This section describes how to configure a health probe on the Cisco 4700 Series Application Control Engine (ACE) appliance.

Guide Contents
Overview
Setting Up an ACE Appliance
Creating a Virtual Context
Configuring Access Control Lists
Configuring Role-Based Access Control
Configuring Server Load Balancing
Configuring a Load-Balancing Predictor
Configuring Server Persistence Using Stickiness
Configuring SSL Security
Configuring Health Monitoring Using Health Probes (this section)

Contents

Overview

After reading this section, you should have a basic understanding of how the ACE appliance supports server health monitoring using health probes, and how to configure an HTTP health probe.

To detect failures and make reliable load-balancing decisions, you can configure the ACE appliance to track the health of servers and server farms by periodically sending out health probes (sometimes referred to as keepalives). By default, the ACE implicitly checks for server failures.

You can configure probes on the ACE to make active connections and explicitly send traffic to servers. The ACE evaluates the server?s response to determine the health of that server.

When the ACE determines the health of a server, the result is one of the following:

- Passed?The server returned a valid response.
- Failed?The server failed to provide a valid response to the ACE within a specified number of retries.

When a server fails in response to the probe, the ACE can check for network problems that prevent a client from accessing that server. The ACE can place the server out of service.

A probe can be any of several types, including TCP, UDP, ICMP, Telnet, and HTTP. You can also configure scripted probes using the TCL scripting language.

You can configure a probe by following these steps:

1. Create the probe and specify its name, type, and attributes.

2. Associate the probe with one of the following:

    ◊ A real server.

    ◊ A real server that is associated with a server farm. You can associate a single probe or multiple probes to a real server within a server farm.

    ◊ A server farm. All real servers in the server farm receive the probe.

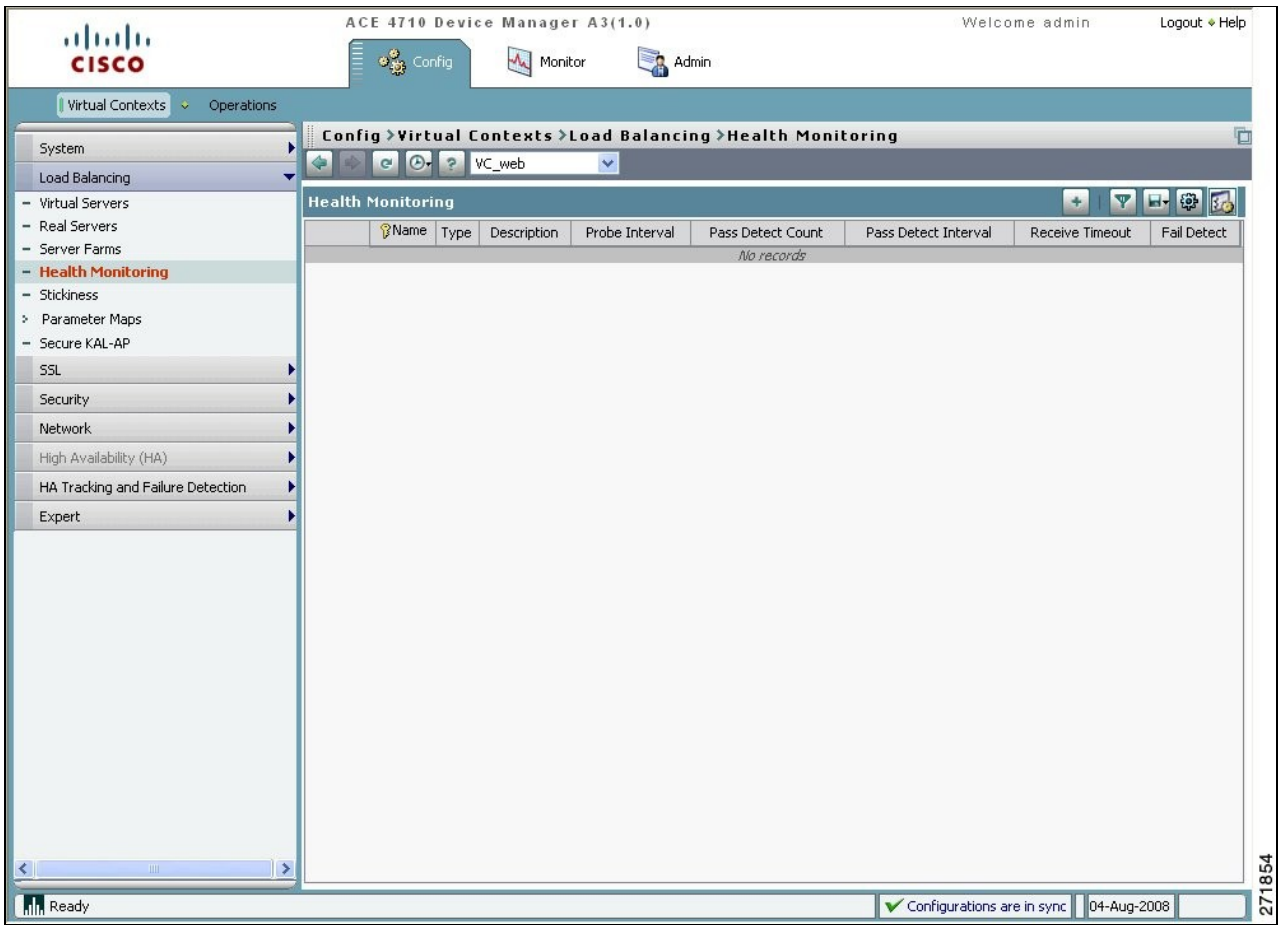
You can configure a probe by using either the ACE Device Manager GUI or the CLI. This section describes how to configure an HTTP probe. For information on how to configure other types of probes, see Cisco ACE 4700 Series Appliance Server Load-Balancing Configuration Guide.

Configuring an HTTP Health Probe Using the Device Manager GUI

You can configure an HTTP health probe using the ACE Device Manager GUI by following these steps:

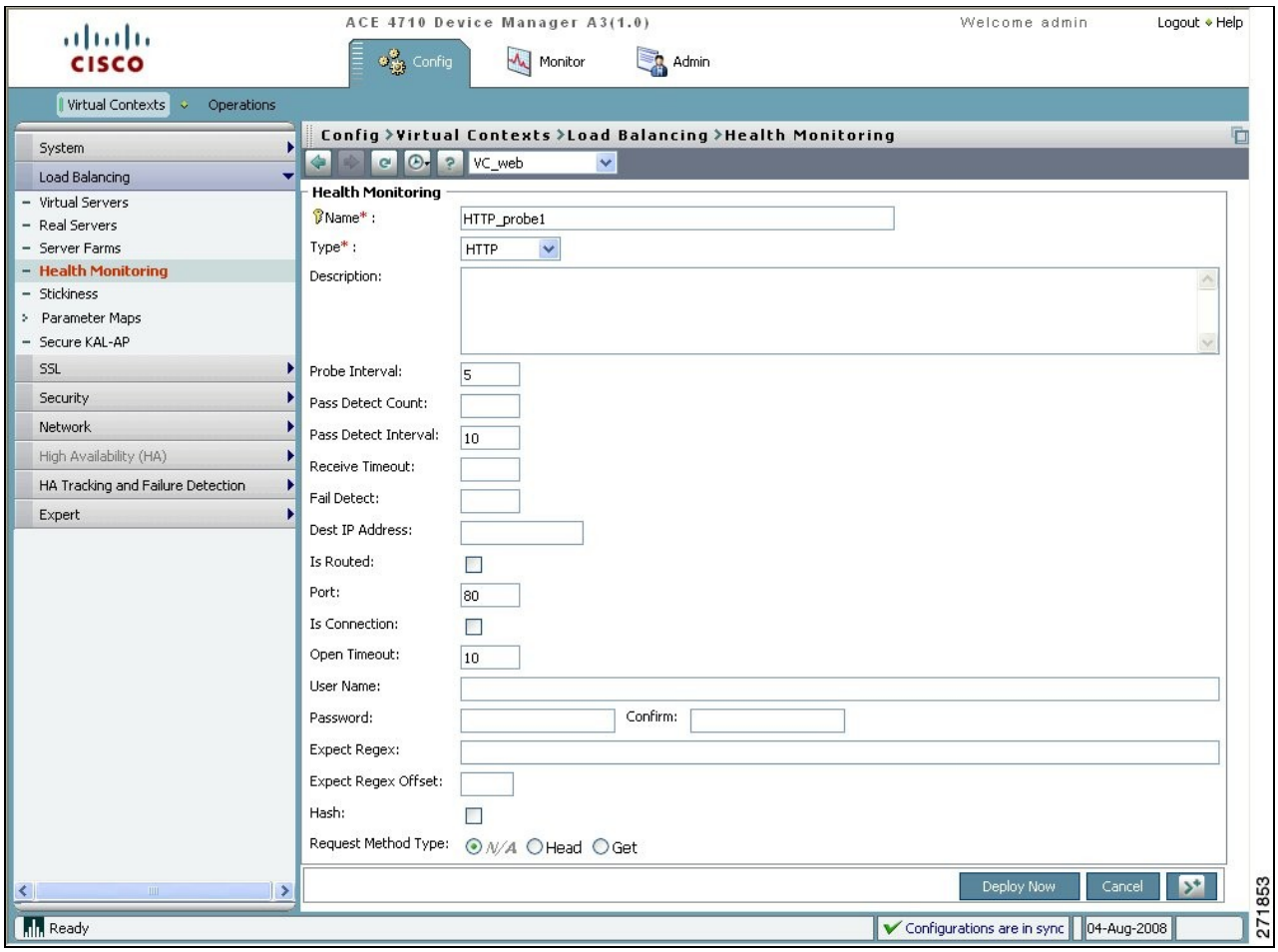
1. Choose **Load Balancing > Health Monitoring**. The Health Monitoring pane appears (Figure 1).

Figure 1 Health Monitoring Pane



2. Click **Add** to add a new health probe. The Health Monitoring window appears (Figure 2).

Figure 2 Health Monitoring Window



3. Enter the following health probe attributes. Leave the remaining attributes blank or with their default values.

◊ Name: HTTP_probe1

◊ Type: HTTP

◊ Probe Interval: 5

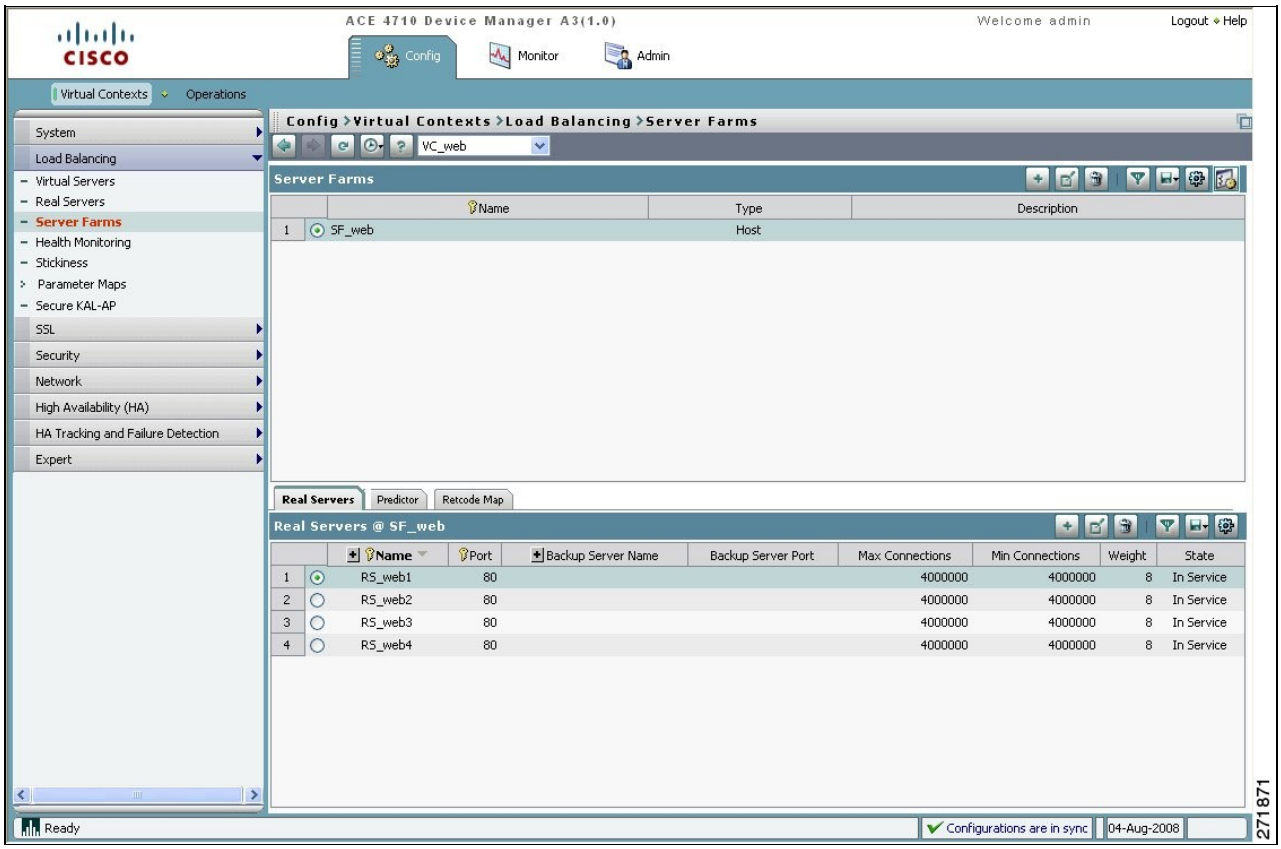
◊ Pass Detect Interval: 10

◊ Port: 80

4. Click **Deploy Now** to deploy this configuration on the ACE appliance.

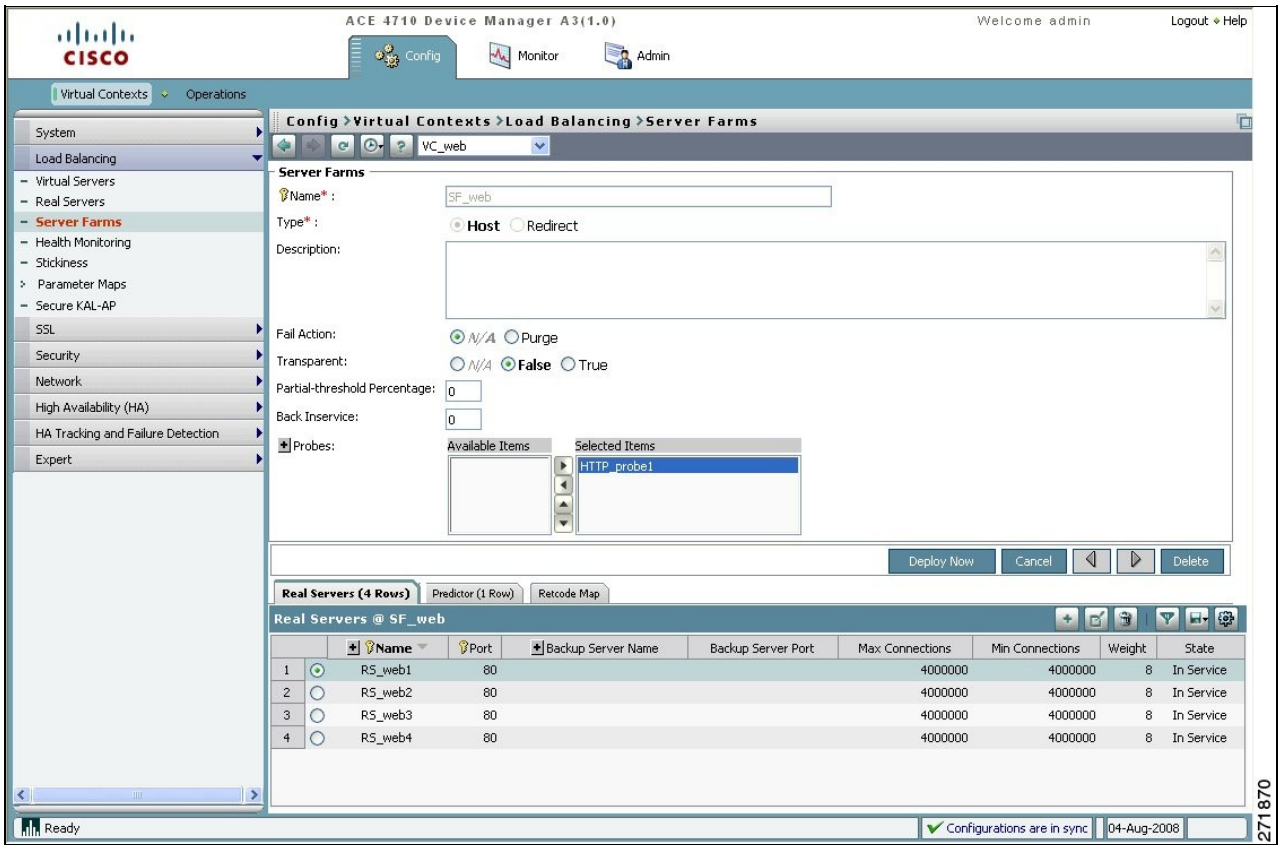
5. Associate the health probe with a server farm by choosing **Load Balancing > Server Farms**. The Server Farms pane appears (Figure 3).

Figure 3 Server Farms Pane



6. Choose the server farm **SF_web** and click **Edit**. The Server Farms window appears (Figure 4).

Figure 4 Server Farms Window



7. For Probes, choose **HTTP_probe1** from the Available Items list, and click the **right-arrow** button to move the probe to the Selected Items list.

8. Click **Deploy Now** to associate the health probe HTTP_probe1 with the server farm SF_web.

Configuring an HTTP Health Probe Using the CLI

You can configure an HTTP health probe using the CLI by following these steps:

1. Verify that you are operating in the desired context by checking the CLI prompt. If necessary, change to the correct context.

```
host1/Admin# changeto VC_web

host1/VC_web#
```

2. Enter configuration mode.

```
host1/VC_web# config

host1/VC_web(config)#
```

3. Define an HTTP probe named HTPP_probe1 to access its configuration mode.

```
host1/VC_web(config)# probe http HTTP_probe1
```

```
host1/VC_web(config-probe-http)#
```

4. Configure port number 80 for the HTTP probe.

```
host1/VC_web(config-probe-http)# port 80
```

5. Configure a time interval of 5 seconds between probes.

```
host1/VC_web(config-probe-http)# interval 5
```

6. Configure a pass detect interval of 10 seconds, after which the ACE will send another probe to a failed server.

```
host1/VC_web(config-probe-http)# passdetect interval 10
```

7. Exit probe configuration mode.

```
host1/VC_web(config-probe-http)# exit

host1/VC_web(config)#
```

8. Associate the probe HTTP_probe1 with the server farm SF_web, and exit configuration mode.

```
host1/VC_web(config)# serverfarm SF_web

host1/VC_web(config-sfarm-host)# probe HTTP_probe1

host1/VC_web(config-sfarm-host)# exit

host1/VC_web(config)# exit

host1/VC_web#
```

9. Display the HTTP probe configuration.

```
host1/VC_web# show running-config probe
```

In this section, you have configured an HTTP health probe.