CHAPTER 17

Network Management, MIBs, and SNMP on the Home Agent

This chapter contains information pertaining to various aspects of Network Management on the Cisco Mobile Wireless Home Agent.

This chapter includes the following sections:

- Operating and Maintaining the Cisco Mobile Wireless Home Agent, page 17-1
- Statistics, page 17-2
- SNMP, MIBs and Network Management, page 17-2
- Conditional Debugging, page 17-5
- Monitoring and Maintaining the HA, page 17-5

Operating and Maintaining the Cisco Mobile Wireless Home Agent

This section describes configuration details, statistics, and MIBs supported by the Home Agent. A definitive description of each Mobile IP command can be found at the following URL: http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fipras_r/1rfmobip.htm

The Home Agent can be managed using either the Cisco IOS CLI or using Cisco Works for Mobile Wireless.

Cisco's Mobile Wireless Home Agent has the following configurable parameters:

- Managing user profiles (local users)
- Configuring IP pools locally
- Configuring security associations with communicating nodes
- Configuring ingress/egress filtering
- Configuring mobile binding updates
- Configuring routing information

Statistics

The Mobile Wireless Home Agent maintains statistics on a global basis for the following parameters:

- Advertisements, received and sent
- Registrations, requests and replies
- Registrations, accepted and denied
- Bindings
- Binding Updates
- Gratuitous and Proxy ARPs
- Route Optimization Binding Updates

The Mobile Wireless Home Agent maintains statistics on a per FA-HA tunnel basis for the following parameters:

- Source and Destination IP address of the tunnel
- Tunnel Type, IPinIP or GRE
- Reverse Tunneling allowed
- Number of Users using that tunnel
- Traffic sent on the tunnel, packets and bytes
- Traffic received on the tunnel, packets and bytes

The Mobile Wireless Home Agent maintains statistics per Host, identified by NAI or Home IP Address, for the following parameters:

- Lifetime
- Session duration
- Traffic transmitted to the host, packets and bytes
- Traffic received from the host on the reverse tunnel, packets and bytes

Note The statistics can be cleared from the CLI. The MIB counters are not cleared.

SNMP, MIBs and Network Management

The HA implements SNMPv2 as specified in the suite of protocols: RFC 1901 to RFC 1908. The Home Agent supports the MIB defined in The Definitions of Managed Objects for IP Mobility Support UsingSMIv2, RFC 2006, October 1995. An additional Cisco MIB, CISCO- MOBILE-IP-MIB provides enhanced management capabilities. The RADIUS MIB, as defined in RADIUS Authentication Client MIB, RFC 2618, June 1999. A full list of MIBs that are supported on the Cisco 7600 series platform can be found at the following URL: http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

Session counters maintained in the MIB cannot be reset using SNMP or the Cisco IOS CLI. Home Agent CPU and Memory Utilization counters are accessible using the CISCO-PROCESS-MIB.

Release 3.0 adds a Home Agent Version MIB Object.

SNMPv3 is supported.

HA Release 5.0 MIB Enhancements

In HA Release 5.0, the CISCO-MOBILE-IP-MIB has the MAC address added as a per binding variable. The RADIUS-CLIENT-AUTHENTICATION-MIB contains entries for timeout on AAA access. The trap is added in the CISCO-RADIUS-MIB. The new CISCO-SLB-DFP-MIB is added.

For more information about MIBs, please refer to the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

CLI for IP-LOCAL-POOL-MIB

Cisco Mobile Wireless Home Agent Release 3.0 enhanced the CISCO-IP-LOCAL-POOL-MIB to generate traps when pool utilization reached a low threshold or high threshold in percentage. Objects "cIpLocalPoolPercentAddrThldLo" and "cIpLocalPoolPercentAddrThldHi" are defined for the high and low threshold watermark, respectively.

When the percentage of used addresses in an IP local pool equals or exceeds the high threshold, a "cilpPercentAddrUsedHiNotif" notification is generated. Once the notification is generated, it is disarmed and will not be generated again until the number of used addresses falls below the value indicated by "cIpLocalPoolPercentAddrThldLo".

When the percentage of used addresses in an IP local pool falls below the low threshold, a "cilpPercentAddrUsedLoNotif" notification will be generated. Once the notification is generated, it is disarmed and will not be generated again until the number of used addresses equals or exceeds the value indicated by "cIpLocalPoolPercentAddrThldHi".

The Cisco IOS 12.3(11)YX5 release implements new variables to the **ip local pool** command to configure the low and high threshold.

The command syntax is as follows:

ip local pool {**default** | ***poolname***} [*low-ip-address* [*high-ip-address*]] [**group** *group-name*] [**cache-size** *size*] [**threshold** *low-threshold high-threshold*]

The *low-threshold* argument is the low threshold to generate pool utilization traps, and *high threshold* argument is the high threshold to generate pool utilization traps.

Additionally, two additional varbinds will be seen in cilpPercentAddrUsedHiNotif notification:

- cIpLocalPoolChildIndex : IP Pool Name
- cIpLocalPoolPercentAddrThldHi: High IP Local Pool threshold percentage value

And two additional varbinds will be seen in cilpPercentAddrUsedLoNotif notification:

- cIpLocalPoolChildIndex : IP Pool Name
- cIpLocalPoolPercentAddrThldLo: : Low IP Local Pool threshold percentage value

Note The CISCO-IP-LOCAL-MIB file has not been changed as per the SNMP SMIv2 standard.

Restrictions

The following restrictions apply to the IP Local Pool Threshold Trap:

- The IP Local Pool name can be up to 240 ASCII characters long (depending on the parameters used).
- SNMP Trap names are limited to a maximum of 48 characters in length because the SNMP MIB only supports names that are up to 48 characters long.
- No Trap is generated if the Pool Name is longer than 48 characters.

How to Configure IP Overlapping Address Pools

This section contains the following procedure:

- Configuring and Verifying a Local Pool Group

Configuring and Verifying a Local Pool Group

This section contains the steps necessary to configure a local pool group and verify that it exists.

SUMMARY STEPS

1. enable
2. configure terminal
3. **ip local pool** {**default** | ***poolname***} [*low-ip-address* [*high-ip-address*]] [**group** *group-name*] [**cache-size** *size*] [**threshold** *low-threshold high-threshold*]
4. **show ip local pool** [*poolname* | [**group** *group-name*]]

Detailed Steps

	Command or Action	Purpose
Step 1	**enable** Example: `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip local pool** {**default**	*poolname*} [*low-ip-address* [*high-ip-address*]] [**group** *group-name*] [**cache-size** *size*] [**threshold** *low-threshold high-threshold*] Example: `Router(config)# ip local pool XYZPool 100.1.1.1 100.1.1.10 group MWG cache-size 50 threshold 50 90`	Configures a group of local IP address pools, gives this group a name, and specifies a cache size. *low-threshold* is the low threshold configured to generate pool utilization traps. The value of this variable should never be greater than the value the *high threshold.* *high threshold* is the high threshold configured to generate pool utilization traps. The value of this variable should never be less than the value the *lowthreshold.*
Step 4	**show ip local pool** [*poolname*	[**group** *group-name*]] Example: `Router(config)# show ip local pool group testgroup testpool`	Displays statistics for any defined IP address pools.

Conditional Debugging

The HA supports conditional debugging based on NAI, as well as conditional debugging based on the MN's Home address. Only AAA and Mobile IP components will support conditional debugging.

From the CLI, it is possible to trace activity of all or a particular user identified by NAI. Monitoring the activity of a particular user, called conditional debugging, will display the user activity related to Mobile IP messages and the RADIUS messages.

Starting in Release 3.0, an option is provided to display the condition (username/IMSI), along with each debug statement. This helps to match a debug statement to its condition. To enable this feature, use the following command:

ip mobile home-agent debug include username

The following MobileIP debugs are supported for conditional debugging:

- **debug ip mobile**
- **debug ip mobile host**

The following AAA debugs are supported for conditional debugging:

- **debug aaa authentication**
- **debug aaa authorization**
- **debug aaa accounting**
- **debug aaa ipc**
- **debug aaa attr**
- **debug aaa id**
- **debug aaa subsys**

The following RADIUS debugs are supported for conditional debugging:

- **debug radius**
- **debug radius accounting**
- **debug radius authentication**
- **debug radius retransmit**
- **debug radius failover**
- **debug radius brief**

Monitoring and Maintaining the HA

To monitor and maintain the HA, use the following commands in privileged EXEC mode:

Command	Purpose
`Router# clear ip mobile binding`	Removes mobility bindings.
`Router# clear ip mobile host-counters`	Clears the mobility counters specific to each mobile station.
`Router#clear ip mobile secure`	Clears and retrieves remote security associations.

Command	Purpose
`Router#clear ip mobile traffic`	Clears IP mobile traffic counters.
`Router# debug ip mobile advertise`	Displays advertisement information.
`Router# debug aaa pod`	Displays debug information for Radius Disconnect message processing at AAA subsystem level
`Router# debug ip mobile ?` `advertise` Mobility Agent advertisements `dfp` DFP Agent `host` Mobile host activities `ipc` Distributed HA Mobile activities `local-area` Local area mobility `mib` Mobile MIB Events `redundancy` MobileIP redundancy debugging `router` Mobile router activities `udp-tunneling` UDP Tunneling `vpdn-tunnel` VPDN tunnel	Displays IP mobility activities. The following list identifies all of the various options for the debug ip mobile command: • **advertise**-Mobility Agent advertisements • **dfp**-DFP Agent • **host**-Mobile host activities • **ipc**-Distributed HA Mobile activities • **local-area**-Local area mobility • **mib**-Mobile MIB Events • **redundancy**-MobileIP redundancy debugging • **router**-Mobile router activities • **udp-tunneling**-UDP Tunneling • **vpdn-tunnel**-VPDN tunnel
`Router# debug ip mobile host mac`	Displays mobility event information. In HA Release 5.0 a new option is introduced. The **mac** keyword displays the MN identified by the MAC address.
`Router# debug ip mobile redundancy`	Displays display IP mobility events.
`Router# debug radius`	Displays information associated with RADIUS.
`Router#debug tacacs`	Displays information associated with TACACS.
`Router# show ip mobile binding`	Displays the mobility binding table.
`Router# show ip mobile binding vrf`	Displays all the bindings on the HA that are VRF-enabled.
`Router#show ip mobile binding vrf realm`	Displays all bindings for the realm that are VRF-enabled.
`Router#show ip mobile globals`	Displays global information for Mobile Agents.
`Router# show ip mobile host`	Displays mobile station counters and information.
`Router# show ip mobile proxy`	Displays information about a proxy Mobile IP host.
`Router#show ip mobile secure`	Displays mobility security associations for Mobile IP.
`Router#show ip mobile traffic`	Displays Home Agent protocol counters. For Single IP, this command shows all redundancy binding counters as 0. For these counters there is a new command introduced **show ip mobile redundancy statistics.**

Command	Purpose
`Router#show ip mobile redundancy statistics`	Displays the redundancy status of the HA.
`Router#show ip mobile tunnel`	Displays information about the mobile IP tunnel.
`Router#show ip mobile violation`	Displays information about security violations.
`Router#show ip route vrf`	Displays the routing table information corresponding to a VRF.