



**Symantec.**

Confidence in a connected world.

# Veritas Cluster Server: Steps to minimize application downtime with Intelligent Monitoring Framework (IMF)

*Anthony Herr*

*Storage and Availability Management Group*

*November 2011*

# Contents

Executive Summary .....	3
Introduction .....	3
Legacy Poll-Based Monitoring.....	3
Intelligent Monitoring.....	3
Feature Description.....	3
Technology Overview.....	4
Agent Support in VCS 5.1SP1 .....	4
Storage Foundation Based Agents.....	4
Process Based and Application Agents.....	4
The haimfconfig utility.....	4
Command Usage:.....	4
Enable IMF for all available IMF-aware agents in a cluster and enable the kernel module:....	4
Disable IMF for all IMF-aware agents on a cluster and disable the kernel module:.....	4
How to Enable IMF without the haimfconfig script.....	5
IMF Status Check.....	5
How to Disable IMF without the haimfconfig script.....	5
Recommended values for setting IMF Attributes .....	7
IMF Mode value settings: .....	8
Conclusion .....	8

## Executive Summary

Veritas Cluster Server provides immediate detection of application faults with the Intelligent Monitoring Framework (IMF) feature. This white paper provides a detailed overview of the IMF feature and its benefits in your data center.

## Introduction

All clustering applications have a single requirement in common: monitor the health of applications that they are protecting. There are two basic methodologies to achieve this: Legacy Monitoring and Intelligent Monitoring.

### Legacy Poll-Based Monitoring

Legacy monitoring is a poll based monitor and it has two major setbacks: delay in monitoring and inefficient use of CPU resources. There is an interval of time between polling the state of an application component. If the component faults, there will be a gap from the point the component has faulted until the next monitor is run. If a monitor is run every 60 seconds then there is a potential delay in detecting the fault for up to a minute. Also with a large number of monitored components or resources within a cluster, there will be some amount of CPU needed just to determine the state of the application on each node in the cluster. Even if the load for each check is small, combined, these loads can take up a measurable amount of CPU. Thus this can add up to a few percent of the CPU when clusters have hundreds of resources.

### Intelligent Monitoring

Intelligent Monitoring Framework, introduced in VCS 5.1SP1, addresses both issues seen with the Legacy Monitoring. By removing the constant monitoring cycle needed in legacy clustering products, there is no additional CPU overhead. In addition, this framework provides instant notification to VCS when a resource goes offline. As a result, action is taken immediately instead of any delay caused by the timing of the monitoring process. With IMF, as soon as an application faults, VCS is alerted and it immediately takes action.

NOTE: VCS for Windows also has the IMF feature introduced within this document. The Windows release of VCS implemented IMF in SFWHA 6.0.

## Feature Description

IMF is an extension to the VCS agent framework. The intelligent monitoring functionality of VCS agents is disabled by default during a fresh installation or an upgrade to VCS 5.1 SP1. You can enable or disable this functionality as needed. In future releases of VCS, this feature will be enabled by default. There is a downloadable script called *haimfconfig* that will assist with complete enable/disable of IMF for a cluster. Details on downloading the *haimfconfig* utility will be addressed in this document.

The benefits of intelligent monitoring over legacy poll-based monitoring are as follows:

- Significant reduction of failover or restart times, based on cluster configuration, due to faster notification of resource faults
- Reduction in VCS system utilization which enables VCS to effectively monitor a larger number of resources without load on the CPU

When an IMF-enabled agent starts up, the agent initializes the IMF notification module. After the resource is in a steady state, the agent registers with the IMF notification module the monitoring details of the resource. The IMF notification module hooks into system calls and other kernel interfaces of the operating system to get notifications on various events such as when a process starts or dies, or when a block device gets mounted or unmounted from a mount point.

The current IMF implementation is supported with process-based and mount-based agents. They use the Intelligent Monitoring Framework (IMF) kernel module that provides asynchronous event notifications to the agents that are enabled for intelligent resource monitoring.

In the event that the kernel module has an issue, IMF will monitor the resource using the legacy monitor. Also, when IMF determines that a resource has changed its state, based on notification from the kernel module, a traditional monitor is run to validate this state change.

## Technology Overview

### Agent Support in VCS 5.1SP1

Storage Foundation Based Agents	Process Based and Application Agents
Mount	Process
CFSMount	Application
CVMVxconfigd	Oracle
	Netlsnr
	DB2 (Introduced in VCS 5.1SP1RP2)

There is also support for the above agents running within a Solaris Zone or AIX WPAR

### The haimfconfig utility

When the IMF feature was introduced, it was disabled by default to allow customers to choose to enable it. You can still go through the steps outlined in a later section to manually enable or disable IMF or you can download the utility to enable or disable IMF from a single command. The *haimfconfig* command was created to assist users working with IMF. This utility will come as part of the binaries on future releases and it would be best to download it to your systems to the following location:

*/opt/VRTSvcs/bin/haimfconfig*

Online, the utility can be downloaded using the following link and is only valid on the 5.1SP1 release:

<https://www-secure.symantec.com/connect/downloads/intelligent-monitoring-framework-imf-configuration-utility-vcs-51sp1>.

### Command Usage:

**haimfconfig [-validate] {-enable|-disable} [-agent <agent>|-amf]**

**haimfconfig -enable** : This will set AMF\_START = 1, load kernel module and enable IMF for all possible agents

**haimfconfig -disable** : Disable IMF for all agent. This will unload the kernel module and set AMF\_START = 0

**haimfconfig -enable -agent <agents>** : Enable IMF for specified set of agent

**haimfconfig -disable -agent <agents>** : Disable IMF for specified set of agent

**haimfconfig -enable -amf** : This will set AMF\_START = 1 and load the kernel module

**haimfconfig -disable -amf** : This will unload the kernel module and set AMF\_START = 0

**-validate** : Displays a brief description of the steps performed by the script for specified command.

Enable IMF for all available IMF-aware agents in a cluster and enable the kernel module:

```
# haimfconfig -enable
```

Disable IMF for all IMF-aware agents on a cluster and disable the kernel module:

```
# haimfconfig -disable
```

## How to Enable IMF without the haimfconfig script

Use the following procedures to start IMF's kernel module, which is the kernel module IMF uses for kernel notification.

Run the first two commands on each node in the cluster:

1. Set the value of the AMF\_START variable to 1 in the following file:

```
# vi /etc/sysconfig/amf
```

2. Start the kernel module. Run the following command:

```
# /etc/init.d/amf start
```

3. Make the VCS configuration writable:

```
# haconf -makerw
```

To enable intelligent resource monitoring within VCS run steps 4 through 6 just once for the cluster per agent and then continue on with step 7 when completed with the steps after enabling IMF for the chosen agents.

4. Run the following command to enable intelligent resource monitoring. For more info on the Mode number, see the IMF [agent recommend values](#) section of this document:

```
# hatype -modify <Resource Type> IMF -update Mode <Mode Number>
```

5. Change the values of the MonitorFreq and the RegisterRetryLimit key of the IMF attribute:

```
# hatype -modify <Resource Type> IMF -update MonitorFreq <Frequency Value>
```

6. Change the value of the RegisterRetryLimit key of the IMF attribute if needed:

```
# hatype -modify <Resource Type> IMF -update RegisterRetryLimit <default is 3>
```

7. Save the VCS configuration:

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

8. Make sure that the kernel module is configured. Please run the following command on every node in the cluster:

```
# /etc/init.d/amf status
```

9. Restart the agent on each node in the cluster:

```
# haagent -stop <Agent Type> -force -sys <Cluster Node Name>
```

```
# haagent -start <Agent Type> -sys <Cluster Node Name>
```

## IMF Status Check

To check the current status of IMF you can run the command:

```
# /opt/VRTS/bin/amfstat
```

This will show the resources currently registered with the IMF kernel module.

Note: If you validate the IMF functionality by causing a monitored application to fault and restart, after each restart of the resource, the agent needs to reregister with the IMF kernel module. Based on the Agent framework there is also an allowance of time for the agent to go into a steady state. If you attempt to cause the agent to fault immediately after a restart occurs, it will take a full monitoring cycle to report the fault. Instead, if you wait a few minutes before attempting a second fault then the resource will be registered with IMF and fault detection will be immediate. You can run the *amfstat* command above to ensure that the agent has reregistered with the kernel module.

## How to Disable IMF without the haimfconfig script

Once IMF has been enabled, IMF will be persistently on. The following instructions are designed to turn off IMF for each Agent. To view which agents currently have active IMF registrations run the following command :

```
# /opt/VRTS/bin/amfstat
```

The following process can be used to disable IMF for an agent.

1. Run on each node in the cluster and for each Agent that had IMF enabled:

```
# haagent -stop <Agent> -force -sys <System>
```

2. After the agent(s) are stopped, stop the kernel module on each node in the cluster:

```
# /etc/init.d/amf stop
```

3. Make sure that the IMF kernel module is no longer configured on all nodes in the cluster. Run on each cluster node to ensure IMF was turned off for all agents:

```
# /etc/init.d/amf status
```

4. Run on each node and for each agent that had IMF enabled to restart the agent:  
*# haagent -start <Agent > -sys <System>*
5. Modify the startup file if IMF will be disabled for all Agents:  
Set the value of the AMF\_START variable to 0 in the following file on every node in the cluster to disable IMF from starting automatically upon boot:  
*# vi /etc/sysconfig/amf*
6. Make the VCS configuration writable:  
*# haconf -makerw*
7. Run the following command to disable intelligent resource monitoring for each resource type that has IMF enabled:  
*# hatype -modify <Agent > IMF -update Mode 0*
8. Save and close the VCS configuration:  
*# haconf -dump -makero*

## Recommended values for setting IMF Attributes

The values of the IMF attributes are per agent type and need to be set per agent. Recommended values for setting these attributes per agent are:

Mount	
Mode:	3 Online and Offline Monitoring
RegisterRetryLimit:	3 (default value)
MonitorFreq:	0
FSType attribute value is vxfs:	
For VxFS version 5.1 SP1	You can either set the MonitorFreq to 0 or a high value. Setting the value of the MonitorFreq key to a high value will ensure that the agent does not run the monitor function frequently. Setting the MonitorFreq key to 0 will disable the traditional monitoring while IMF monitoring is in progress. Traditional monitoring will be done only after receiving the notification for a resource.
For VxFS versions 5.1 or earlier	With VxFS versions prior to 5.1 SP1, VCS IMF only monitors file systems getting mounted and unmounted. To monitor other events, you must enable poll-based monitoring. Set the MonitorFreq key value to the frequency at which you want the agent to run the monitor agent function.
FSType attribute value is other than vxfs	VCS IMF monitors file systems getting mounted and unmounted. To monitor other events, you must enable poll-based monitoring. For the following cases, set the MonitorFreq key value to the frequency at which you want the agent to run the monitor agent function. <ul style="list-style-type: none"> <li>■ OptCheck attribute is not set to 0</li> <li>■ AccessPermissionChk is not set to 0</li> <li>■ SecondLevelMonitor (applicable only for nfs) is set to 1</li> </ul>

CFSMount	
Mode	3 Online and Offline Monitoring
RegisterRetryLimit	3 (default value)
MonitorFreq	0 or 1 – Similar considerations as the Mount Resource Type

Process	
Mode	3 Online and Offline Monitoring
RegisterRetryLimit	3 (default value)
MonitorFreq	0 IMF is good enough for process monitoring No need of traditional monitoring)

Application	
Mode	3 Online and Offline Monitoring
RegisterRetryLimit	3 (default value)
MonitorFreq	0 IMF is good enough for process monitoring No need of traditional monitoring)

Oracle	
Mode	3 Online and Offline Monitoring
RegisterRetryLimit	3 (default value)
MonitorFreq	0 IMF is good enough for process monitoring No need of traditional monitoring)

Netlsnr	
Mode	3 Online and Offline Monitoring
RegisterRetryLimit	3 (default value)
MonitorFreq	0 IMF is good enough for process monitoring No need of traditional monitoring)

CVMVxconfigd	
Mode	2 Online only Monitoring
RegisterRetryLimit	3 (default value)
MonitorFreq	0 IMF is good enough for process monitoring No need of traditional monitoring)

CFSFsckd	
Mode	3 Online and Offline Monitoring
RegisterRetryLimit	3 (default value)
MonitorFreq	0 IMF is good enough for process monitoring No need of traditional monitoring)

**IMF Mode value settings:**

- Mode value 0 has IMF disabled
- Mode value 1 is Offline only Monitoring
- Mode value 2 is Online only Monitoring
- Mode value 3 is both Online and Offline Monitoring

**Conclusion**

IMF was developed to take the CPU cost of application monitoring away. It provides an added benefit of instant notification with application monitoring to make a significant reduction of failover or restart times due to faster notification of resource faults. It is our recommendation that you implement this green technology to reduce your CPU load for your clustering product as well as reduce the amount of application downtime within your cluster.



## About Symantec

Symantec is a global leader in providing security, storage and systems management solutions to help businesses and consumers secure and manage their information.

Headquartered in Mountain View, Calif.,

Symantec has operations in 40 countries. More

information is available at

[www.symantec.com](http://www.symantec.com).

For specific country offices and contact numbers, please visit

our Web site. For product information in the U.S., call

toll-free 1 (800) 745 6054.

Symantec Corporation

World Headquarters

350 Ellis Street

Mountain View, CA 94043 USA

+1 (408) 517 8000

1 (800) 721 3934 ☐

[www.symantec.com](http://www.symantec.com)

Copyright © 2008 Symantec Corporation. All rights reserved. Symantec and the Symantec logo 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.

02/08