IRIS FailSafe™ Gauntlet™ Administrator's Guide

Document Number 007-3481-001

CONTRIBUTORS

Written by Susan Ellis
Illustrated by Dany Galgani
Edited by Christina Cary
Production by Michael Dixon
Engineering contributions by Chander Kant, Raghu Mallena

IRIS FailSafe™ Gauntlet™ Administrator’s Guide
Document Number 007-3481-001

Contents

List of Examples

List of Figures

About This Guide

This guide provides information about configuring IRIS FailSafe™ systems with the IRIS FailSafe Gauntlet™ option. This option enables the Gauntlet firewall to be failed over from one node to another if a component fails. This guide is intended as a supplement to the information about configuring IRIS FailSafe included in the *IRIS FailSafe Administrator's Guide*.

This guide was prepared in conjunction with the initial release of the IRIS FailSafe Gauntlet option. It describes IRIS FailSafe Gauntlet software for Gauntlet Release 3.2 running on IRIX™ 6.2 and IRIX 6.4.

Audience

This guide is written for system administrators who are responsible for configuring and administering an IRIS FailSafe system with the optional IRIS FailSafe Gauntlet software. These system administrators must be familiar with Gauntlet configuration.

Structure of This Document

This guide contains two chapters:

- Chapter 1, "Configuring IRIS FailSafe for Gauntlet," describes how to configure an IRIS FailSafe cluster with Gauntlet as a high-availability service.
- Chapter 2, "Configuration File Blocks for Gauntlet," contains reference information for the Gauntlet blocks in IRIS FailSafe configuration files.

Related Documentation

For Gauntlet administration information, see the *Gauntlet for IRIX Administrator's Guide*.

Besides this guide, other documentation for the IRIS FailSafe system includes

- *IRIS FailSafe Administrator's Guide*
- *IRIS FailSafe Programmer's Guide*

The IRIS FailSafe reference pages are

- ha_admin(1M)
- ha_appmon(1M)
- ha_cfgchksum(1M)
- ha_cfginfo(1M)
- ha_cfgverify(1M)
- ha_exec(1M)
- ha_hbeat(1M)
- ha_ifa(1M)
- ha_ifmx(1M) (IRIS FailSafe INFORMIX® option)
- ha_killd(1M)
- ha_nc(1M)
- ha_orcl(1M) (IRIS FailSafe Oracle® option)
- ha_spng(1M)
- ha_sybs(1M) (IRIS FailSafe Sybase® option)
- http_ping(1M) (IRIS FailSafe Web option)
- macconfig(1M)
- ha.conf(4)
- failsafe(7M)

Release notes are included with each IRIS FailSafe product. The names of the release notes are as follows:

ha_base	Release notes for IRIS FailSafe
ha_gauntlet	Release notes for IRIS FailSafe Gauntlet
ha_nfs	Release notes for IRIS FailSafe NFS®
ha_www	Release notes for IRIS FailSafe Web
ha_ orcl	Release notes for IRIS FailSafe Oracle
ha_ ifmx	Release notes for IRIS FailSafe INFORMIX
ha_ sybs	Release notes for IRIS FailSafe Sybase

Conventions Used in This Guide

These type conventions and symbols are used in this guide:

Bold — Literal command-line arguments and literal parameter values

Italics — Command names, filenames, new terms, the names of *inst* subsystems, manual/book titles, variable command-line arguments, and variables to be supplied by the user in examples, code, and syntax statements

`Fixed-width type`
Examples of command output that is displayed in windows on your monitor and of the contents of files

`Bold fixed-width type`
Commands and text that you are to type literally in response to shell and command prompts

# — IRIX shell prompt for the superuser (*root*)

Chapter 1

Configuring IRIS FailSafe for Gauntlet

This chapter provides information about how to configure and test an IRIS FailSafe cluster that is providing a Gauntlet firewall as a high-availability service. It assumes that you are familiar with the basic components of IRIS FailSafe described in Chapter 1 of the *IRIS FailSafe Administrator's Guide*.

The major sections in this chapter are as follows:

- "IRIS FailSafe Gauntlet Monitoring and Failover" on page 1
- "Planning IRIS FailSafe Gauntlet Configuration" on page 2
- "Installing Required Software" on page 4
- "Configuring Gauntlet and IRIS FailSafe Gauntlet" on page 5
- "Testing Gauntlet Failover" on page 9

IRIS FailSafe Gauntlet Monitoring and Failover

IRIS FailSafe provides high-availability for Gauntlet firewalls by monitoring them and quickly failing them over to the other node in the cluster if a failure is detected. The Gauntlet services that have been failed over experience a disconnection. It is the client's responsibility to handle the disconnection by trying to reconnect until the Gauntlet firewall (now on the other node) responds. See Chapter 1 of the *IRIS FailSafe Administrator's Guide* for more information about the interruption of services for clients.

The IRIS FailSafe Gauntlet option includes a script that performs monitoring of Gauntlet firewalls, */var/ha/actions/ha_gauntlet_lmon*.

Planning IRIS FailSafe Gauntlet Configuration

In an IRIS FailSafe Gauntlet cluster, both nodes run Gauntlet software at all times. However, only one node is providing firewall services. If a failure in the node providing firewall services occurs, IRIS FailSafe performs a failover and the other node begins providing all high-availability services. Thus, IRIS FailSafe Gauntlet operates in an active/backup configuration.

Shared disks are not used by IRIS FailSafe Gauntlet. All files required for Gauntlet operation are duplicated on local disks on each node. Gauntlet runs on both nodes in an IRIS FailSafe cluster concurrently. It should be configured similarly on both nodes. For example, if a trusted user is added on the primary node, it also needs to be added on the backup node.

A cluster providing Gauntlet as a high-availability service also provides IP address failover, but is not able to provide any other high-availability services. Thus, the system cannot be configured to provide volume, filesystem, NFS, Web, or database failover.

Figure 1-1 shows an example physical view of IRIS FailSafe Gauntlet configuration. In an IRIS FailSafe Gauntlet configuration, the IP addresses for the firewall internal and external hostnames are IP aliases. Networks on both sides of the firewall are aware of only the IP aliases for the network interfaces on their side of the firewall, not the fixed IP addresses. Clients see the configuration shown in Figure 1-2.

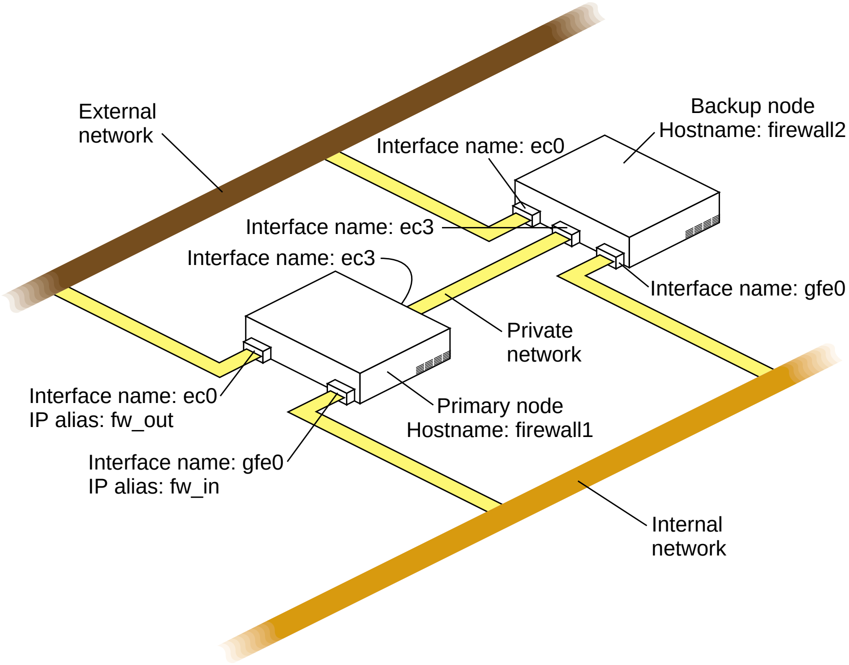

Figure 1-1 Example IRIS FailSafe Gauntlet Configuration

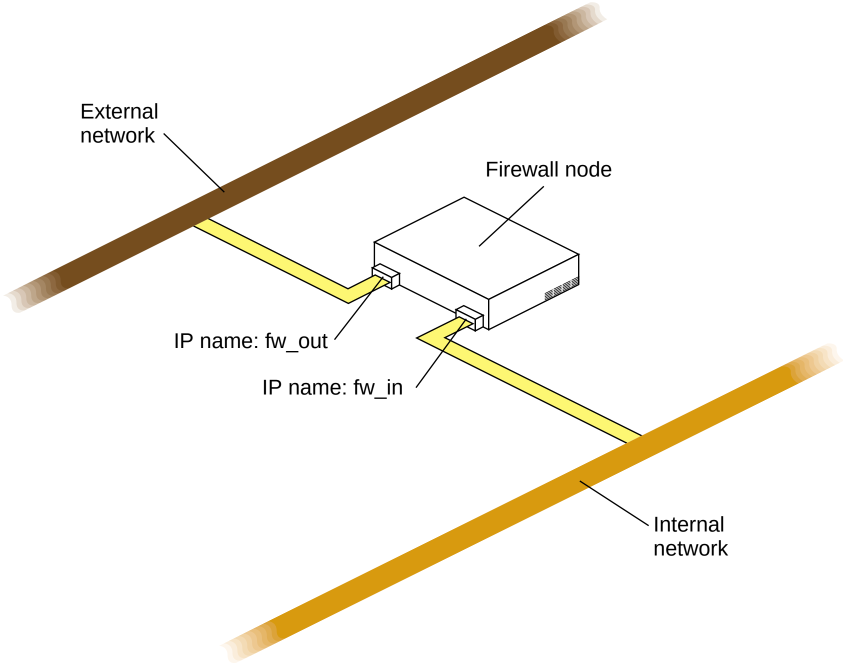

Figure 1-2 Client View of IRIS FailSafe Gauntlet Example

For information about choosing the IP aliases, see the section “Network Interface and IP Address Configuration” in Chapter 2 of the *IRIS FailSafe Administrator’s Guide*.

Installing Required Software

The required software for Gauntlet failover is as follows:

- Gauntlet software

  See the *Gauntlet for IRIX Administrator’s Guide* for information.

- base IRIS FailSafe software

  See the section “Installing the IRIS FailSafe Software” in Chapter 2 of the *IRIS FailSafe Administrator’s Guide* for a complete list of required base software.

- IRIS FailSafe Gauntlet software

  The software subsystem is *ha_gauntlet.sw.base*.

Note: Each node in the cluster requires a FLEX*lm*® license for Gauntlet.

Configuring Gauntlet and IRIS FailSafe Gauntlet

The step-by-step procedure for configuring Gauntlet and IRIS FailSafe Gauntlet on a cluster is given below. Before beginning this procedure, review this overview of the procedure:

- Configure and test IRIS FailSafe without Gauntlet.
- Without running IRIS FailSafe, configure the IP aliases up on the primary node.
- Configure Gauntlet on the primary node.
- Move the IP aliases from the primary node to the backup node.
- Configure Gauntlet on the backup node.
- Add Gauntlet information to the IRIS FailSafe configuration file */var/ha/ha.conf*.
- Start IRIS FailSafe on both nodes.

Follow these steps to configure Gauntlet and IRIS FailSafe Gauntlet on a cluster:

1. Perform the planning for IRIS FailSafe described in Chapter 2 of the *IRIS FailSafe Administrator's Guide*.
2. Perform the node configuration tasks described in Chapter 3 of the *IRIS FailSafe Administrator's Guide*.
3. If *portmap -a* is used on nodes in an IRIS FailSafe cluster rather than *rpcbind* to convert RPC program numbers into TCP or UDP protocol port numbers, you must add the private network address that is used by IRIS FailSafe on the other node to the file */etc/config/portmap.options* on each node.

   As an example, say that the host firewall1 uses the interface priv-firewall1 with IP address 192.0.5.1 for the private network and host firewall2 uses the interface priv-firewall2 with the IP address 192.0.5.2 for the private network.

   The file */etc/config/portmap.options* on firewall1 would include these two lines:

   ```
   -a 127.0.0.1
   -a 192.0.5.2
   ```

The file */etc/config/portmap.options* on firewall2 would include these two lines:

```
-a 127.0.0.1
-a 192.0.5.1
```

4. Prepare an IRIS FailSafe configuration file */var/ha/ha.conf* as described in the section “Creating a Configuration File” in Chapter 4 of the *IRIS FailSafe Administrator's Guide*. The template files you need in step 2 of that procedure are *ha.conf.system* and *ha.conf.interfaces*.
5. Append a copy of */var/ha/templates/ha.conf.gauntlet* to the end of the copy of the configuration file.
6. Complete the block called “application-class gauntlet.” See the section “Gauntlet Application-Class Block” in Chapter 2 and the comments in the file for information.
7. Complete the “gauntlet” block. See the section “Gauntlet Block” in Chapter 2 and the comments in the file for information.
8. Using information in the section “Gauntlet Action and Action-Timer Blocks” in Chapter 2, prepare the “action gauntlet” and “action-timer gauntlet” blocks.
9. Check the configuration file with the *ha_cfgverify* command:

   ```
   # /usr/etc/ha_cfgverify copy_filename
   ```

   See the section “Verifying the Configuration File” in Chapter 4 of the *IRIS FailSafe Administrator's Guide* for information about checking the configuration file with *ha_cfgverify*.
10. Comment out all the Gauntlet related blocks you prepared in steps 6 through 8 above. Do not comment out the blocks for interfaces used by Gauntlet.
11. Copy the configuration file to */var/ha/ha.conf* on each node.
12. Use the procedures in the sections “Testing the Serial Connection,” “Testing the Private Network,” and “Testing the Public Network Interfaces” in the *IRIS FailSafe Administrator's Guide* to test the basic IRIS FailSafe configuration.
13. Enter these commands on each node to start up IRIS FailSafe:

   ```
   # chkconfig failsafe on
   # /etc/init.d/failsafe start
   ```

14. Verify that each node is in normal state by giving this command on each node:

   ```
   # /usr/etc/ha_admin -i
   ha_admin: Node controller state normal
   ```

   If either node is not in normal state, wait 30 seconds and try the command again.

15. Enter this *chkconfig* command on each node to configure FailSafe off:

    ```
    # chkconfig failsafe off
    ```

16. Enter these commands on each node to shut down IRIS FailSafe:

    ```
    # chkconfig failsafe off
    # /etc/init.d/failsafe stop
    ```

    Wait for these commands to complete on the first node before entering them on the second node.

17. On the primary node, configure the IP aliases used by IRIS FailSafe by entering this command:

    ```
    # /var/ha/actions/takeback ‘ha_cfgchksum‘
    ```

18. On the primary node, list the configured interfaces. For example:

    ```
    # netstat -i
    Name Mtu   Network     Address            Ipkts Ierrs    Opkts Oerrs  Coll
    ec0  1500  192.0.4     firewall1.             2     0      138     0     0
                           fw_out.
    ec3  1500  192.0.5     priv-firewall1.   244063     9    11379     0  2062
    gfe0 1500  192.0.3     fast-firewall1.        7     0        3     0     0
                           fw_in.
    lo0  8304  loopback    localhost          60168     0    60168     0     0
    ```

    In the output, look for the network interfaces used for the internal and external networks. In this example output (for the example shown in Figure 1-1), ec0 has the IP alias fw_out and gfe0 has the IP alias fw_in.

19. Edit the file */usr/gauntlet/config/template.ipfilterd.conf* on each node and make these changes:

    - Uncomment the second *accept* command in the file.
    - Replace `ec3` in the accept command with the name of the network interface on that node that is used for the private network.

    The correct line looks like this:

    ```
    accept -i interface
    ```

    interface is the name of the network interface on that node that is used for the private network between the two nodes in the cluster. This line specifies that the network interface to the private network is a trusted interface.

20. On the primary node, configure Gauntlet using the procedure described in the *Gauntlet for IRIX Administrator's Guide*. During the configuration, you should not set the private network and the interface to the private network to be trusted or untrusted.

21. On the primary node, remove the IP aliases by entering this command:

```
# /var/ha/actions/giveaway ‘ha_cfgchksum‘
```

22. On the primary node, verify that the IP aliases are no longer configured by listing the configured interfaces. For example:

```
# netstat -i
Name Mtu   Network     Address            Ipkts Ierrs    Opkts Oerrs  Coll
ec0  1500  192.0.4     firewall1.             2     0      138     0     0
ec3  1500  192.0.5     priv-firewall1.   244063     9    11379     0  2062
gfe0 1500  192.0.3     fast-firewall1.        7     0        3     0     0
lo0  8304  loopback    localhost          60168     0    60168     0     0
```

Notice that fw_in and fw_out no longer appear in the output.

23. Reboot the primary node. For example:

```
# reboot
```

24. On the backup node, configure the IP aliases used by IRIS FailSafe by entering this command:

```
# /var/ha/actions/takeover ‘ha_cfgchksum‘
```

25. On the backup node, verify that the IP aliases are configured interfaces. For example:

```
# netstat -i
Name Mtu   Network     Address            Ipkts Ierrs    Opkts Oerrs  Coll
ec0  1500  192.0.4     firewall2.             2     0      138     0     0
                       fw_out.
ec3  1500  192.0.5     priv-firewall2.   244063     9    11379     0  2062
gfe0 1500  192.0.3     fast-firewall2.        7     0        3     0     0
                       fw_in.
lo0  8304  loopback    localhost          60168     0    60168     0     0
```

In the output, look for the network interfaces used for the internal and external networks. In this example output (for the example shown in Figure 1-1), ec0 has the IP alias fw_out and gfe0 has the IP alias fw_in.

26. On the backup node, configure Gauntlet using the procedure described in the *Gauntlet for IRIX Administrator's Guide*. The configuration of this node should be identical to the configuration of the primary node.

27. On the backup node, remove the IP aliases by entering this command:

    ```
    # /var/ha/actions/giveback ‘ha_cfgchksum‘
    ```

28. On the backup node, verify that the IP aliases are no longer configured by listing the configured interfaces. For example:

    ```
    # netstat -i
    Name Mtu   Network     Address            Ipkts Ierrs    Opkts Oerrs  Coll
    ec0  1500  192.0.4     firewall2.             2     0      138     0     0
    ec3  1500  192.0.5     priv-firewall2.   244063     9    11379     0  2062
    gfe0 1500  192.0.3     fast-firewall2.        7     0        3     0     0
    lo0  8304  loopback    localhost          60168     0    60168     0     0
    ```

    Notice that fw_in and fw_out no longer appear in the output.

29. Reboot the backup node. For example:

    ```
    # reboot
    ```

30. On one node, uncomment all of the Gauntlet blocks in */var/ha/ha.conf*.

31. Copy the */var/ha/ha.conf* file from step 30 to the other node.

32. On both nodes, verify that the file */etc/config/routed.options* contains `-q` among any other options. For example, you might see

    ```
    -h -Prdisc_interval=45 -q
    ```

    (The `-q` option is added during the process of configuring interfaces for IRIS FailSafe, but might have been deleted by the Gauntlet administration tool. The `-q` option is required for IRIS FailSafe to function correctly.)

Testing Gauntlet Failover

After configuring Gauntlet and IRIS FailSafe Gauntlet as described in the previous section, follow this procedure to test Gauntlet configuration and failover:

1. Enter these commands on each node to start up IRIS FailSafe:

    ```
    # chkconfig failsafe on
    # /etc/init.d/failsafe start
    ```

2. Verify that each node is in normal state by giving this command on each node:

    ```
    # /usr/etc/ha_admin -i
    ha_admin: Node controller state normal
    ```

    If either node is not in normal state, wait 30 seconds and try the command again.

3. On the primary node, verify that Gauntlet is running by looking for the processes *ipfilterd* and *authsrv* in the output of *ps*. For example:

```
# ps -ef | grep ipfilterd
   root    161  1  0  11:47:20 ?  0:00 /usr/etc/ipfilterd
# ps -ef | grep authsrv
   root    324  1  0  11:47:25 ?  0:00 /usr/etc/authsrv -daemon 7777
```

4. On the primary node, shut down Gauntlet by entering this command:

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

   Shutting down Gauntlet should result in a failover to the backup node.

5. Verify that the primary node is now in standby state and the backup node is in degraded state by giving this command on the primary node:

```
# /usr/etc/ha_admin -a
Node controller states
          Node:            firewall1    State: standby
          Node:            firewall2    State: degraded

Interface-pairs
Interface-pair:                  one    Owner: firewall2
IP aliases in interface-pair one: fw_out
Interface-pair:                  two    Owner: firewall2
IP aliases in interface-pair two: fw_in
```

6. On the backup node, verify that Gauntlet is running by looking for the processes *ipfilterd* and *authsrv* in the output of *ps*. For example:

```
# ps -ef | grep ipfilterd
   root    161  1  0  11:47:20 ?  0:00 /usr/etc/ipfilterd
# ps -ef | grep authsrv
   root    324  1  0  11:47:25 ?  0:00 /usr/etc/authsrv -daemon 7777
```

7. On the backup node, verify that the IP aliases are configured on their corresponding interfaces. For example:

```
# netstat -i
Name Mtu   Network     Address            Ipkts Ierrs    Opkts Oerrs  Coll
ec0  1500  192.0.4     firewall2.             2     0      138     0     0
                       fw_out.
ec3  1500  192.0.5     priv-firewall2.   244063     9    11379     0  2062
gfe0 1500  192.0.3     fast-firewall2.        7     0        3     0     0
                       fw_in.
lo0  8304  loopback    localhost          60168     0    60168     0     0
```

In the output, look for the network interfaces used for the internal and external networks. In this example output (for the example shown in Figure 1-1), ec0 has the IP alias fw_out and gfe0 has the IP alias fw_in.

8. On the primary node, enter this command to return both nodes to normal state:

   ```
   # /usr/etc/ha_admin -fr
   ```

9. On the primary node, verify that Gauntlet is running by looking for the processes *ipfilterd* and *authsrv* in the output of *ps*. For example:

   ```
   # ps -ef | grep ipfilterd
      root    161  1  0  11:47:20 ?  0:00 /usr/etc/ipfilterd
   # ps -ef | grep authsrv
      root    324  1  0  11:47:25 ?  0:00 /usr/etc/authsrv -daemon 7777
   ```

10. Verify that the primary node is in normal state by entering this command on the primary node:

    ```
    # /usr/etc/ha_admin -i
    ha_admin: Node controller state normal
    ```

    If the node is not in normal state, wait 30 seconds and try the command again.

Chapter 2

Configuration File Blocks for Gauntlet

Configuration parameters for Gauntlet must be specified in the configuration file */var/ha/ha.conf*. The sections in this chapter describe each Gauntlet specific block that must be added and the configuration parameters within each of those blocks. It also describes a requirement for the heartbeat block that is imposed by Gauntlet. The sections are as follows:

- "Gauntlet Application-Class Block" on page 13
- "Gauntlet Block" on page 14
- "Gauntlet Action and Action-Timer Blocks" on page 14
- "Gauntlet and the Heartbeat Sections" on page 15

The examples in this chapter show the Gauntlet configuration file blocks for the example discussed in the section "Planning IRIS FailSafe Gauntlet Configuration" in Chapter 1.

Gauntlet Application-Class Block

Example 2-1 shows the application-class block in a Gauntlet configuration.

Example 2-1 Gauntlet Application-Class Block

```
application-class gauntlet
{
        server-node = firewall1
}
```

The application-class gauntlet block contains this configuration parameter:

server-node Lists the node that is the primary server for the Gauntlet firewall. The value must match a node block label. Because Gauntlet is supported only in an active/backup configuration, this block can have at most one server-node parameter.

Gauntlet Block

IRIS FailSafe configuration files contain one Gauntlet block. Example 2-2 shows an example gauntlet block.

Example 2-2 Gauntlet Block

```
gauntlet firewall
{
        server-node = firewall1
        backup-node = firewall2
        startup-script = /etc/init.d/gauntlet
}
```

The label for the gauntlet block, firewall in this example, is a name of your choice. The configuration parameters used in gauntlet blocks are as follows:

server-node The primary node for the Gauntlet firewall. The value must be the same as the value of the server-node parameter in the application-class gauntlet block.

backup-node The backup node for the Gauntlet firewall. Its value matches the label of the node block for the node that is not the primary node for Gauntlet.

startup-script The location of the script used by IRIS FailSafe to start up Gauntlet.

Gauntlet Action and Action-Timer Blocks

Example 2-3 shows the action and action-timer blocks for Gauntlet. The action block specifies the pathnames of the local monitoring script, and the action-timer block specifies monitoring timing and timeout values for the monitoring of the *ipfilterd* daemon and the Gauntlet *authsrv* process.

Example 2-3 Gauntlet Action and Action-Timer Blocks

```
action gauntlet
{
        local-monitor = /var/ha/actions/ha_gauntlet_lmon
}

action-timer gauntlet
{
        start-monitor-time = 120
        lmon-probe-time = 60
        lmon-timeout = 30
}
```

The parameters used in action and action-timer blocks for Gauntlet are as follows:

local-monitor The pathname of the local monitoring script for Gauntlet. Do not change this value.

start-monitor-time

Specifies the amount of time that the application monitor waits before it starts using the local monitoring script to monitor the *ipfilterd* daemon and the Gauntlet *authsrv* process. The value of this parameter should be greater than or equal to the value of long-timeout. The suggested value is 120.

lmon-probe-time

The local monitoring script for Gauntlet is executed by the application monitor this often (in seconds). The suggested value is 60.

lmon-timeout

The local monitoring script for Gauntlet is considered to have timed out if no response is received in this many seconds. The suggested value is 30.

Gauntlet and the Heartbeat Sections

For security reasons, on IRIS FailSafe clusters that provide Gauntlet as a high-availability service, heartbeat messages that normally occur on the private network cannot fail over to a public network if the private network fails. For this reason, the heartbeat section of each node block cannot contain the parameter hb-public-ipname. Example 2-4 shows a heartbeat section when Gauntlet is an application class.

Example 2-4 Heartbeat Section With Gauntlet

```
node firewall1
{
        ...
        heartbeat
        {
                hb-private-ipname = priv-firewall1
                /* hb-public-ipname = must not be present */
                hb-probe-time = 5
                hb-timeout = 5
                hb-lost-count = 3
        }
        ...
}
```