# Symantec™ ApplicationHA Agent for Apache HTTP server Configuration Guide

Linux

5.1 Service Pack 2

July 1, 2011

Symantec

# Symantec™ ApplicationHA Agent for Apache HTTP server 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: 5.1.100.000

Document version: 5.1.100.000.0

## Legal Notice

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 Web site 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_apac@symantec.com |
| Europe, Middle-East, and Africa | semea@symantec.com |
| North America and Latin America | supportsolutions@symantec.com |

## 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 Web site.

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

## 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

# Contents

Chapter 1

# Introducing the Symantec ApplicationHA Agent for Apache HTTP server

This chapter includes the following topics:



## About the Symantec agent for Apache HTTP server

The Symantec ApplicationHA agents monitor specific components within an enterprise application. They determine the status of the application instances and start or stop them according to external events.

The Symantec ApplicationHA agent for Apache HTTP server provides high availability for Apache HTTP server instances.

## About installing and removing the ApplicationHA agent for Apache HTTP server

When you run the installer or uninstall program that accompanies the quarterly agent pack release of high availability agents from Symantec, the latest version of the ApplicationHA agent for Apache HTTP server is automatically installed or

removed. For more information, see the *Symantec ApplicationHA Agent Pack Installation Guide*.

# Supported software

The Symantec ApplicationHA agent for Apache HTTP server supports the following software versions:

| | |
|---|---|
| Symantec ApplicationHA | ■ 5.1 Service Pack 2 |
| VMware ESX and VMware ESXi | ■ 4.1<br>■ 4.1 U1 |
| Operating Systems | ■ Red Hat Enterprise Linux (RHEL) 5 with Update 3 (2.6.18-128.el5 kernel) or later; architecture: x86_64<br>■ SUSE Linux Enterprise Server (SLES) 11 Service Pack 1 (2.6.32.12-0.7 kernel) or later; architecture: x86_64<br>■ Oracle Enterprise Linux (OEL) 5 with Update 3 (2.6.18-128.0.0.0.1.el5) or later; architecture: x86_64 |
| Apache HTTP server | ■ 1.3, 2.0, and 2.2.<br>Also supports the IBM HTTP Server 7.x. |

# Apache HTTP server agent functions

The agent consists of resource type declarations and agent executables. The agent executables are organized into online, offline, monitor, and clean functions.

## Online

When you click **Start Application**, ApplicationHA performs the following Online tasks :

- Starts the Apache HTTP server by executing the `httpd` program with the appropriate arguments from the location specified using the httpdDir attribute.
- If the EnvFile attribute is configured, the file is sourced before the agent executes the `httpd` program.

## Offline

When you click **Stop Application**, ApplicationHA performs the following Offline tasks :

- Stops the Apache HTTP server by executing the `httpd` program with the appropriate arguments from the location specified using the httpdDir attribute.
- For Apache v1.3, sends a TERM signal to the HTTP Server parent process.
- If the EnvFile attribute is configured, the file is sourced before the agent executes the `httpd` program.

## Monitor

This function monitors the state of the Apache HTTP server instances running in a virtual machine, by performing the following tasks:

- Conducts a first level check, to ensure that all the processes of an Apache HTTP server instance is running.
- The processes of an Apache HTTP server instance is identified by applying the pattern matching on command lines of processes running in the virtual machine.
- Depending upon the value of the MonitorProgram attribute, the monitor function can perform an optional check on the Apache HTTP server instance by using the `ab` utility (Apache benchmarking utility).

---

**Note:** To configure second level monitoring, use CLI.

ApplicationHA wizards configure Apache HTTP server agent for basic or first level monitoring. To enable detailed or second level monitoring, use CLI/Veritas Operation Manager (VOM).

For more information on VCS commands, refer to Veritas Cluster Server documentation.

Also, for more information on detailed monitoring, See "Setting up detail monitoring for ApplicationHA agent for Apache HTTP server" on page 31.

---

## Clean

The clean function performs the following tasks:

- Removes the Apache HTTP server system resources that may remain after a server fault or after an unsuccessful attempt to start or stop the applciation. These resources include the parent Apache HTTP server processes and its child processes.

Chapter 2

# Configuring application monitoring with Symantec ApplicationHA

This chapter includes the following topics:



## About configuring application monitoring with ApplicationHA

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

Consider the following points before you proceed:

- You configure an application for monitoring on a virtual machine using the Application Monitoring Configuration Wizard.
- The Application Monitoring Configuration Wizard is launched when you click **Configure Application Monitoring** on the ApplicationHA tab of the VMware vSphere Client.
- 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 Veritas Cluster Server (VCS) commands.
  For more information read the following technote:
  http://www.symantec.com/docs/TECH159846
- After configuring Apache HTTP server for monitoring, if you create another Apache HTTP server instance, this new instance is 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 Apache HTTP server

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

- Install ApplicationHA Console.
- Install ApplicationHA guest components on the virtual machine that you need to monitor.
- Install VMware Tools on the virtual machine. Install a version that is compatible with VMware ESX server.
- If you do not have the latest version of ApplicationHA agent for Apache HTTP server installed, then run the agent pack installer on the virtual machine that you need to monitor.
- Install the VMware vSphere Client.
- 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 and Upgrade Guide* for a list of ports and services used.

# Configuring application monitoring for Apache HTTP server

Perform the following steps to configure monitoring for Apache HTTP server on a virtual machine hosted on a VMware vCenter Server managed ESX Server.

**To configure application monitoring for Apache HTTP server**

1. Launch the VMware vSphere Client and connect to the VMware vCenter Server that hosts the virtual machine.

   The vSphere Client is used to configure and control application monitoring.

2. From the vSphere Client's Inventory view in the left pane, select the virtual machine where you want to configure application monitoring for Apache HTTP server.

3. From the vSphere Client's Management view in the right pane, click the **ApplicationHA** tab.

   The ApplicationHA view displays the status of all the supported applications that are installed on the selected virtual machine.

4. In the ApplicationHA view, click **Configure Application Monitoring**.

   This launches the Application Monitoring Configuration Wizard.

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

   The wizard lists all the supported applications for the system.

6. Select Apache, and then click **Next**.

   The Apache HTTP server Instance Selection screen appears.

7. Enter the complete path of the Apache HTTP server binary file.

8. Enter the absolute path of the Apache HTTP server configuration file.

9. Click **Add**.

   The Apache HTTP server instance is added.

10. To add more Apache HTTP server instances, repeat steps 7 to 9.

11. Click **Configure**.

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

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

---

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

After you check the details of the failiure and resolve the issues, you have to run the wizard again to configure the application monitoring.

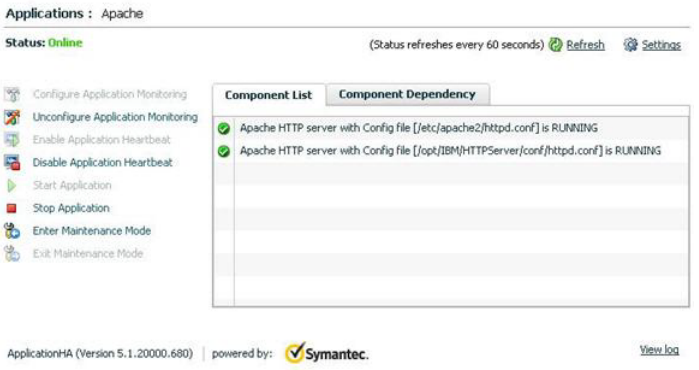
---

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

This completes the application monitoring configuration.

14 To view the status of the configured application on a virtual machine, in the inventory view of the vSphere Client, click the appropriate virtual machine, and then click the **ApplicationHA** tab.

The ApplicationHA view appears.

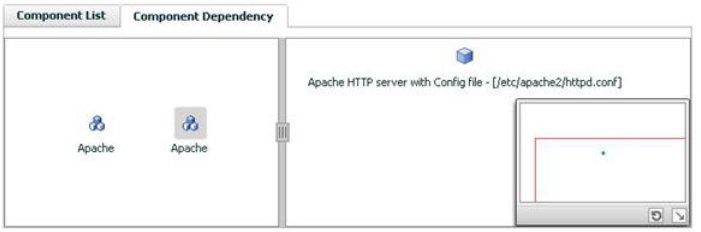



By default, the Component List tab appears. The tab lists each component of the configured application and the status description of each component.

For more information on viewing and administering applications by using the vSphere Client, see the *Symantec ApplicationHA User's Guide*.

15 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*.

Chapter 3

# Troubleshooting the agent for Apache HTTP server

This chapter includes the following topics:



## Starting the Apache HTTP server instance outside ApplicationHA control

If you face problems while working with an instance, you must disable the instance within the ApplicationHA framework. A disabled instance is not under the control of the ApplicationHA framework, and so you can test the Apache HTTP server instance independent of the ApplicationHA framework. Refer to the *Veritas Cluster Server Administrator's Guide* for information about disabling a resource.

You can then restart the Apache HTTP server instance outside the ApplicationHA framework.

---

**Note:** When you restart the instance outside the ApplicationHA framework, use the same parameters that the instance attributes define within the ApplicationHA framework.

---

When you bring an Apache HTTP server online outside of ApplicationHA control, first source its environment file. Start the server with the `-f` option so the server knows which instance to start.

A sample procedure to start an Apache HTTP server instance outside the ApplicationHA framework is illustrated as follows.

**To restart the Apache HTTP server outside the framework**

1 Log in as an Apache User.

```
# su ApacheUser
```

2 Start the Apache HTTP Server.

```
$ httpdDir/envvars; httpdDir/httpd -f ConfigFile
```

For example:

```
$ /apache/v2.2/bin/envars;/apache/v2.2/bin/httpd -f
/apache/v2.2/conf/httpd.conf -k start
```

If the Apache HTTP server works correctly outside the ApplicationHA framework, you can then attempt to implement the Apache HTTP server within the ApplicationHA framework.

# Monitoring Apache HTTP server processes

The agent for Apache HTTP server monitors all the processes similar to the following pattern:

| | |
|---|---|
| Apache HTTP server 2.0 | `/usr/sbin/httpd -f /etc/apache2/httpd.conf` |
| IBM HTTP Server | `/opt/IBM/HTTPServer/bin/httpd -f /opt/IBM/HTTPServer/conf/httpd.conf` |

# Stopping Apache HTTP server processes forcefully

When an attempt to gracefully stop the Apache HTTP server fails, the agent for Apache HTTP server kills all the processes similar to the following pattern:

| | |
|---|---|
| Apache HTTP server 2.0 | `/usr/sbin/httpd -f /etc/apache2/httpd.conf` |
| IBM HTTP Server | `/opt/IBM/HTTPServer/bin/httpd -f /opt/IBM/HTTPServer/conf/httpd.conf` |

# Reviewing error log files

If you face problems while using Apache HTTP server or the agent for Apache HTTP server, use the log files described in this section to investigate the problems.

## Using Apache HTTP server log files

If an Apache HTTP server faces problems, you can access the server log files to diagnose the problem. Typically the Apache HTTP server log files are located in the `/var/log/httpd` directory. Alternatively you can find the exact location of the log files, specified using the ErrorLog tag, in the Apache HTTP server configuration file.

## Reviewing ApplicationHA log files

If you face problems while using the agent for Apache HTTP server, you can also access the ApplicationHA engine, Apache HTTP server, and ApplicationHA log files for more information about a particular instance. The log files are located at the following location:

- The ApplicationHA engine log file is `/var/VRTSvcs/log/engine_A.log`
- Apache HTTP server agent log file is `/var/VRTSvcs/log/Apache_A.log`
- ApplicationHA log file is `/var/VRTSvcs/log/AppControlOperations_A.log`

## Using trace level logging

The ResLogLevel attribute controls the level of logging that is written in a log file for each Apache HTTP server instance. You can set this attribute to TRACE, which enables very detailed and verbose logging.

If you set ResLogLevel to TRACE, a very high volume of messages is produced. Symantec recommends that you localize the ResLogLevel attribute for a particular instance.

**To localize ResLogLevel attribute for a resource**

1. ApplicationHA commands reside in the `/opt/VRTS/bin` directory. Add this directory to your PATH environment variable. To set the path variable, perform the following step:

   ```
   # export PATH=$PATH:/opt/VRTS/bin
   ```

2. Identify the instance for which you want to enable detailed logging.
3. Localize the ResLogLevel attribute for the identified resource:

   ```
   # hares -local Resource_Name ResLogLevel
   ```

4. Set the ResLogLevel attribute to TRACE for the identified resource:

   ```
   # hares -modify Resource_Name ResLogLevel TRACE -sys SysA
   ```

5. Note the time before you begin to operate the identified resource.
6. Test the identified resource. The function reproduces the problem that you are attempting to diagnose.
7. Note the time when the problem is reproduced.
8. Set the ResLogLevel attribute back to INFO for the identified resource:

   ```
   # hares -modify Resource_Name ResLogLevel INFO -sys SysA
   ```

9. Save the configuration.

   ```
   # haconf -dump -makero
   ```

10. Review the contents of the log file. Use the time noted in Step 5 and Step 7 to diagnose the problem.

    You can also contact Symantec support for more help.

# Reconfiguring ApplicationHA when Apache HTTP server fails to start

This section describes the procedure to reconfigure application monitoring for Apache HTTP server.

**Perform the following steps to reconfigure application monitoring:**

1. In the ApplicationHA tab of the vSphere Client, click **Unconfigure Application Monitoring**. A confirmation box appears.
2. Click **OK**.
3. Click **Configure** and proceed with configuring application monitoring for Apache HTTP server. See "Configuring application monitoring for Apache HTTP server" on page 15.

Appendix A

# Resource type definitions

This appendix includes the following topics:



## About the resource type and attribute definitions

The resource type represents the configuration definition of the agent and specifies how the agent is defined in the configuration file. The attribute definitions describe the attributes associated with the agent. The required attributes describe the attributes that must be configured for the agent to function.

## Apache HTTP server agent attributes

**Table A-1** Required attributes

| Required attributes | Description |
|---|---|
| ConfigFile | Specifies the full path and name of the main configuration file for the Apache HTTP server.<br>Type and dimension: string-scalar<br>Example: "/apache/server1/conf/httpd.conf" |
| httpdDir | Specifies the full path of the directory to the httpd binary file<br>Type and dimension: string-scalar<br>Example: "/apache/server1/bin" |

**Table A-1** Required attributes *(continued)*

| Required attributes | Description |
|---|---|
| ResLogLevel | Controls the agent's logging detail for a specific instance of a resource. The values are:<br>■ ERROR: Logs error messages.<br>■ WARN: Logs error and warning messages.<br>■ INFO: Logs error, warning, and informational messages.<br>■ TRACE: Logs error, warning, informational, and trace messages. Trace logging is verbose. Use it for initial configuration or troubleshooting.<br>Type and dimension: boolean-scalar<br>Default: 0<br>Example: "TRACE" |
| EnvFile | This attribute may be required when you use IBM HTTP Server. |

**Table A-2** Optional attributes

| Optional attributes | Description |
|---|---|
| DirectiveAfter | Specifies a list of directives that the `httpd` program processes after reading the configuration file.<br>Type and dimension: string-association<br>Example: DirectiveAfter{} = { KeepAlive=On } |
| DirectiveBefore | Specifies a list of directives that the `httpd` program processes before it reads the configuration file.<br>Type and dimension: string-association<br>Example: DirectiveBefore{} = { User=nobody, Group=nobody } |
| User | Specifies the account name that the agent uses to execute the httpd program. If you do not specify this value, the agent executes httpd as the root user.<br>Type and dimension: string-scalar<br>Example: "apache1" |

**Table A-2** Optional attributes *(continued)*

| Optional attributes | Description |
|---|---|
| EnableSSL | If this attribute is set to 1 (true) the online agent function adds support for SSL, by including the option `-DSSL` in the start command.<br>For example: `/usr/sbin/httpd -f path_to_httpd.conf -k start -DSSL`<br>Where path_to_httpd.conf file is the path to the httpd.conf file.<br>If this attribute is set to 0 (false) the agent excludes the SSL support.<br>Type and dimension: boolean-scalar<br>Default: 0<br>Example: "1" |
| HostName | Specifies the virtual host name that is assigned to the Apache HTTP serverApache HTTP server instance. The host name is used in second-level monitoring for benchmarking the Apache HTTP server.<br>You can use IPv4 or IPv6 addresses for the HostName attribute.<br>**Note:** The HostName attribute is only required when the value of SecondLevelMonitor is 1 (true).<br>Type and dimension: string-scalar<br>Example: "web1.example.com" |
| Port | Specifies the port number where the Apache HTTP server instance listens. The port number is used in second-level monitoring for benchmarking the Apache HTTP server. Specify this attribute only if SecondLevelMonitor is set to 1 (true).<br>Type and dimension: integer-scalar<br>Default: 80<br>Example: "80" |

**Table A-2** Optional attributes *(continued)*

| Optional attributes | Description |
|---|---|
| EnvFile | Specifies the path and name of the file that is sourced before executing the `httpdDir/httpd` program. With Apache 2.0, the file *ServerRoot*/bin/envvars, which is supplied in most Apache 2.0 distributions, is commonly used to set the environment before executing httpd. Specifying this attribute is optional. If EnvFile is specified, the shell for user must be Bourne, Korn, or C shell.<br><br>This attribute may be required when you use the IBM HTTP Server if the online action fails. For example: Set the EnvFile to /usr/IBM/HTTPServer/bin/envvars.<br><br>Type and dimension: string-scalar<br><br>Example: "/apache/server1/bin/envvars" |
| PidFile | This attribute is for internal-use only. |
| SharedObjDir | Specifies the full path of the directory in which the Apache HTTP server shared object files are located. This attribute is used when the HTTP Server is compiled using the SHARED_CORE rule. If you specify this attribute, the directory is passed to the `-R` option when executing the httpd program. Refer to the httpd man pages for more information about the `-R` option.<br><br>Type and dimension: boolean-scalar<br><br>Example: "/apache/server1/libexec" |
| SecondLevelMonitor | Enables second-level monitoring for the resource. Second-level monitoring is a deeper, more thorough state check of the Apache HTTP server. The valid attribute values are 1 (true) and 0 (false).<br><br>Type and dimension: boolean-scalar<br><br>Default: 0<br><br>Example: "1" |

**Table A-2** Optional attributes *(continued)*

| Optional attributes | Description |
|---|---|
| SecondLevelTimeout | Specifies the number of seconds that the monitor agent function waits on the execution of the second-level monitor. If the second-level monitor program does not return to calling the monitor agent function before the SecondLevelTimeout window expires, the monitor agent function no longer blocks on the program sub-process. It does, however, report that the resource is offline. The value should be high enough to allow the second level monitor enough time to complete. The value should be less than the value of the agent's MonitorTimeout.<br>Type and dimension: integer-scalar<br>Default: 30 |

Appendix B

# Detail monitoring

This appendix includes the following topics:



## Setting the PATH variable

ApplicationHA commands reside in the `/opt/VRTS/bin` directory. Add this directory to your PATH environment variable.

**To set the PATH variable**

- Perform one of the following steps:

  For the Bourne Shell (sh or ksh), type:

  ```
  # PATH=/opt/VRTS/bin:$PATH; export PATH
  ```

  For the C Shell (csh or tcsh), type:

  ```
  # setenv PATH :/opt/VRTS/bin:$PATH
  ```

## Setting up detail monitoring for ApplicationHA agent for Apache HTTP server

This section describes the procedure to enable and disable detail monitoring for Apache HTTP server.

### To enable detail monitoring for Apache HTTP server

1 Make the ApplicationHA configuration writable:

```
# haconf -makerw
```

2 Freeze the service group to avoid automated actions by ApplicationHA in case of an incomplete configuration:

```
# hagrp -freeze Apache_SG_1
```

3 Enable detail monitoring for an Apache HTTP server instance by using the following ApplicationHA commands:

```
# hares -modify Apache_res_1 SecondLevelMonitor frequency

# hares -modify Apache_res_1 HostName hostname

# hares -modify Apache_res_1 Port port number
```

---

**Note:** For more information on SecondLevelMonitor attribute, See "Apache HTTP server agent attributes" on page 25.

---

4 Save the configuration and unfreeze the service group.

```
# haconf -dump -makero

# hagrp -unfreeze Apache_SG_1
```

### To disable detail monitoring for Apache HTTP server

1 Make the ApplicationHA configuration writable:

```
# haconf -makerw
```

2 Freeze the service group to avoid automated actions by ApplicationHA in case of an incomplete configuration:

```
# hagrp -freeze Apache_SG_1
```

3 Enable detail monitoring for an Apache HTTP server instance by using the following ApplicationHA commands:

```
# hares -modify Apache_res_1 SecondLevelMonitor 0
```

4 Save the configuration and unfreeze the service group.

```
# haconf -dump -makero

# hagrp -unfreeze Apache_SG_1
```

# Index