Engine Errors and Severities, see the <a href="#">Microsoft article</a>.</p>
Event severity when major errors occur in threads executed by CQS service	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the errors in the threads are of major severity. The default is 10.
<b>Raise an event if threads executed by CQS have minor errors?</b>	<p>Select <b>Yes</b> to raise an event if the CCDB processing threads that are executed by CQS have minor SQL Server Database Engine errors. Minor errors include informational messages. Minor errors fall under the severity level 0-9 of the Database Engine Error Severities.</p> <p>For information on Database Engine Errors and Severities, see the <a href="#">Microsoft article</a>.</p>
Event severity when minor errors occur in threads executed by CQS service	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the errors in the threads are of minor severity. The default is 25.
<b>Data Collection</b>	
Collect data for database space utilization?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns the percentage of database space used on the Control Center server. The default is No.
Collect data for log space utilization?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns the percentage of log space used on the Control Center server. The default is No.
Collect data for Command Queue Server connection status?	<p>Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns <b>100</b> if CQS is up and <b>0</b> if CQS is down. The default is No.</p> <p><b>NOTE:</b> This script only collects data for the CQS connection status when you run the script on the Control Center database (NQCCDB). The script does <i>not</i> collect data when you run it on the CQS.</p>
Collect data for SQL Server service status?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns <b>100</b> if SQL Server services are up and <b>0</b> if SQL Server services are down. The default is No.
<b>Monitoring</b>	
SQL username	<p>Specify the username required to access SQL Server on the Control Center server. Ensure that the user has sufficient permission to access SQL Server on the Control Center server.</p> <p>For more information, see the “Configuring Security Manager for AM Health” section in <i>AppManager for Self Monitoring Management Guide</i>.</p>

## 3.3 HeartbeatUNIX

Use this Knowledge Script to monitor the heartbeat of the AppManager agent running on a UNIX or Linux server. A **heartbeat** is a periodic signal generated by an AppManager agent computer to indicate that it is still running. If an AppManager agent fails to send either data or an event to the QDB within the specified grace period, this script considers the agent to be offline.

This Knowledge Script also monitors the health of jobs running on UNIX and Linux agents. Use this script to monitor Knowledge Script job run time against job schedule and against a user-specified maximum run time.

---

**NOTE:** Job monitoring is only available for UNIX agent 8.0 and above.

---

You can set this script to raise an event for the following conditions:

- ◆ Agent Heartbeat
  - The heartbeat fails
  - An agent is healthy, such as when the heartbeat returns after failing
  - The agent heartbeat fails a user-specified number of times
- ◆ Job Monitoring (UNIX agent 8.0 and above)
  - A job takes longer than its schedule to execute
  - A job exceeds the user-specified maximum job run time
- ◆ Knowledge Script
  - An unexpected error occurs in this Knowledge Script.
- ◆ Heartbeat Investigation
  - An attempt to contact an offline agent computer using ICMP ping and the `traceroute` diagnostic tool both fail

If you use this Knowledge Script with the AppManager Operator Console, you can access the Actions and Advanced tab, but the options on those two tabs will not function.

This script generates data for consolidated events, which can be managed with the Health Check options found in Control Center. To access Health Check, click **Options** on the **Main** tab, and then click **Health Check**. For more information, see the online Help for Health Check.

### 3.3.1 Resource Objects

UNIX and Linux servers

### 3.3.2 Default Schedule

The default interval for this script is **every five minutes**.

### 3.3.3 Setting Parameter Values

Set the following parameters as needed:

---

Parameter	How to Set It
<b>Heartbeat Options</b>	

---

<b>Parameter</b>	<b>How to Set It</b>
<b>Raise event if the agent heartbeat fails?</b>	Select <b>Yes</b> to raise an event if the heartbeat for the AppManager agent server stops. The default is Yes.
Event severity when the agent heartbeat fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the heartbeat stops. The default is 5.
<b>Raise an event when agent heartbeat restarts?</b>	Select <b>Yes</b> to raise an event if the heartbeat starts again after stopping. The default is Yes.
Event severity when the agent heartbeat restarts	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the heartbeat starts again after stopping. The default is 25.
Number of consecutive heartbeat failures before raising an event	Specify the number of times the heartbeat must fail before raising an event. The default is 1.
Generate heartbeat data?	Select <b>Yes</b> to enable the heartbeat check. If you select Yes and the data point from this job is missing, AppManager raises an event. If you select No, the heartbeat check will not look for the data point from this job. The default is Yes.
<b>Job Monitoring Options</b>	
Monitor individual jobs?	Select <b>Yes</b> to monitor individual AppManager jobs. The default is Yes.
Raise an event if jobs take longer than their schedule to execute?	Select <b>Yes</b> to raise an event when a monitored AppManager job takes longer than its scheduled time to execute. The default is Yes.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a monitored Appmanager job took longer than its scheduled time to execute. The default is 5.
Raise an event if job exceeds maximum job run time?	Select <b>Yes</b> to raise an event when a monitored AppManager job exceeds the maximum job run time you set. The default is Yes.
List of Knowledge Scripts to skip (comma-separated)	Enter one or more comma-separated Knowledge Script names to exempt from job monitoring.
Maximum job run time	Set the number of seconds, from 1 to 32767, to indicate the maximum run time for a monitored Knowledge Script job before an event is raised. The default is 180 seconds.
Event severity	Set the event severity level, from 1 to 40, to indicate the importance of an event where a monitored Knowledge Script job exceeded the maximum job run time you set. The default is 5.
<b>Knowledge Script Options</b>	
Event severity when unexpected error occurs	Set the event severity level, from 1 to 40, to indicate the importance of an event where an unexpected error occurs in the HeartbeatUNIX Knowledge Script. The default is 35.
<b>Heartbeat Investigation Steps (Used by Management Server)</b>	
Attempt to contact agent computer by ICMP ping?	Select <b>Yes</b> to send an ICMP ping request to the agent computer. If AppManager cannot contact an agent with an ICMP ping, the agent computer might have been shut down or disconnected from the network, or a firewall is blocking the ICMP communication.  The default is Yes.



Parameter	How to Set It
Perform tracert diagnostic if ICMP ping fails?	Select <b>Yes</b> to run a tracert (traceroute) diagnostic test if the ping request fails. The default is Yes.  A traceroute test helps you troubleshoot network routing problems that can block ICMP traffic. This script raises an event if the tracert fails.

### 3.3.4 Understanding the Event Detail Messages

Many AppManager jobs run periodically as a series of iterations. The time that should elapse between subsequent iterations is the job schedule. The maximum amount of time the job should run in any of its iterations is the maximum job run time.

A job has a state, typically **Running** or **Stopped**. A job whose state is **Running** can have iteration status **Currently Running** when a job iteration is in progress, or **Completed** when a job is inactive between iterations.

This script can raise events for the following job run time conditions:

- When a job exceeds the *Maximum job run time* parameter you set in this Knowledge Script.
- When a job iteration takes longer than its schedule to execute.
- When a job iteration takes longer than its schedule time to execute and is still running.

The following are examples of the event detail messages this script creates when it detects these conditions in running jobs.

#### Job Exceeded Maximum Run Time

This Knowledge Script raises an event when one or more jobs exceeds the *Maximum job run time* parameter you set. The following is an example of the detail message.

Details for jobs exceeding maximum run time (180 seconds) at Sun Aug 3 20:33:06 2014 are :

```
Job ID 1269 (UNIX_ExecUtil)
  Iteration number : 759 (Completed)
  Iteration start time : Sun Aug 3 20:29:06 2014
  Iteration execution time : 3 minutes 33 seconds
Job ID 1271 (UNIX_ExecUtil)
  Iteration number : 380 (Currently Running)
  Iteration start time : Sun Aug 3 20:29:11 2014
  Iteration execution time : 3 minutes 55 seconds
  Last iteration run time : 4 minutes 13 seconds
Job ID 1273 (UNIX_ExecUtil)
  Iteration number : 758 (Completed)
  Iteration start time : Sun Aug 3 20:32:15 2014
  Iteration execution time : 4 minutes 28 seconds
```

The first line details the current *Maximum job run time* value and the time the report was generated. Each job that exceeds the maximum run time has its own entry in the detail message. Each entry starts with the agent-assigned job identifier and the Knowledge Script name.

The job entry details the iteration number, that is, how many iterations the job has run, and the current job iteration status. A job is *Completed* if it completed its iteration and is awaiting the next iteration. A job is *Currently Running* if the iteration is still in progress.



---

**NOTE:** Iteration status is separate from the job state. The job state is *Running* to indicate the job is either actively running or waiting to run. The iteration status is either *Completed* or *Currently Running* to indicate the current job iteration is either finished or is still active.

---

The job entry also details the iteration execution time and, in the case of a job currently running, the execution time for the previous iteration.

## Completed Job Exceeded Scheduled Run Time

This Knowledge Script raises an event when a job exceeds its scheduled run time. For example, if a job is scheduled to run every 30 seconds and in iteration  $n$  runs for 40 seconds, iteration  $n + 1$  cannot start at its scheduled time, it must start at least 10 seconds later. The following is an example of the detail message.

Details for jobs exceeding scheduled run time at Mon Aug 4 00:33:06 2014 are :

```
Job ID 1273 (UNIX_ExecUtil)
  Iteration number : 807 (Completed)
  Iteration start time : Mon Aug 4 00:32:02 2014
  Execution time : 6 minutes 56 seconds

  Next iteration schedule time : Mon Aug 4 00:37:02 2014
  Next Iteration schedule delayed by : 1 minutes 56 seconds
```

Note: Please increase the job schedule by at least 1 minutes 56 seconds to correct this problem

The first line details the time the report was generated. Each job that exceeded its scheduled run time has an entry in the detail message. Each entry starts with the agent-assigned job identifier and the Knowledge Script name.

The entry details the execution time and the schedule time for the next iteration. A note indicates how much you should add to the current schedule to bring the iterations back on schedule and prevent future schedule overruns.

## Running Job Exceeded Scheduled Run Time

This Knowledge Script raises an event when a job exceeds its schedule and is still running. The following is an example of the event detail message.

Details for jobs exceeding scheduled run time at Thu Aug 21 05:55:49 2014 are :

```
Job ID 679 (UNIX_ExecUtil)
  Iteration number : 4 (Currently Running)
  Iteration start time : Thu Aug 21 05:55:34 2014
  Next iteration Schedule time : Thu Aug 21 05:55:44 2014
  Next iteration schedule delayed by : 5 seconds
```

The first line details the time the report was generated. Each job that exceeded its scheduled run time has an entry in the detail message. Each entry starts with the agent-assigned job identifier and the Knowledge Script name.

The *Next iteration schedule delayed by* value is based on the time the event was raised. It represents the amount of time the next iteration would be delayed if the current iteration completed now. For example, Job ID 679 is still in process and there is no way to determine exactly when it will end. But based on overrun at the time of this event detail message, the next iteration will be delayed by at least 5 seconds.

## 3.4 HeartbeatWin

Use this Knowledge Script to test the heartbeat of the AppManager Windows agent computer. A **heartbeat** is a periodic signal generated by an AppManager agent computer to indicate that it is still running. If an AppManager agent fails to send either data or an event to the QDB within the specified grace period, the AMHealth\_HeartbeatWin job considers the agent to be offline.

This script raises events if the heartbeat for the agent computer stops or restarts, and it generates a data point about the heartbeat events. You can also use this script to track whether jobs finish in the expected time frame, or if they exceed the maximum run time.

You can set this script to raise an event for the following conditions:

- ◆ Heartbeat fails
- ◆ An agent is healthy, such as when the heartbeat returns after failing
- ◆ The agent heartbeat fails a user-specified number of times
- ◆ Jobs take longer than expected to execute
- ◆ Jobs exceed maximum run time
- ◆ No jobs are found

If an agent computer is offline, you can specify that the management server take additional steps to diagnose the level of non-connectivity that exists between the agent and the QDB.

If you use this Knowledge Script with the AppManager Operator Console, you can access the Actions and Advanced tab, but the options on those two tabs will not function.

This script generates data for consolidated events, which can be managed with the Health Check options found in Control Center. To access Health Check, click **Options** on the **Main** tab, and then click **Health Check**. For more information, see the online Help for Health Check.

### 3.4.1 Resource Objects

Windows servers

### 3.4.2 Default Schedule

The default interval for this script is **every five minutes**.

### 3.4.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
<b>General Settings</b>	
<b>Job Failure Notification</b>	
<b>Raise event if job fails unexpectedly?</b>	Select <b>Yes</b> to raise an event if the AMHealth_HeartbeatWin fails unexpectedly. The default is Yes.
Event severity when job fails unexpectedly	Set the event severity level, from 1 to 40, to indicate the importance of an event in which this Knowledge Script job fails or any other unexpected event occurs. The default is 5.

Parameter	How to Set It
<b>Additional Settings</b>	
<b>Event Details</b>	
Event detail format	Select the format in which you want to display the event detail. You can select from <i>HTML Table</i> or <i>Plain Text</i> . The default is HTML Table.
<b>Heartbeat Options</b>	
<b>Raise an event if the agent heartbeat fails?</b>	<p>Select <b>Yes</b> to raise an event for the selected agent if the agent is detected as having failed the heartbeat. The default is Yes.</p> <p>If several agents fail the heartbeat at the same time, the script raises a single event for those agents, and the event message lists all the offline agents. By default, you receive a single, consolidated event instead of multiple, individual events when 30% of the agents go offline. You can change this setting in Control Center by clicking <b>Options</b> on the <b>Main</b> tab, and then clicking <b>Health Check</b>.</p> <p>If you set the <i>Monitor individual jobs?</i> parameter in this script to Yes, you can control the severity of all heartbeat failure events using the <i>Event severity when the agent heartbeat fails</i> parameter instead of using the same severity for all consolidated events.</p>
Event severity when the agent heartbeat fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which an agent failed the heartbeat. The default is 5.
<b>Raise an event when agent heartbeat restarts?</b>	Select <b>Yes</b> to raise an event if the heartbeat starts again after stopping. The default is Yes.
Event severity when the agent heartbeat restarts	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the heartbeat starts again after stopping. The default is 25.
Number of consecutive heartbeat failures before raising an event	Specify the number of times the heartbeat must fail before raising an event. The default is 2.
Generate heartbeat data?	Select <b>Yes</b> to enable the heartbeat check. If you select Yes and the data point from this job is missing, AppManager raises an event. If you select No, the heartbeat check will not look for the data point from this job. The default is Yes.
<b>Job Monitoring Options</b>	
<b>Monitor individual jobs?</b>	Select <b>Yes</b> to monitor <i>all</i> jobs running on the agent that came from the same QDB as the heartbeat job. If you select No, the heartbeat job simply sends the heartbeat event and data according to the heartbeat-related parameters you set for this Knowledge Script. The default is Yes.
<b>Raise an event if jobs take longer than average to execute?</b>	Select <b>Yes</b> to raise an event that lists all jobs that are taking longer to execute than their average execution time. The agent stores a list of the average times jobs take to execute. The default is Yes.
Ignore jobs running for less than this amount of time	If you want to ignore jobs that are running for certain length of time, specify the running time for jobs that will be ignored. The default is 30 seconds.

Parameter	How to Set It
Grace period	<p>Specify a number to represent the grace period for job execution. The grace period is a multiple of the average time a job takes to execute. The agent stores a list of the average times jobs take to execute.</p> <p>For example, if you specified a grace period of 5, this script would take that value and multiply it by the average time a job takes to execute. If a job took one second to execute on average, the grace period would be 5 seconds. If the job takes longer than 5 seconds, the script raises an event.</p> <p>The default grace period is 5.</p>
Event severity when jobs take longer than average to execute	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the execution time for Knowledge Script jobs is longer than the average execution time for that job. The default is 5.
<b>Raise an event if jobs take longer than their schedule to execute?</b>	Select <b>Yes</b> to raise an event that lists all jobs that are taking longer to execute than their scheduled time to execute. The scheduled time is how often the job is set to run, such as every five minutes. The default is Yes.
Event severity when jobs take longer than their schedule to execute	Specify the event severity, from 1 to 40, to indicate the importance of an event in which the execution time for a Knowledge Script job is longer than the script's schedule. The default is 5.
<b>Raise an event if job exceeds maximum job run time?</b>	Select <b>Yes</b> to raise an event if the run time for a Knowledge Script job exceeds the <i>Maximum job run time</i> threshold. The default is Yes.
List of Knowledge Scripts to skip "Maximum job run time" check	Provide a comma-separated list of the Knowledge Scripts that you do not want to compare to the <i>Maximum job run time</i> threshold.
Maximum job run time	Specify the maximum number of seconds a Knowledge Script job can run before an event is raised. The default is 180 seconds.
Event severity when job exceeds maximum job run time	Set the event severity, from 1 to 40, to indicate the importance of an event in which the run time for a Knowledge Script job exceeds the <i>Maximum job run time</i> threshold. The default is 5.
<b>Raise an event if no jobs found?</b>	Select <b>Yes</b> to raise an event if no Knowledge Script jobs are running. The default is No.
Event severity when no jobs found	Set the event severity level, from 1 to 40, to indicate the importance of an event in which no Knowledge Script jobs are running. The default is 35.
Timeout when processing jobs	<p>Specify how long AppManager should wait for the agent to process jobs before assuming the agent either will not respond or has timed out. Use this parameter to monitor agents that are consistently taking longer than expected to respond.</p> <p>If the agent does not respond before your specified timeout value, AppManager raises an event stating that it was unable to process this command and suggesting you increase the timeout value. The event might also include data about a Windows error, if one was generated.</p> <p>The default timeout is 10 seconds.</p>
<b>Heartbeat Investigation Steps (Used by Management Server)</b>	AppManager performs the following steps only if the heartbeat event or the heartbeat data is missing.
Attempt to contact agent computer by ICMP ping?	Select <b>Yes</b> to send an ICMP ping request to the agent computer. If AppManager cannot contact an agent with an ICMP ping, the agent computer might have been shut down or disconnected from the network, or a firewall is blocking the ICMP communication.

Parameter	How to Set It
Perform tracert diagnostic if ICMP ping fails?	<p>Select <b>Yes</b> to run a tracert (traceroute) diagnostic test if the ping request fails. The default is Yes. A traceroute test helps you troubleshoot network routing problems that can block ICMP traffic.</p> <p>This script raises an event if the tracert fails.</p>
Connect to agent NetIQmc port if ICMP ping succeeds?	<p>Select <b>Yes</b> to attempt a connection to the NetIQmc port on the agent computer. The default is Yes. The connection is attempted only if the ping attempt succeeds.</p> <p>This script raises an event if the ping fails.</p>
Use RPC to probe agent if port check succeeds?	<p>Select <b>Yes</b> to send a Remote Procedure Call (RPC) to the agent computer. The default is Yes. The RPC is sent only if the port connection succeeds.</p> <p>This script raises an event if the RPC probe fails.</p>
Test agent computer registry if RPC probe succeeds?	<p>Select <b>Yes</b> to allow the management server to attempt to use the Remote Registry Service to connect to the Windows Registry on the agent computer.</p> <p>The connection is attempted only if the RPC probe succeeds. The management server must have sufficient privileges to connect to the Registry. The default is No.</p> <p>This script raises an event if the management server cannot connect to the Registry.</p>
Check status of agent services if registry test succeeds?	<p>Select <b>Yes</b> to allow the management server to verify whether the NetIQ agent services, NetIQccm and NetIQmc, are running. This test is attempted only if the registry test succeeds.</p> <p>The management server must have sufficient privileges to access the agent services. The default is No.</p> <p>This script raises an event if the agent services are up or down.</p>

## 3.5 QDBComponentsHealth

Use this Knowledge Script to monitor the health of NetIQ AppManager repository (QDB) and management server components.

This script monitors SQL Server resources associated with the QDB, including the percentage of database space and log space used, the time taken for a SQL command or query to execute, missing data tables and the status of AppManager scheduled tasks. If a service or job is down, this script can restart it.

This script monitors the data tables in the repository and raises an event if the previously created data tables and the future data tables are missing. An event is raised at the first occurrence of the missing data table and thereafter every 24 hours if the data tables are missing.

**NOTE:** The Knowledge Script raises an event automatically if the future data tables are missing in the repository.

This script monitors the data available in the Data Rejected table and if the data exceeds the threshold, an event is raised. The Data Rejected table contains data that are unable to link to its corresponding jobs because of the corrupted data headers or the corresponding jobs are missing.

If the QDB does not have an AppManager agent installed on it, Discovery\_AMHealth discovers the QDB components on the management server. As a result, the service and database monitoring parameters for this script run remotely. In this situation, you must have sufficient privileges on the service account for the NetIQMC service on the management server so that the service account can remotely access the SQL Server service on the QDB to obtain its status.

If the account does not have proper privileges, the script will be unable to access the service status and will report that the SQL Server service is down even when it is not. If you do not have sufficient access for the service account for the NetIQMC service, deselect the *Raise an event if SQL Server services are down* and *Restart SQL Server services that stop unexpectedly* parameters in this script to avoid raising unnecessary events.

You can set this script to raise an event for the following conditions:

- ◆ SQL Server services are down or have been restarted.
- ◆ SQL jobs are disabled, missing, or have failed.
- ◆ Database or log space is low.
- ◆ The management server isn't connected to the QDB database.
- ◆ The management server service is down.
- ◆ SQL Server queries against the QDB are taking too long to process.
- ◆ Database or log space is low, and there is insufficient disk space for further growth.
- ◆ Data in the Data Rejected table exceeds the specified threshold value.

When monitoring management server performance, QDBComponentsHealth does not use the standard method of creating events through the management server, because the problem being detected might prevent events from being generated through the management server. As a result, this script generates events for these conditions directly in the QDB, and Action scripts will not operate with certain QDBComponentsHealth parameters to generate actions when the conditions occur. Because these events are generated directly in the QDB, event collapsing is always enabled for these events, and it cannot be turned off.

The following QDBComponentsHealth parameters related to management server monitoring will not generate actions:

- ◆ Raise an event if the management server service is down?
- ◆ Raise an event if the management server service fails to restart?
- ◆ Raise an event if the management server service restarts successfully?
- ◆ Raise an event if the management server service is not connected to the QDB?
- ◆ Raise an event if data map file usage exceeds threshold?
- ◆ Raise an event if event map file usage exceeds threshold?
- ◆ Raise an event if job map file usage exceeds threshold?

If you do not have an agent installed on the QDB server, the repository component gets discovered on the management server. If you try to remotely monitor the QDB from the management server using the QDBComponentsHealth script, the script will not be able to obtain the disk information remotely.

As a result, if the repository component is monitored remotely from the management server by the QDBComponentsHealth script, the following QDB component monitoring parameters under the *SQL Server File Size and Growth Settings Monitoring* Event Notification Knowledge Script section will not be available:

- ◆ Raise an event if insufficient space available for further file growth?
- ◆ Raise an event if SQL Server maximum file size exceeds available disk space?

### 3.5.1 Resource Objects

- ◆ AppManager repository (QDB) server
- ◆ management server

### 3.5.2 Default Schedule

The default interval for this script is **every thirty minutes**.

### 3.5.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
<b>General Settings</b>	
<b>Job Failure Notification</b>	
<b>Raise event if job fails unexpectedly?</b>	Select <b>Yes</b> to raise an event if the AMHealth_QDBComponentsHealth fails unexpectedly. The default is Yes.
Event severity when job fails unexpectedly	Set the event severity level, from 1 to 40, to indicate the importance of an event in which this Knowledge Script job fails or any other unexpected event occurs. The default is 5.
<b>Additional Settings</b>	
<b>Event Details</b>	
Event detail format	Select the format in which you want to display the event detail. You can select from <i>HTML Table</i> or <i>Plain Text</i> . The default is HTML Table.
<b>Event Notification</b>	
<b>Raise an event if SQL Server services are down?</b> (Use MC based Action Script for notification)	Select <b>Yes</b> to raise an event if SQL Server services are down. The default is Yes. <b>Tip</b> Use Action Script for notification, as the QDB will not be available.
Event severity when a SQL Server service is down	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a SQL Server service is down. The default is 5.
<b>Restart SQL Server services if the services are stopped unexpectedly?</b>	Select <b>Yes</b> to restart the SQL Server services if the services stop unexpectedly. The default is Yes.

Parameter	How to Set It
<b>Raise an event if a SQL Server service restart fails?</b>	Select <b>Yes</b> to raise an event if a SQL Server service fails to restart. The default is Yes.  <b>Tip</b> Use Action Script for notification, as the QDB will not be available.
Event severity when a SQL Server service fails to restart	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a SQL Server service fails to restart. The default is 5.
<b>Raise an event if a SQL Server service restart succeeds?</b>	Select <b>Yes</b> to raise an event if a SQL Server service restart succeeds. The default is Yes.
Event severity when a SQL Server service restarts succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a SQL Server service restarts successfully. The default is 30.
<b>Raise an event if the Management Server service is not connected to the QDB?</b>	Select <b>Yes</b> to raise an event if the management server service is not connected to the QDB. The default is Yes.
Event severity when the Management Server service is not connected	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the management server service is not connected. The default is 5.
<b>Raise an event if the QDB SQL jobs are missing?</b>	Select <b>Yes</b> to raise an event if the QDB SQL jobs are missing. The default is Yes.
Event severity for missing QDB SQL jobs	Set the event severity level, from 1 to 40, to indicate the importance of an event in which QDB SQL jobs are missing. The default is 15.
<b>Raise an event if QDB SQL jobs are disabled?</b>	Select <b>Yes</b> to raise an event if QDB SQL jobs are disabled. The default is Yes.
Event severity for disabled QDB SQL jobs	Set the event severity level, from 1 to 40, to indicate the importance of an event in which QDB SQL jobs are disabled. The default is 15.
<b>Enable QDB SQL jobs if they are disabled?</b>	Select <b>Yes</b> to enable QDB SQL jobs if they are disabled. The default is No.
Event severity when QDB SQL jobs cannot be enabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the QDB SQL jobs cannot be enabled. The default is 15.
<b>Raise an event if attempt to enable QDB SQL jobs succeeds?</b>	Select <b>Yes</b> to raise an event if the attempt to enable the QDB SQL job succeeds. The default is No.
Event severity when attempt to enable QDB SQL job succeeds	Set the event severity level, from 1 to 40, to indicate the importance of an event in which an attempt to enable the QDB SQL job succeeds. The default is 20.
<b>Raise an event if a QDB SQL job fails?</b>	Select <b>Yes</b> to raise an event if a QDB SQL job fails. The default is Yes.
Event severity when a QDB SQL job fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a QDB SQL job fails. The default is 20.
<b>Raise an event if QDB disks are fragmented?</b>	Select <b>Yes</b> to raise an event if AppManager detects QDB disk fragmentation. The default is unselected.  <b>Tip</b> To avoid errors when running this script on a Windows Server 2008 server, disable User Account Control (UAC).



Parameter	How to Set It
Event severity for QDB disk fragmentation	Set the event severity level, from 1 to 40, to indicate the importance of an event in which a disk is fragmented. The default is 15.
<b>Raise an event if QDB file size exceeds threshold?</b>	Select <b>Yes</b> to raise an event if the QDB file size exceeds the threshold you set. The default is Yes.  <b>NOTE:</b> This parameter is applicable only on AppManager 9.1 running on SQL Server Express edition.
Threshold - Maximum QDB file size	Specify how large the QDB file size can grow before raising an event. If the QDB file size is too small, AppManager performance will not be optimal in larger environments. The default is 8 GBytes.
Event severity when QDB file size exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the QDB file size exceeds the threshold. The default is 20.
<b>Raise an event if the query process time exceeds threshold?</b>	Select <b>Yes</b> to raise an event if the query process time exceeds threshold. The default is Yes.
Threshold -- Maximum process run time	Specify the longest process run time allowed before an event is raised. The default is 300 seconds.
Event severity when query process time exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the query process time exceeds the threshold you set. The default is 5.
Event severity when attempt to retrieve query process time fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the attempt to retrieve query process time fails. The default is 10.
<b>Raise an event if unable to retrieve QDB component information?</b>	Select <b>Yes</b> to raise an event if AppManager is unable to retrieve QDB component information. The default is Yes.
Event severity when unable to connect to retrieve QDB component information	Set the event severity level, from 1 to 40, to indicate the importance of an event in which AppManager is unable to connect to retrieve QDB component information. The default is 10.
<b>Raise event if previously created data tables are missing in the repository?</b>	Select <b>Yes</b> to raise an event if the data tables that were created in the repository are missing. The default is No.
Event severity when previously created data tables are missing in the repository	Set the event severity level, from 1 to 40, to indicate the importance of an event in which data tables that were created in the repository are missing. The default is 15.
<b>Management Server Performance Monitoring</b>	
<b>Raise an event if the Management Server map file is not enabled?</b>	Select <b>Yes</b> to raise an event if the management server map file is not enabled. The default is Yes.
Threshold -- Current map file size	Specify the current map file size. If the map file size is too small, the management server performance will not be optimal in larger environments. The default is 5 MB.
Event severity if map file is not enabled	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the map file is not enabled. The default is 15.
<b>Raise an event if current data rate per minute exceeds threshold?</b>	Select <b>Yes</b> to raise an event if the current data rate per minute of the management server exceeds the threshold you set. The default is Yes.

<b>Parameter</b>	<b>How to Set It</b>
Threshold - Maximum current data rate per minute	Specify the maximum data rate that the management server can handle before raising an event. The default is 4000 data rate per minute.
Event severity when current data rate per minute exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the current data rate per minute exceeds the threshold. The default is 15.
<b>Raise an event if current event rate per minute exceeds threshold?</b>	Select <b>Yes</b> to raise an event if the current event rate per minute of the management server exceeds the threshold you set. The default is Yes.
Threshold - Maximum current event rate per minute	Specify the maximum event rate that the management server can handle before raising an event. The default is 2000 events per minute.
Event severity when current event rate per minute exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the current event rate per minute exceeds the threshold. The default is 15.
<b>Raise an event if data map file usage exceeds threshold?</b>	Select <b>Yes</b> to raise an event if the data map file usage exceeds the threshold you set. The default is Yes.
Threshold -- Maximum data map file usage	Specify the highest level of data map file usage that can occur before an event is raised. The default is 80%.
Event severity when data map file usage exceeds the threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the data map file usage exceeds the threshold you set. The default is 15.
<b>Raise an event if event map file usage exceeds threshold?</b>	Select <b>Yes</b> to raise an event if the event map file usage exceeds the threshold you set. The default is Yes.
Threshold -- Maximum event map file utilization	Specify the highest level of event map file utilization that can occur before an event is raised. The default is 80%.
Event severity when event map file utilization exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the event map file utilization exceeds the threshold you set. The default is 15.
<b>Raise an event if job map file usage exceeds threshold?</b>	Select <b>Yes</b> to raise an event if the job map file usage exceeds the threshold you set. The default is Yes.
Threshold -- Maximum job map file utilization	Specify the highest level of job map file utilization before an event is raised. The default is 80%.
Event severity when job map file usage exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the job map file usage exceeds the threshold you set. The default is 15.
<b>Monitoring SQL Server File Size and Growth Settings</b>	
<b>Raise an event if insufficient space available for further file growth?</b>	Select <b>Yes</b> to raise an event if there is not enough space for additional file growth on the SQL Server. The default is Yes.
Event severity for insufficient space	Set the event severity level, from 1 to 40, to indicate the importance of an event in which there is not enough space for additional file growth on the SQL Server. The default is 10.
<b>Raise an event if SQL Server file growth rate is lower than the threshold?</b>	Select <b>Yes</b> to raise an event if the SQL Server file growth rate is lower than the threshold you set. The default is Yes.
Threshold -- Minimum growth rate in MB	Specify the lowest possible growth rate for the SQL Server in MB. The default is 256 MB.

Parameter	How to Set It
Threshold -- Minimum growth rate as a percentage	Specify the lowest possible growth rate for the SQL Server as a percentage. The default is 9%.
Event severity when file growth rate is lower than the threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the file growth rate is lower than the threshold you set. The default is 10.
<b>Raise an event if Autogrowth is not enabled and file utilization exceeds the threshold?</b>	Select <b>Yes</b> to raise an event if Autogrowth is not enabled, and file utilization exceeds the threshold you set. The default is Yes.
Threshold -- Maximum file utilization with Autogrowth disabled	Specify the highest percentage of file utilization with Autogrowth disabled before an event is raised. The default is 90%.
Event severity if Autogrowth disabled and usage exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which usage exceeds the threshold and Autogrowth is disabled. The default is 10.
<b>Raise an event if SQL Server maximum file size exceeds available disk space?</b>	Select <b>Yes</b> to raise an event if the SQL Server maximum file size exceeds the available disk space. The default is Yes.
Event severity when SQL Server file size exceeds disk space	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the SQL Server file size exceeds the available disk space. The default is 15.
<b>Monitoring the Management Server Service</b>	
<b>Raise an event if the Management Server service is down?</b>	Select <b>Yes</b> to raise an event if the management server service is down. The default is Yes.  <b>NOTE:</b> AppManager cannot create an event for this if the management server is down.
Event severity when the Management Server service is down	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the management server service is down. The default is 10.
<b>Restart the Management Server service if the service is down?</b>	Select <b>Yes</b> if you want to restart the management server service if the service is down. The default is Yes.
<b>Raise an event if the Management Server service restarts successfully?</b>	Select <b>Yes</b> to raise an event if the management server service restarts successfully. The default is Yes.  <b>NOTE:</b> AppManager cannot create an event for this if the management server is down.
Event severity when the Management Server service restarts successfully	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the management server service restarts successfully. The default is 25.
<b>Raise an event if the Management Server service fails to restart?</b>	Select <b>Yes</b> to raise an event if the management server service fails to restart. The default is Yes.  <b>NOTE:</b> AppManager creates an event directly in the QDB if the management server fails to restart.
Event severity when the Management Server service fails to restart	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the management server service fails to restart. The default is 5.

Parameter	How to Set It
<b>Raise an event if agents are not FIPS-compliant?</b>	Select <b>Yes</b> to raise an event if the agents are not FIPS-compliant. The default is Yes.
Event severity when agents are not FIPS-compliant	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the agents are not FIPS-compliant. The default is 10.
<b>Raise an event if data in DataRejected table exceeds threshold?</b>	Select <b>Yes</b> to raise an event if the job entries in the Data Rejected table exceed the Threshold type that you have selected. The default is Yes.  <b>NOTE:</b> This parameter runs every 24 hours irrespective of the default interval of the Knowledge Script.
Threshold type	Select one of the following threshold types: <ul style="list-style-type: none"> <li>♦ <b>Time duration for a job in the table:</b> This threshold value corresponds to the maximum time period for which a job entry can exist in the Data Rejected table prior to the current iteration. When you select this threshold, you must specify the time in the <b>Threshold- Maximum time for which a job exists in the table</b> parameter.  For example: If you specify the threshold as 60 minutes and the current iteration starts at 10 a.m., an event is raised if any job entry exists in the Data Rejected table for more than 60 minutes (1 hour) prior to 10 a.m.</li> <li>♦ <b>Number of job entries:</b> This threshold value corresponds to the maximum number of job entries that can exist in the Data Rejected table. When you select this threshold, you must specify the count in the <b>Threshold- Maximum number of entries of any job</b> parameter.</li> </ul>
Threshold- Maximum time for which a job exists in the table	Specify how long a job entry can exist prior to the current iteration in the Data Rejected table before an event is raised. The default is 60 minutes.
Threshold- Maximum number of entries of any job	Specify the maximum number of job entries that can exist in the Data Rejected table before an event is raised. The default is 50 counts.
Event severity when data in DataRejected table exceeds threshold	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the job entries exceed the threshold type that you specified. The default is 10.
<b>Raise an event if job language is not matching with KS language?</b>	Select <b>Yes</b> to raise an event if job language does not match with KS language. The default is Yes.
Threshold - maximum number of mismatches between job language and KS language	Specify the maximum number of mismatches, from 0 to 255, that can exist before an event is raised. Default value is 0.
Event severity when job language is not matching with KS language	Set the event severity level, from 1 to 40, to indicate the importance of an event, in which there is a mismatch between the job language and the KS language. Default severity is 5.
<b>Data Collection</b>	
Collect data for database space utilization?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns the percentage of the database data file currently being used. The default is No.

Parameter	How to Set It
Collect data for log space utilization?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns the percentage of the database log file currently being used. The default is No.
Collect data for Management Server connection status?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns 100 if the management server is connected to the repository and 0 if the management server is not connected. The default is No.
Collect data for SQL Server service status?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns 100 if the SQL Server service is up and 0 if the service is down. The default is No.
Collect data for Management Server current data rate per minute?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns the current data rate of the Management Server. The default is No.
Collect data for Management Server current event rate per minute?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns the current event rate of the Management Server. The default is No.
Collect data for Management Server data map file utilization?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns the percentage of the management server data map file currently being utilized. The default is No.
Collect data for Management Server event map file utilization?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns the percentage of the management server event map file currently being utilized. The default is No.
Collect data for Management Server job map file utilization?	Select <b>Yes</b> to collect data for charts and reports. If enabled, data collection returns the percentage of the management server job map file currently being utilized. The default is No.
<b>Monitoring</b>	
SQL username	Specify the user name required to access SQL Server on the Control Center server. The SQL or Windows user must have at least Server Administrator rights in SQL Server.  For more information, see the “Configuring Security Manager for AM Health” section in <i>AppManager for Self Monitoring Management Guide</i> .

## 3.6 TaskSchedulerHealth

Use this Knowledge Script to monitor the health status of Task Scheduler. Task Scheduler is a Windows service that schedules SQL jobs operating on the QDB and NQCCDB. This script also monitors the connectivity between the Task Scheduler and the QDB/NQCCDB.

This script restarts the Task Scheduler service if the service is not running. You can specify this script to raise an event if the service is down, or if the restart fails or succeeds.

---

**NOTE:** Task Scheduler is available only on AppManager 9.1 and later.

---

### 3.6.1 Resource Objects

Task Scheduler

## 3.6.2 Default Schedule

The default interval for this script is **every five minutes**.

## 3.6.3 Setting Parameter Values

Set the following parameters as needed:

Parameter	How to Set It
<b>General Settings</b>	
<b>Job Failure Notification</b>	
<b>Raise event if job fails unexpectedly?</b>	Select <b>Yes</b> to raise an event if the AMHealth_TaskSchedulerHealth job fails unexpectedly. The default is Yes.
Event severity when job fails unexpectedly	Set the event severity level, from 1 to 40, to indicate the importance of an event that is raised when the job fails unexpectedly. The default is 5.
<b>Additional Settings</b>	
<b>Event Details</b>	
Event detail format	Select the format in which you want to display the event detail. You can select from <i>HTML Table</i> or <i>Plain Text</i> . The default is HTML Table.
<b>Monitor NetIQ TaskScheduler</b>	
Restart the Task Scheduler Service if the service is down?	Select <b>Yes</b> to restart the Task Scheduler service if the service is not running. The default is Yes.  <b>NOTE:</b> This Parameter setting does not restart the Task Scheduler service if the service is in the disable mode.
<b>Event Notification</b>	
<b>Raise an event if Task Scheduler service is down?</b>	Select <b>Yes</b> to raise an event if the Task Scheduler service is down. The default is Yes.
Event severity when Task Scheduler service is down	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the Task Scheduler service is down. The default is 5.
<b>Raise an event if Task Scheduler service restart fails?</b>	Select <b>Yes</b> to raise an event if the Task Scheduler service fails to restart. The default is Yes.
Event severity when Task Scheduler service restart fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the Task Scheduler service fails to restart. The default is 5.
<b>Raise an event if Task Scheduler service restart succeeds?</b>	Select <b>Yes</b> to raise an event if the Task Scheduler service restart succeeds. The default is No.
Event severity when a Task Scheduler service restarts successfully	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the Task Scheduler service restarts successfully. The default is 25.
<b>Raise an event if Task Scheduler connectivity to repository fails?</b>	Select <b>Yes</b> to raise an event if the Task Scheduler connection to the repository fails. The default is Yes.
Event severity when a Task Scheduler connectivity to repository fails	Set the event severity level, from 1 to 40, to indicate the importance of an event in which the Task Scheduler connectivity to the repository fails. The default is 5.

Parameter	How to Set It
<b>Raise an event if no repository exists in Task Scheduler?</b>	Select <b>Yes</b> to raise an event if you have not added any repository to Task Scheduler. The default is Yes.
Event severity when no repository exists in Task Scheduler	Set the event severity level, from 1 to 40, to indicate the importance of an event in which no repository exists in Task Scheduler. The default is 15.
<b>Data Collection</b>	
<b>Collect data for Task Scheduler service status?</b>	Select <b>Yes</b> to collect data for the Task Scheduler service status for charts and reports. The default is No.

## 3.7 Recommended Knowledge Script Groups

You can find the AMHealth Knowledge Script Groups (KSGs) on the RECOMMENDED tab of the Knowledge Script pane in the Operator Console.

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

The AMHealth KSGs enable a “best practices” usage for monitoring your AppManager environment. You can use these KSGs with AppManager monitoring policies. A monitoring policy lets you efficiently and consistently monitor all the resources in your environment using a set of preconfigured Knowledge Scripts. For more information, see “About Policy-Based Monitoring” in the AppManager Help.

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

In some cases, default script parameter settings are different when the script is deployed as part of a KSG, as opposed to 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 AMHealth KSG and want to restore it to its original form, you can reinstall the AppManager Self Monitoring module on the repository computer.

### 3.7.1 AMHealth\_HealthCheckAMAgentComponents Recommended KSG

The following Knowledge Scripts are members of the AMHealth\_HealthCheckAMAgentComponents recommended KSG:

- ◆ [HeartbeatUNIX](#)
- ◆ [HeartbeatWin](#)

This KSG is applied as a monitoring policy to the *Agent Managed Computers* management group. The monitoring policy creates an AMHealth\_HeartbeatWin job on Windows agents and an AMHealth\_HeartbeatUNIX job on UNIX agents in repositories running AppManager 8.x or later.

## 3.7.2 AMHealth\_HealthCheckAMCoreComponents Recommended KSG

The following Knowledge Scripts are members of the AMHealth\_HealthCheckAMCoreComponents recommended KSG:

- ◆ [CCComponentsHealth](#)
- ◆ [QDBComponentsHealth](#)
- ◆ [TaskSchedulerHealth](#)

This KSG is applied as a monitoring policy to the *Agent Managed Computers* management group.