Symantec™ ApplicationHA 6.2 Generic Agent Configuration Guide - Linux on KVM

November 2014

Symantec

Symantec™ ApplicationHA Generic Agent Configuration Guide

The software described in this book is furnished under a license agreement and may be used only in accordance with the terms of the agreement.

Agent version: 6.2

Document version: 6.2 Rev 1

Legal Notice

Copyright © 2014 Symantec Corporation. All rights reserved.

Symantec, the Symantec Logo, the Checkmark Logo, Veritas, Veritas Storage Foundation, CommandCentral, NetBackup, Enterprise Vault, and LiveUpdate are trademarks or registered trademarks of Symantec Corporation or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Symantec Corporation and its licensors, if any.

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. SYMANTEC CORPORATION SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be commercial computer software as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, "Rights in Commercial Computer Software or Commercial Computer Software Documentation", as applicable, and any successor regulations, whether delivered by Symantec as on premises or hosted services. Any use, modification, reproduction release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement.

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043

http://www.symantec.com

Technical Support

Symantec Technical Support maintains support centers globally. Technical Support's primary role is to respond to specific queries about product features and functionality. The Technical Support group also creates content for our online Knowledge Base. The Technical Support group works collaboratively with the other functional areas within Symantec to answer your questions in a timely fashion. For example, the Technical Support group works with Product Engineering and Symantec Security Response to provide alerting services and virus definition updates.

Symantec's support offerings include the following:

- A range of support options that give you the flexibility to select the right amount of service for any size organization
- Telephone and/or Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers software upgrades
- Global support purchased on a regional business hours or 24 hours a day, 7 days a week basis
- Premium service offerings that include Account Management Services

For information about Symantec's support offerings, you can visit our website at the following URL:

www.symantec.com/business/support/index.jsp

All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policy.

Contacting Technical Support

Customers with a current support agreement may access Technical Support information at the following URL:

www.symantec.com/business/support/contact_techsupp_static.jsp

Before contacting Technical Support, make sure you have satisfied the system requirements that are listed in your product documentation. Also, you should be at the computer on which the problem occurred, in case it is necessary to replicate the problem.

When you contact Technical Support, please have the following information available:

- Product release level
- Hardware information

- Available memory, disk space, and NIC information
- Operating system
- Version and patch level
- Network topology
- Router, gateway, and IP address information
- Problem description:
  - Error messages and log files
  - Troubleshooting that was performed before contacting Symantec
  - Recent software configuration changes and network changes

Licensing and registration

If your Symantec product requires registration or a license key, access our technical support Web page at the following URL:

www.symantec.com/business/support/

Customer service

Customer service information is available at the following URL:

www.symantec.com/business/support/

Customer Service is available to assist with non-technical questions, such as the following types of issues:

- Questions regarding product licensing or serialization
- Product registration updates, such as address or name changes
- General product information (features, language availability, local dealers)
- Latest information about product updates and upgrades
- Information about upgrade assurance and support contracts
- Information about the Symantec Buying Programs
- Advice about Symantec's technical support options
- Nontechnical presales questions
- Issues that are related to CD-ROMs or manuals

Support agreement resources

If you want to contact Symantec regarding an existing support agreement, please contact the support agreement administration team for your region as follows:

Asia-Pacific and Japan	customercare_apj@symantec.com
Europe, Middle-East, and Africa	semea@symantec.com
North America and Latin America	supportsolutions@symantec.com

About Symantec Connect

Symantec Connect is the peer-to-peer technical community site for Symantec's enterprise customers. Participants can connect and share information with other product users, including creating forum posts, articles, videos, downloads, blogs and suggesting ideas, as well as interact with Symantec product teams and Technical Support. Content is rated by the community, and members receive reward points for their contributions.

http://www.symantec.com/connect/storage-management

Documentation

Product guides are available on the media in PDF format. Make sure that you are using the current version of the documentation. The document version appears on page 2 of each guide. The latest product documentation is available on the Symantec website.

https://sort.symantec.com/documents

Your feedback on product documentation is important to us. Send suggestions for improvements and reports on errors or omissions. Include the title and document version (located on the second page), and chapter and section titles of the text on which you are reporting. Send feedback to:

doc_feedback@symantec.com

For information regarding the latest HOWTO articles, documentation updates, or to ask a question regarding product documentation, visit the Storage and Clustering Documentation forum on Symantec Connect.

https://www-secure.symantec.com/connect/storage-management/forums/storage-and-clustering-documentation

Contents

Technical Support 4

Chapter 1 Generic Application agent for Symantec ApplicationHA 8

About the generic agent 8
IMF awareness 9
Using Generic agent with IMF 9
Agent functions 10
State definitions 12

Chapter 2 Configuring the generic Application agent 13

About configuring application monitoring with ApplicationHA 13
Before configuring application monitoring for Application 14
Accessing the Symantec High Availability view 14
Configuring application monitoring for Application 15

Appendix A Resource type definitions 21

Resource type definition 21
Agent attributes 22

Appendix B Sample configurations 27

Sample configuration for init and custom processes 27

Appendix C Sample scripts for custom application 30

Sample scripts to start, stop, and monitor a custom application 30

Appendix D Custom monitor programs 32

Writing custom monitor programs to monitor multiple processes 32
Writing custom monitor programs to monitor multiple processes using PID files 33

Chapter 1

Generic Application agent for Symantec ApplicationHA

This chapter includes the following topics:

- About the generic agent

About the generic agent

The generic agent brings the custom applications online, takes them offline, and monitors their status. Use it to specify different executables for the online, offline, and monitor routines for different programs. The executables must exist locally on the virtual machine. You can use this agent to provide high availability for applications that are not by default supported by ApplicationHA.

For the list of supported applications, see *Symantec ApplicationHA Installation Guide*.

An application runs in the default context of root.

You can monitor the application in the following ways:

- Use the monitor program
- Specify a list of processes
- Specify a list of process ID files
- Any combination of the above

Note: ApplicationHA Custom Application wizard configures only the monitor program. For more details on how to specify the list of processes or process ID files: See "Writing custom monitor programs to monitor multiple processes using PID files" on page 33.

IMF awareness

The Application agent is Intelligent Monitoring Framework (IMF)-aware and uses asynchronous monitoring framework (AMF) kernel driver for IMF notification. If you use the Symantec ApplicationHA Configuration Wizard to configure the agent, the wizard by default enables IMF.

For more information on IMF and intelligent resource monitoring, refer to the *Symantec Cluster Server Administrator's Guide*.

For more information about IMF-related Application agent functions, see Agent functions.

Using Generic agent with IMF

Intelligent monitoring is supported for the Generic agent only under specific configurations. The complete list of such configurations is provided in the following table:

Table 1-1

MonitorProgram	MonitorProcesses	PidFiles	IMF Monitoring Mode
Not Configured	Not Configured	Not Configured	Not Applicable
Not Configured	Not Configured	Configured	Online, Offline
Not Configured	Configured	Not Configured	Online, Offline
Not Configured	Configured	Configured	Online, Offline
Configured	Not Configured	Not Configured	Offline Only
Configured	Not Configured	Configured	Online, Offline
Configured	Configured	Not Configured	Online, Offline
Configured	Configured	Configured	Online, Offline

Note: When you do not configure MonitorProcesses, IMF monitors only the StartProgram on the offline node. Hence, the MonitorFreq of IMF attribute must be set to 1 so that IMF monitors the resource on the offline node every monitor cycle.

When multiple processes are configured under the MonitorProcesses attribute and only some of them are running, offline registration with IMF fails repeatedly until RegisterRetryLimit is reached. In such a scenario, IMF cannot determine when the resource goes ONLINE and the agent monitors the resource in the traditional way.

Agent functions

Online	Runs the executable that you specify in the value of the StartProgram attribute. Runs the executable with the specified parameters in the context of the specified user. To bring the resource online, the agent function performs the command: `su [-] user -c executable_to_online_resource`
Offline	Runs the executable that you specify in the value of the StopProgram attribute. Runs the executable with the specified parameters in the context of the specified user. To take the resource offline, the agent function performs the command: `su [-] user -c executable_to_offline_resource`
Monitor	If you specify the MonitorProgram attribute, the agent executes the user defined MonitorProgram in the user-specified context. If you specify the PidFiles attribute, the routine verifies that the process ID that is found in each listed file is running. If you specify the MonitorProcesses attribute, the routine verifies that each listed process is running in the context of the user you specify. Use any combination among these attributes (MonitorProgram, PidFiles, or MonitorProcesses) to monitor the application. If any of the processes that are specified in either PidFiles or MonitorProcesses is determined not to be running, the monitor returns OFFLINE. If the process terminates ungracefully, the monitor returns OFFLINE and failover occurs. To monitor the resource, the agent function performs the command: `su [-] user -c executable_to_monitor_resource`
imf_init	Initializes the agent to interface with the asynchronous monitoring framework (AMF) kernel driver. This function runs when the agent starts up.
imf_getnotification	Gets notification about resource state changes. This function runs after the agent initializes with the AMF kernel driver. The agent continuously waits for notification and takes action on the resource upon notification.

imf_register	Registers the resource entities, which the agent must monitor, with the AMF kernel driver. For example, the function registers the PID for online monitoring of a process. This function runs for each resource after the resource goes into steady state (online or offline). The Application agent uses IMF for the processes configured with PidFiles and the MonitorProcesses attribute.
Clean	Terminates processes specified in PidFiles or MonitorProcesses. Ensures that only those processes (that are specified in the MonitorProcesses attribute) running with the user ID specified in the User attribute are killed. If the CleanProgram is defined, the agent executes the CleanProgram. To forcefully stop the resource, the agent function performs the command: `su [-] user -c executable_to_clean_resource` Note that the agent uses the `su -` option only when the attribute UseSUDash is enabled (1). The UseSUDash attribute is disabled (0) by default.
Action	The various functions of the action entry point are as follows: ■ program.vfd Checks the availability of the specified program and the execution permissions for the specified program. ■ user.vfd Checks the existence of the specified user on the host. ■ cksum.vfd Checks the existence of the same binary on all nodes. ■ propcv [For internal use only] Invokes the AMF call with arguments to decide whether to allow or prevent processes from starting for an application resource, outside the VCS control, in the cluster. The StartProgram and the processes configured under MonitorProcesses, registered with AMF for offline monitoring, are prevented from starting on the offline node. This helps prevent concurrency violation at an early stage. ■ getcksum Returns the checksum of the specified program

State definitions

ONLINE	Indicates that all processes that are specified in the PidFiles and the MonitorProcesses attribute are running and that the MonitorProgram returns ONLINE.
OFFLINE	Indicates that at least one process that are specified in the PidFiles attribute or MonitorProcesses is not running, or that the MonitorProgram returns OFFLINE.
UNKNOWN	Indicates an indeterminable application state or invalid configuration.

Chapter 2

Configuring the generic Application agent

This chapter includes the following topics:

- About configuring application monitoring with ApplicationHA
- Before configuring application monitoring for Application
- Accessing the Symantec High Availability view
- Configuring application monitoring for Application

About configuring application monitoring with ApplicationHA

This chapter describes the steps to configure application monitoring with ApplicationHA in a virtualization environment.

Consider the following points before you proceed:

- You configure an application for monitoring on a virtual machine using the Symantec ApplicationHA Configuration Wizard.
- The Symantec ApplicationHA Configuration Wizard is launched when you click **Configure Application Monitoring** in the Symantec High Availability view of the Veritas Operations Manager (VOM) Management Server console.
- In this release, the wizard allows you to configure monitoring for only one application per virtual machine.
  To configure another application using the wizard, you must first unconfigure the existing application monitoring.

- After you have configured monitoring for an application using the wizard, you can configure monitoring for other applications residing in the same virtual machine, using Symantec Cluster Server (VCS) commands.
  For more information read the following technote:
  http://www.symantec.com/docs/TECH159846
- After configuring Application for monitoring, if you create another instance of the application, these new components are not monitored as part of the existing configuration.
  In such a case, you must first unconfigure the existing configuration and then reconfigure the application using the wizard. You can then select all the instances for monitoring.

Before configuring application monitoring for Application

Ensure that you complete the following tasks before configuring application monitoring for Application on a virtual machine:

- Install Veritas Operations Manager (VOM) Management Server. For more information on working with VOM, see the *Symantec ApplicationHA User's Guide*. For information on accessing the Symantec High Availability view: See "Accessing the Symantec High Availability view " on page 14.
- Install ApplicationHA guest components on the virtual machine that you need to monitor.
- Assign ApplicationHA - Configure Application Monitoring (Admin) privileges to the logged-on user on the virtual machine where you want to configure application monitoring.
- Install the application and the associated components that you wish to monitor on the virtual machine.
- If you have configured a firewall, ensure that your firewall settings allow access to ports used by ApplicationHA installer, wizards, and services.
  Refer to the *Symantec ApplicationHA Installation Guide* for a list of ports and services used.

Accessing the Symantec High Availability view

To administer an application on a virtual machine that is running in the KVM environment, you must access the Symantec High Availability view of the Veritas Operations Manager (VOM) Management Server console.

From the Symantec High Availability view, you can perform administrative actions such as:

- Start an application
- Stop an application
- Configure application monitoring
- Unconfigure application monitoring
- Enable application heartbeat
- Disable application heartbeat
- Enter maintenance mode
- Exit maintenance mode

To access the Symantec High Availability view

1. Log on to the VOM Management Server console.
2. Select the Server perspective and expand Manage in the left pane.
3. Expand the Organization, or Uncategorized Hosts to navigate to the virtual machine.
4. Right-click the required virtual machine, and then click **Manage ApplicationHA**.

   The Symantec High Availability view appears.

Configuring application monitoring for Application

Perform the following steps to configure monitoring for Application on a virtual machine.

To configure application monitoring for Application

1. In the Symantec High Availability view of the Veritas Operations Manager Management Server Console, click **Configure Application Monitoring**.

   This launches the Symantec ApplicationHA Configuration Wizard.
2. Review the information on the Welcome screen and then click **Next**.

   The wizard lists all the supported applications for the system.

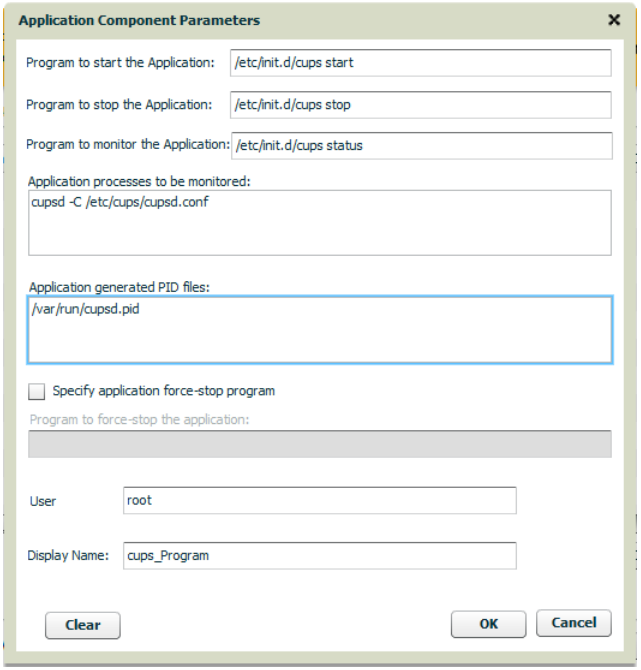
3 Select **Custom Application**, and then click **Next**.

Note: The wizard configures ApplicationHA to monitor custom applications with Intelligent Monitoring Framework (IMF).

The Program Selection screen appears.

4 To specify the application components to monitor, click **Add Component**.

The **Application Component Parameters** dialog box appears.



5 Specify the following details to configure the component for monitoring:

- **Start program**: The complete path of the start program script.
- **Stop program**: The complete path of the stop program script.
- **Force-stop program:**: The complete path of the program script to forcefully stop the application.
- At least one or more of the following:
  - **Monitor program**: The complete path of the monitor program script.

- **Application-related processes to monitor**: Names of the application processes that must be monitored.
- **Application-generated PID files**: Path names of the process ID (PID) files of your application.

- **User**: The user name. Ensure that you specify a valid user with adequate privileges on the virtual machine where you configure the application. Else, application monitoring may fail.

To remove a component from the Component box, use the Remove icon.

Note: The wizard automatically populates the **Display Name** for the specified component. You can, however, edit the information in this field. Also, ensure that you specify a user with valid credentials and adequate privileges on the virtual machine (guest) where you configure the generic agent. Else, application monitoring may fail.

---

6 Click **OK**.

The specified component appears on the Program Selection screen.

7 To specify more application components for monitoring, repeat step 4 to step 6. Else, click **Next**.

The other components you specified in steps 4 to 7 appear in the Components list.

The Define Start Stop Order screen appears. The screen lists the previously selected components.

8 In the Parent Component list, click a component.

9 To set up dependency with the specified parent component, in the Components list, click a component. Repeat this step for all parent components.

10 Click **Configure**.

The wizard performs the application monitoring configuration tasks. The ApplicationHA Configuration screen displays the status of each task.

After all the tasks are complete, click **Next**.

Note: If the configuration tasks fail, click **Diagnostic information**to check the details of the failure.

You then have to run the wizard again to configure the application monitoring.

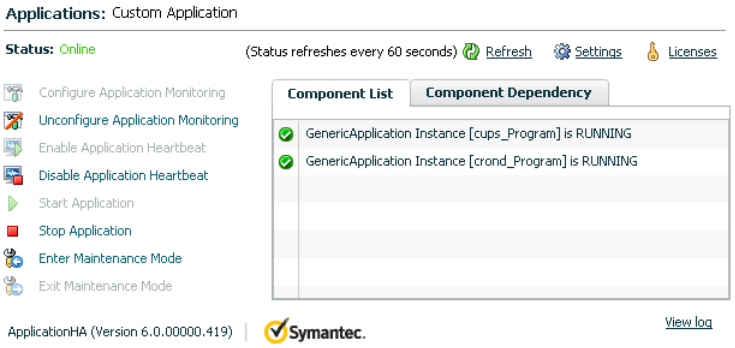
---

11 Click **Finish** to complete the wizard.

This completes the application monitoring configuration. See "Sample configuration for init and custom processes" on page 27.

12 To view the status of the configured application on a virtual machine, on the Veritas Operations Manager Management Server console, right-click the appropriate virtual machine and then click **Manage ApplicationHA**.

The Symantec High Availability view appears.

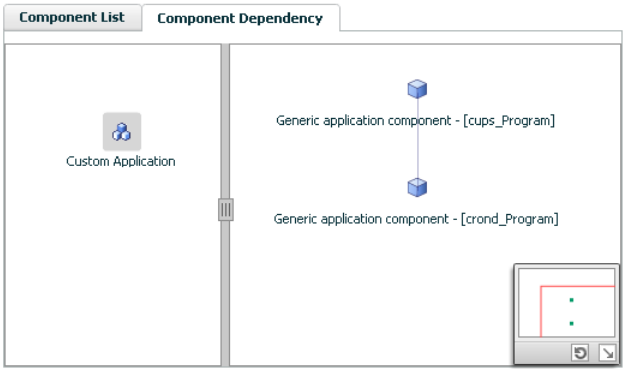


By default, the Component List tab appears. The tab lists each component of the configured application and the status description of each component.

For more information on viewing and administering applications through Veritas Operations Manager, see the *Symantec ApplicationHA User's Guide*.

13 To view component dependency for the monitored application, click the **Component Dependency** tab.

The component dependency graph appears.



The graph illustrates the dependencies between a selected component group (an application or a group of inter-related components) and its components for the configured application. The left pane displays component groups and/or configured applications. The right pane displays components of the selected component group or application.

For more information on viewing component dependency for any configured application, see the *Symantec ApplicationHA User's Guide*.

Appendix A

Resource type definitions

This appendix includes the following topics:

- Resource type definition
- Agent attributes

Resource type definition

```
type Application (
        static keylist SupportedActions = { "program.vfd", "user.vfd",
        "cksum.vfd", getcksum, propcv }
        static str ArgList[] = { User, StartProgram, StopProgram,
        CleanProgram, MonitorProgram, PidFiles, MonitorProcesses,
        EnvFile, UseSUDash, State, IState }
        static int IMF{} = { Mode = 3, MonitorFreq = 1,
        RegisterRetryLimit = 3 }
        static str IMFRegList[] = { MonitorProcesses, User, PidFiles,
        MonitorProgram, StartProgram, LevelTwoMonitorFreq }
        static int LevelTwoMonitorFreq = 1
        str User = root
        str StartProgram
        str StopProgram
        str CleanProgram
        str MonitorProgram
        str PidFiles[]
        str MonitorProcesses[]
        str EnvFile
        boolean UseSUDash = 0
)
```

Agent attributes

Table A-1 Required attributes

Required attribute	Description
StartProgram	The executable which starts the application. The attribute specifies the complete path of the executable. Applicable command line arguments follow the name of the executable. The executable path and argument have spaces separating them. For example, if the attribute for StartProgram is /usr/sbin/vxnotify -g dg00 -m >> /var/log/vxnotify.log (and vxnotify is blocking command) set it like: /usr/sbin/vxnotify -g dg00 -m >> /var/log/vxnotify.log & **Note:** The agent logs the return value of the StartProgram executable. The agent does not treat a non-zero return value as failure of execution and brings the resource online. **Note:** Do not use the opening and closing ({ }) brace symbols in this string. **Note:** In the script, specify a return value that is between 0 and 255. Type and dimension: string-scalar Example: "/usr/sbin/sample_app start"
StopProgram	The executable which stops the application. The Attribute specifies the complete path of the executable. Applicable command line arguments follow the name of the executable. **Note:** The agent logs the return value of the StopProgram executable. The agent does not treat a non-zero return value as failure of execution and takes the resource offline. **Note:** Do not use the opening and closing ({ }) brace symbols in this string. **Note:** In the script, specify a return value that is between 0 and 255. Type and dimension: string-scalar Example: "/usr/sbin/sample_app stop"

Table A-1 Required attributes *(continued)*

Required attribute	Description
At least one of the following attributes: ■ MonitorProcesses ■ MonitorProgram ■ PidFiles	See Table A-2 on page 23.

Table A-2 Optional attributes

Optional attribute	Description
CleanProgram	The executable which forcibly stops the application. The Attribute specifies the complete path of the executable. Applicable command line arguments follow the name of the executable. The executable path and argument have spaces separating them. **Note:** Symantec recommends to have the CleanProgram on the local storage so that in case of loss of storage connectivity ApplicationHA can take appropriate action to stop the application. **Note:** If the CleanProgram executable returns a non-zero value, the agent treats it as a clean failure and the resource will not fault. Type and dimension: string-scalar Example: "/usr/sbin/sample_app stop"
MonitorProcesses	A list of processes that you want monitored and cleaned. Each process name is the name of an executable. Qualify the executable name with its complete path if the path starts the executable.. The process name must be the name that the `ps -ef` command displays for the process. Type and dimension: string-vector Example: "nmbd"

Table A-2 Optional attributes *(continued)*

Optional attribute	Description
MonitorProgram	The executable which monitors the application. The Attribute specifies the complete path of the executable. Applicable command line arguments follow the name of the executable. The executable path and argument have spaces separating them. MonitorProgram can return the following values: OFFLINE value is 100 or 1; ONLINE values range from 101 to 110 or 0 (depending on the confidence level); 110 equals confidence level of 100%. Any other value = UNKNOWN. **Note:** Do not use the opening and closing ({ }) brace symbols in this string. If MonitorProgram is configured and not available, then resource state will be: ■ OFFLINE if the resource was in OFFLINE state and not waiting for any action. ■ UNKNOWN if the resource was in any other state or waiting for some action. Type and dimension: string-scalar Example: "/usr/sbin/sample_app_monitor all"
PidFiles	A list of PID (process ID) files that contain the PID of the processes that you want monitored and cleaned. These are application generated files. Each PID file contains one monitored PID. Specify the complete path of each PID file in the list. The process ID can change when the process restarts. If the application takes time to update the PID file, the agent's Monitor function may return an incorrect result. If incorrect results occur, increase the ToleranceLimit in the resource definition. Type and dimension: string-vector

Table A-2 Optional attributes *(continued)*

Optional attribute	Description
User	The user ID for running StartProgram, StopProgram, MonitorProgram, and CleanProgram. The processes that are specified in the MonitorProcesses list must run in the context of the specified user. Monitor checks the processes to make sure they run in this context. Type and dimension: string-scalar **Note:** If the configured user does not exist or if the home directory is not set for a configured user, the resource state will be UNKNOWN. Default: root Example: user1
EnvFile	The environment file that should get sourced before running any of the StartProgram, StopProgram, MonitorProgram or CleanProgram. Type and dimension: string-scalar Default: "" **Note:** Please make sure that the EnvFile adheres the default shell syntax of the configured user. Example: /home/username/envfile
UseSUDash	When the value of this attribute is 0, the agent performs an `su user` command before it executes the StartProgram, the StopProgram, the MonitorProgram, or the CleanProgram agent functions. When the value of this attribute is 1, the agent performs an `su - user` command before it executes the StartProgram, the StopProgram, the MonitorProgram or the CleanProgram agent functions. Type and dimension: boolean-scalar Default: 0 Example: 1

Table A-3 shows the internal attributes for the generic Application agent.

Table A-3 Internal attribute for the agent for generic applications

Internal attributes	Description
IMF	This resource-type level attribute determines whether the generic Application agent must perform intelligent resource monitoring. This attribute includes the following keys: ■ Mode: Define this attribute to enable or disable intelligent resource monitoring. Valid values are as follows: ■ 0—Does not perform intelligent resource monitoring ■ 1—Performs intelligent resource monitoring for offline resources and performs poll-based monitoring for online resources ■ 2—Performs intelligent resource monitoring for online resources and performs poll-based monitoring for offline resources ■ 3—Performs intelligent resource monitoring for both online and for offline resources Default: 3 ■ MonitorFreq: This key value specifies the frequency at which the agent invokes the monitor agent function. The value of this key is an integer. Default: 1 You can set this key to a non-zero value for cases where the agent requires to perform both poll-based and intelligent resource monitoring. If the value is 0, the agent does not perform poll-based process check monitoring. After the resource registers with the AMF kernel driver, the agent calls the monitor agent function as follows: ■ After every (MonitorFreq x MonitorInterval) number of seconds for online resources ■ After every (MonitorFreq x OfflineMonitorInterval) number of seconds for offline resources ■ RegisterRetryLimit: If you enable intelligent resource monitoring, the agent invokes the imf_register agent function to register the resource with theAMF kernel driver. The value of the RegisterRetryLimit key determines the number of times the agent must retry registration for a resource. If the agent cannot register the resource within the limit that is specified, then intelligent monitoring is disabled until the resource state changes or the value of the Mode key changes. Default: 3

Appendix B

Sample configurations

This appendix includes the following topics:

- Sample configuration for init and custom processes

Sample configuration for init and custom processes

This section describes steps to configure init processes, such as httpd and sendmail, as well as to configure custom processes for high availability by using Symantec ApplicationHA.

Let us assume that the sample custom application component, MyComponent1, can be started, stopped, forcibly stopped, and monitored by using the following scripts, respectively: `startMyComponent1`, `stopMyComponent1`, `forcestopMyComponent1`, and `monitorMyComponent1`. The `monitorMyComponent1` script is written to comply with the MonitorProgram attribute of Application.

See "Agent attributes" on page 22.

To configure application monitoring for an init process

1. In the Symantec High Availability view of theVeritas Operations Manager Management Server Console, click **Configure Application Monitoring**.

   This launches the Symantec ApplicationHA Configuration Wizard.

2. Review the information on the Welcome screen and then click **Next**.

   The wizard lists all the supported applications for the system.

3. Select **Custom Application**, and then click **Next**.

   The Program Selection screen appears.

4 To specify the httpd details to monitor, click **Add Component**.

The **Application Component Parameters** dialog box appears.

Note: init processes such as httpd, do not require special monitor scripts. ApplicationHA uses the status option of the init script for monitoring. However you can also use your own program scripts to monitor such processes.

---

5 Enter the following values in the respective fields and then click **OK**:

Program to start the application	`/etc/init.d/httpd start`
Program to stop the application	`/etc/init.d/httpd stop`
Program to monitor the application	`etc/init.d/httpd status`

Note: If you do not select the 'forcestop' option, then ApplicationHA uses the program script that you selected for stopping the application.

---

6 To specify the application components to monitor, click **Add Component**.

The Application Component Parameters dialog box appears.

7 Enter the following values in the respective fields:

Program to start the application	`/home/user1/myapplication/bin/startMyComponent1`
Program to stop the application	`/home/user1/myapplication/bin/stopMyComponent1`
Program to monitor the application	`/home/user1/myapplication/bin/monitorMyComponent1`
Program to forcibly stop the application	`/home/user1/myapplication/bin/forcestopMyComponent1`
User name to use when administering the application	*username* The default user name is ‘root’

8 Click **OK**.

The specified component appears on the Program Selection screen.

9 If you want to specify more application components for monitoring, repeat step 4 to step 8.

If you want to define the relationship between httpd and MyApplication components, click **Next**. The Define Start Stop Order screen appears. The screen lists the previously selected components.

10 To bring the httpd program online first and then the MyApplication program, in the Parent Component list, click on startMyComponent1_Program.

11 In the Component list, click to check httpd_Program and then click **Configure**.

The wizard performs the application monitoring configuration tasks. The ApplicationHA Configuration screen displays the status of each task. After all the tasks are complete, click **Next**.

12 Click **Finish** to complete the wizard. This completes the application monitoring configuration.

Appendix C

Sample scripts for custom application

This appendix includes the following topics:

- Sample scripts to start, stop, and monitor a custom application

Sample scripts to start, stop, and monitor a custom application

You can write your own scripts for the generic agent to bring a custom application online, take the custom application offline, and monitor the custom application's status. You can also modify the following sample scripts and use them, to start, stop, and monitor the custom application.

- Sample script to start a custom application:

```
#!/bin/sh
touch /tmp/sampleapp   # add any steps, if required
exit 0
```

  You can modify the sample start script to suit the custom application requirements. If you save the start script with the name `startsampleapp`, then to bring the custom application online, the agent function runs the following command:

  `su - root -c /root/customapp/startsampleapp`

- Sample script to stop a custom application:

```
#!/bin/sh
rm -f /tmp/sampleapp # add any steps, if required
exit 0
```

You can modify the sample stop script to suit the custom application requirements. If you save the stop script with the name `stopsampleapp`, then to bring down the custom application, the agent function runs the following command:

su - root -c /root/customapp/stopsampleapp

Note: The value of the return code for the start and stop scripts must be 0. No other return codes are supported.

- Sample script to monitor a custom application:

```
#!/bin/sh
APPLICATION_IS_ONLINE=110
APPLICATION_IS_OFFLINE=100
if [ -f /tmp/sampleapp ] ; then  # add any steps, if required
        exit $APPLICATION_IS_ONLINE
else
        exit $APPLICATION_IS_OFFLINE
fi
```

If you save the monitor script with the name `monitorsampleapp`, then to monitor the custom application, the agent function runs the following command:

su - root -c/root/customapp/monitorsampleapp

Appendix D

Custom monitor programs

This appendix includes the following topics:

- Writing custom monitor programs to monitor multiple processes
- Writing custom monitor programs to monitor multiple processes using PID files

Writing custom monitor programs to monitor multiple processes

The **Custom Application** option of the Application Monitoring Configuration Wizard does not allow you to monitor multiple processes. Perform the following steps to write a monitor program that can be used to monitor multiple processes using ApplicationHA.

To write a custom monitor program with process path names

1. Verify if each process is running by executing the following command:

   ```
   ps -ef | grep ProcessName
   ```

2. If all the processes are running, exit the monitor program with 110 as the return code.

   If any of the processes is not running, exit the monitor program with 100 as the return code.

3. Save this process in a shell script and assign execute permissions to the script.

4. In the **Application Component Parameters** dialog box of the Application Monitoring Configuration Wizard, enter the absolute path of the script in the **Monitor Program** field followed by the list of processes to monitor.

Writing custom monitor programs to monitor multiple processes using PID files

The **Custom Application** option of the Application Monitoring Configuration Wizard does not allow you to monitor multiple processes by using PID files. Perform the following steps to write a monitor program that can be used to monitor multiple processes using PID files in ApplicationHA.

To write a custom monitor program with PID files

1. Verify if process specified in each PID file is running by executing the following command:

   ```
   ps -ef | grep ProcessID
   ```

   Here, *ProcessID* is the content of the PID file.

2. If all the processes are running, exit the monitor program with 110 as the return code.

   If any of the processes is not running, exit the monitor program with 100 as the return code.

3. Save this process in a shell script and assign execute permissions to the script.

4. In the **Application Component Parameters** dialog box of the Application Monitoring Configuration Wizard, enter the absolute path of the script in the **Monitor Program** field followed by the list of PID files separated by a space.