



EMC® Watch4net®

Version 6.3u2

IP-SLA Probe Configuration CLI Guide

P/N: pending

Rev : 40606

EMC²

Copyright ©2010 - 2014 EMC Corporation. All rights reserved. Published in the USA.

Published April, 2014

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

The information in this publication is provided as is. EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose. Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC², EMC, and the EMC logo are registered trademarks or trademarks of EMC Corporation in the United States and other countries. All other trademarks used herein are the property of their respective owners.

For the most up-to-date regulatory document for your product line, go to EMC Online Support (<https://support.emc.com>).

Contents

Preface	iii
Purpose	iii
Intended Audience	iii
Prerequisites	iii
Document Organization	iv
Where to get help	v
1 Overview	1
1 About probe creation and management	2
2 Listing installed IP-SLA probes	2
3 Communication with SNMP Agents	3
2 Requirements	5
3 Creating IP-SLA Probes	7
4 Creating IP-SLA Probes with CSV Configuration Files	11
1 DNS Lookup	11
2 FTP get operation	12
3 HTTP get operation	13
4 IcmpPathEcho	13
5 IplcmpEcho	13
6 UdpJitter	14
7 TcpConnect	14
8 Sample probe CSV Configurations File	16
5 Creating IP-SLA Probes with XML Configuration Files	17
1 Template Based	19
2 Probe File	20
3 Probe Template File	22

3.1	DNS Lookup	22
3.2	FTP Get Operation	24
3.3	HTTP Get Operation	26
3.4	ICMP Path Echo	28
3.5	ICMP Echo	30
3.6	UDP Jitter	32
3.7	TCP Connect	34
3.8	Cisco IOS Post 15.0 probe creation	36
4	Agent Configuration File	38
6	Removing IP-SLA Probes	41
1	Removing installed IP-SLA probes with CSV configuration files	43
2	Removing installed IP-SLA probes with XML configuration files	43
3	Other methods of removing probes	43
7	Saving the Running Configuration	45
8	OID Mapping Files	47
1	Available probe configuration parameters	47
2	oidProperty	48
2.1	oidProperty.xsd	49
2.2	Probe OID Mapping	50
2.3	Agent Configuration OID Mapping	50

Preface

Purpose

The purpose of this guide is to provide information about the ***IP-SLA Probe Configuration*** command line interface and the way to use it. This guide includes all necessary information for any user of this module.

Intended Audience

This guide is intended to be read by any user of the ***IP-SLA Probe Configuration*** module who will use the interface to create, delete and manage IP-SLA probes.

Prerequisites

It is assumed that the ***IP-SLA Probe Configuration*** command line interface has been properly installed and configured on the server.

Document Organization

1. Overview	This section gives a brief description of the <i>IP-SLA Probe Configuration</i> module.
2. Requirements	This section gives a brief description of the requirements for this module.
3. Creating IP-SLA Probes	This section details the commands used to create probes.
4. Creating IP-SLA Probes with CSV Configuration Files	This section details CSV Probe creation.
5. Creating IP-SLA Probes with XML Configuration Files	This section details XML Probe creation.
6. Removing IP-SLA Probes	This section details the commands used to delete probes.
7. Saving the Running Configuration	This section details the commands used to save the running configuration.
8. OID Mapping Files	This section describes the OID Mapping Files.

Where to get help

EMC support, product, and licensing information can be obtained as follows:

PRODUCT INFORMATION - For documentation, release notes, software updates, or information about EMC products, go to EMC Online Support at:

<https://support.emc.com>

TECHNICAL SUPPORT - Go to EMC Online Support and click Service Center. You will see several options for contacting EMC Technical Support. Note that to open a service request, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or with questions about your account.

1

Overview

Watch4net's ***IP-SLA Probe Configuration*** command line interface is specifically designed to create, delete and manage most common IP-SLA probes running on Cisco IP-SLA enabled devices. Supported probe types include:

- DHCP ,
- DNS Lookup,
- FTP Get Operation,
- HTTP Get Operation,
- ICMP Path Echo,
- IP ICMP Echo (ping),
- UDP Jitter,
- TCP Connect.

The Cisco IP-SLA configuration module is also designed to help save the running configuration of an agent. This is especially helpful after creating probes that are expected to run permanently on a device.

1 About probe creation and management

This module was created, at first, to accompany the Report Pack for Cisco IP-SLA. Some of the probes that can be created by the **IP-SLA Probe Configuration** are not supported by the report pack.

When using the Report Pack, since APG keeps historical data about every metric it receives, creating a probe to replace a new one can be tricky as probes are identified by their index number. So, when a new probe is created in place of an old one using the same index number, this new probe should be the same type as the one it replaces. It could be helpful to organize your probes as follows:

Probe index range	Probe type
1 - 30	UDP Jitter
30 - 60	HTTP
60 - 90	Other types, EXCEPT IplcmpEcho, PathEcho, PathJitter
90 - 120	IplcmpEcho
120 - 150	PathEcho (Not handled by the report pack)
150 - 180	PathJitter (Not handled by the report pack)

Note: If Path Echo and Path Jitter are used, make sure probes are created at indexes higher than the last IplcmpEcho. Doing otherwise will likely affect the polling of other probe types.

2 Listing installed IP-SLA probes

In order to list the IP-SLA probes running on the Cisco device, a walk may be performed on the rttMonCtrlAdminTable (.1.3.6.1.4.1.9.9.42.1.2.1), using Watch4net's MIB browser or a command line tool, such as net-snmp (snmpwalk).

Template Based

The **IP-SLA Probe Configuration** module is **template based**. A template represents two things: values to be set on a list of OIDs and the order in which these values should be set. No probe can be created on an agent without defining the template first.

Keep in mind the following:

First, some IP-SLA probe attributes (OIDs) tend to have particular values that are always the same for probes of the same type.

Second, when probes are created or removed, the order in which the values of OIDs are set on an SNMP agent is important. This order is mostly influenced by the probe type.

Therefore, you can create one template that can be used by many probes of the same type.

Note: The correct order of the OIDs is enforced when creating and editing probe configurations in the **IP-SLA Probe Configuration** module. You do have control over (and therefore should verify) the order of OIDs if you edit configuration XML files directly in the *APG/Web-Servers/Tomcat/Default/webapps/ip-sla-probes/data* directory.

Finally, every probe can be identified by:

- a device (reachable through its SNMP agent) on which it is intended to run
- its index on that agent
- a probe template containing its basic values and the order in which they are to be set

Note: The agent and the index are considered as the “primary keys” of a probe. No two probes can have the same index on the same agent.

3 Communication with SNMP Agents

The **IP-SLA Probe Configuration** Module only works if it can read and write information on target SNMP Agents. The communication with SNMP Agents is either done with direct access or through a collector.

For example, if the server on which the module is installed doesn't have access to an agent, SNMP Collectors that are configured in APG can be used to access that agent. This agent is then configured in relation with the collector and so are the probes and probe templates.

When the server on which the **IP-SLA Probe Configuration** Module is installed has direct access to a SNMP agent, there is no need to associate that agent to a SNMP Collector.

For every collector, there is a node in the Menu and a subfolder in which are saved all configurations associated to that collector.

Note: To avoid confusion, in this document, probes are destroyed from an agent or probe configurations are deleted.

2

Requirements

1. Only Cisco IP-SLA enabled devices are supported. The device's SNMP Agent MUST support the CISCO-RTTMON-MIB as well as provide the required access rights to set some oids within this MIB and the CISCO-CONFIG-COPY-CAPABILITY MIB.
2. Individual probe support also depends upon the Cisco IOS version installed on the polled devices. Probe creation has been tested on IOS 12.2 and 12.4. Probe setup customization may be required for older IOS versions (prior to version 12.2).
3. Supported probes shall be setup as explained in this document. Watch4net provides basic scripts to create and delete the Cisco IP-SLA probes. Scripts are also provided to save the running configuration.

Creating IP-SLA Probes

Watch4net provides a very simple tool used to create sample IP-SLA probes. This tool is provided with no guarantee of suitability to a specific purpose and it is the customer's responsibility to set up the probes required to analyze their network performance.

To use the IP-SLA probe creation tool, simply execute:
//IPSLA_Configuration_DIR/bin/create-IPSLA-probes.sh

Note: A windows version is also available: *create-IPSLA-probes.cmd*

There are 2 possible modes when creating probes with this script: CSV and XML.
In CSV mode, all the configuration values for probes and the agents are put in one file.
In XML mode, on the other hand, the configuration values are separated in 3 files, one for the agents, one for the probe templates and one for the probes. This allows for a more flexible and more complete way of creating probes.

CSV Example:

```
create-IPSLA-probes.sh -d csv -p probeConfigFile -i interfaceIpAddress -o owner
```

XML Example:

```
create-IPSLA-probes.sh -d xml -a agentConfigFile -t probeTeamplateConfigFile -p probeConfigFile
```

All probes created using a CSV file have the following common characteristics:

- Their start time is now (as soon as the probe is created)
- The duration is unlimited (The probe will not expire)
- Lives, Buckets and History are set to none, except for the iplcmpEcho probes, for which it is user configurable

Note: Any pre-existing probe will be overwritten by the one created via this tool. If there is already a probe running at index 10 and a new one is created at index 10, the old probe will be deleted and overwritten by the new one.

The following is a list of all options available when using the creation script.

-d (Data Type): This represents the data type (or mode) that will be used for the configuration files. This option is mandatory. There are 2 possible values: "csv" and "xml".

-a (Agent File): This represents the path of the agent's configuration file. This option is not needed when the data type is set to CSV. On the other hand, it is mandatory when creating probes in XML mode.

-t (Probe Template File): This represents the path of the probe template configuration file. This option is not needed when the data type is set to CSV. On the other hand, it is mandatory when creating probes in XML mode.

-p (Probe File): This represents the path of the probe configuration file. This option is always mandatory.

-i (Interface): This represents the host interface to use in order to reach the SNMP agents of the Cisco devices on which the probes must be created. The option is optional and can only be used when

in CSV mode. When it isn't set, the default value "0.0.0.0" is used. In XML mode, this value can be set in the agent configuration file.

-o (Owner): This option represents the probe owner. It is optional and can only be used when in CSV mode. In XML mode, this value is set in the probe configuration file.

-h (Help): This option prints a help message. It shows an example of usage for each mode and prints the list of options with a small explanation for each one.

-m (Probe OID Mapping File): This represents the Probe OID Mapping File. It is optional and can be used in both modes. The only possible format for this file is XML.

4

Creating IP-SLA Probes with CSV Configuration Files

The following probes are supported by this tool:

1 DNS Lookup

Sets up a DNS lookup probe on the router with the following characteristics: type dns target-addr target-hostname — target-ip-address name-server ip-address

- Probe tag is configurable and should be unique to ease probe identification.
- Probe frequency is user configurable.

In the probe configurations file, the configuration fields appear in this order:
[*dnslookup*] [*router*] [*probeindex*] [*snmpwrite*] [*probetag*] [*probefrequency*]
[*lookupname*] [*dnsserveripaddress, indecimal formata.b.c.d*]

Example (all fields are separated by commas, no space between value and comma):
dns_lookup,192.168.1.3,10,snmpWriteString,test123,240,watch4net.com,10.128.7.2

2 FTP get operation

Sets up an FTP probe on the router with the following characteristics:
type ftp operation get url url [source-ipaddr ip-address — hostname]

- Probe tag is configurable and should be unique to ease probe identification.
- Probe frequency is user configurable.
- Threshold value is user configurable.
- Source address is optional, in IP address decimal format.

In the probe configurations file, the configuration fields appear in this order:
[*ftp*] [*router*] [*probeindex*] [*snmpwrite*] [*probetag*] [*probefrequency*]
[*target : ftp : //user : password@ftp.site.suffix/file*] [*thresholdvalue*] [*sourceaddress*]

Example (all fields are separated by commas, no space between value and comma):
ftp,192.168.1.3,10,snmpWriteString,test234,240,ftp://anonymous:anonymous@ftp.club.fr/index.html,10000

3 HTTP get operation

Sets up a HTTP probe on the router with the following characteristics:
type http operation get url url

- Probe tag is configurable and should be unique to ease probe identification.
- Probe frequency is user configurable.
- Source address is optional, in IP address decimal format.

In the probe configurations file, the configuration fields appear in this order:
[http] [router] [probeindex] [snmpwrite] [probetag] [probefrequency] [targeturl] [sourceaddress]

Example (all fields are separated by commas, no space between value and comma):
http,192.168.1.3,10,snmpWriteString,test345,240,http://www.watch4net.com

4 IcmpPathEcho

Sets up a IcmpPathEcho probe on the router with the following characteristics:
type pathEcho protocol icmpEcho destination-ip-address

- Probe tag is configurable and should be unique to ease probe identification.
- Probe frequency is user configurable.
- Source address is optional, in IP address decimal format.

In the probe configurations file, the configuration fields appear in this order:
[icmppathecho] [router] [probeindex] [snmpwrite] [probetag] [probefrequency]
[targetip, indecimal formata.b.c.d] [sourceaddress]

Example (all fields are separated by commas, no space between value and comma):
icmppathecho,192.168.1.3,10,snmpWriteString,test456,240,72.14.205.103

5 IplcmpEcho

Sets up a IplcmpEcho probe on the router with the following characteristics:
type echo protocol iplcmpEcho destination-ip-address

- Probe tag is configurable and should be unique to ease probe identification.
- Probe frequency is user configurable.
- Source address is optional, in IP address decimal format.

In the probe configurations file, the configuration fields appear in this order:

[ping] [router] [probeindex] [snmpwrite] [probetag] [probefrequency] [targetip, indecimala.b.c.dformat]
[sourceaddress] [lives] [buckets] [filterForHistory]

Example (all fields are separated by commas, no space between value and comma):
ping,192.168.1.3,10,snmpWriteString,test456,240,72.14.205.103,192.168.1.3,2,10,2

In this case, 2 lives and 10 buckets are kept. Filter for history is set to "ALL" (Value is 2).

6 UdpJitter

Sets up a UdpJitter probe on the router with the following characteristics:

type udpjitter dest-ipaddr ip-address dest-port port-number num-packets number-of-packets interval
inter-packet-interval

- Probe tag is configurable and should be unique to ease probe identification.
- Probe frequency is user configurable.
- Source address is optional, in IP address decimal format.

In the probe configurations file, the configuration fields appear in this order:

[udpjitter] [router] [probeindex] [snmpwrite] [probetag] [probefrequency] [targetip, indecimala.b.c.dformat]
[port] [interval] [packets] [sourceaddress]

Example (all fields are separated by commas, no space between value and comma):
udpjitter,192.168.1.3,10,snmpWriteString,test567,240,10.128.128.1,17000,20,20

7 TcpConnect

Sets up a TcpConnect probe on the router with the following characteristics:

type tcpconnect dest-ipaddr destination-ip-address dest-port port-number

- Probe tag is configurable and should be unique to ease probe identification.
- Probe frequency is user configurable.
- Source address is optional, in IP address decimal format.

In the probe configurations file, the configuration fields appear in this order:

*[tcpconnect] [router] [probeindex] [snmpwrite] [probetag] [probefrequency] [targetip, indecimala.b.c.dformat]
[targetport] [sourceaddress]*

Example (all fields are separated by commas, no space between value and comma):

tcpconnect,192.168.1.3,10,snmpWriteString,test678,60,72.14.205.103,80

8 Sample probe CSV Configurations File

```
tcpconnect,192.168.1.3,1,rpasswd,test678,60,72.14.205.103,80,192.168.1.3
dns_lookup,192.168.1.3,2,rpasswd,test123,240,watch4net.com,10.128.7.2
ftp,192.168.1.3,3,rpasswd,test234,240,ftp://anonymous:anonymous@ftp.club-internet.fr/index.html,10000,192.168.1.3
http,192.168.1.3,4,rpasswd,test345,240,http://www.watch4net.com,192.168.1.3
icmppathecho,192.168.1.3,5,rpasswd,test456,240,72.14.205.103,192.168.1.3
ping,192.168.1.3,6,rpasswd,test456,240,72.14.205.103,192.168.1.3
udpjitter,192.168.1.3,7,rpasswd,test567,240,10.128.128.1,17000,20,20,192.168.1.3
```

5

Creating IP-SLA Probes with XML Configuration Files

Creating IP-SLA Probes with XML files is far more flexible than creating them with CSV files. More attributes can be set. The probe creator has far more control.

Unlike the CSV method, it is possible to set a different start time for every probe. In fact, all probe attributes can be modified. For each probe type, some of the attributes have default values.

One very important default value is that of the `oidMappingType` that can be found in the Probe Template file and in the agent file. For example, in the probe template file, this attribute is in fact a key that is matched with the `oidMappingType` found in the Probe OID Mapping File. Always use one of the default values unless a matching configuration, with the same `oidMappingType`, has been created in the OID Mapping File. If you change this value, make sure to create the matching configuration in the Probe OID Mapping File. OID Mapping files

are discussed later.

Regardless of the probe type, some attributes are always mandatory and should always be assigned a value.

In the agent file, these attributes are:

- name
- address
- port
- readCommunity
- writeCommunity

In the probe template file, these attributes are:

- oidMappingType
- name
- type
- protocol
- frequency

In the probe file, these attributes are:

- index
- probeTemplateName
- agentName
- owner
- tag

1 Template Based

The IP-SLA Probe Configuration Module is template based. Many IP-SLA probes attributes tend to have particular values that are always the same depending on the probe type. Therefore, you have the possibility of creating templates that can be used by more than one probe of the same type.

Probes are created and removed on an agent through a series of SNMP commands. A template not only defines the values, but it also defines the order in which all SNMP commands are made. No probe can be created on an agent without defining the template first.

Every probe can be identified by:

- an SNMP agent on which it is intended to run
- its index on that agent
- a probe template containing its basic values

Note: The agent and the index are considered as the "primary keys". No 2 probes can have the same index on the same agent.

2 Probe File

```
<prb:probe>
  <prb:index>61</prb:index>
  <prb:probeTemplateName>DEFAULT\_ping</prb:probeTemplateName>
  <prb:agentName>firstAgent</prb:agentName>
  <prb:owner>owner</prb:owner>
  <prb:tag>DNS\_2</prb:tag>
  <prb:rttLife>2147483647</prb:rttLife>
  <prb:create>true</prb:create>
  <prb:delete>true</prb:delete>
</prb:probe>
```

index:

This attribute sets the value of the IP-SLA operation number. This is an integer greater or equal to 1.

probeTemplateName:

This attribute represents the probe template name. The matching probe template is in the probe template file.

agentName:

This attribute represents the agent name. The matching agent should exist in the agent file.

owner:

This identifies the entity that created the probe. It's a string value representing the probe owner.

tag:

This is a string which is used by a managing application to identify the RTT target. This maximum length for a tag is 16 characters.

rttLife:

This is a string which is used by a managing application to specify the life of the probe. This maximum and default value for this field is 2147483647 which is interpreted as forever.

create:

This is a Boolean value (true or false), with the default value being true. With the creation command, probes are created only if the value is true.

delete:

This is a Boolean value (true or false), with the default value being false. When removing probes, probes are deleted only if the value is true.

3 Probe Template File

Probe templates are at the core of the **IP-SLA Probe Configuration** design. Though they allow for flexibility as they can be edited, for every type of probe there is typical set of values and a typical order in which the values are set.

With the **IP-SLA Probe Configuration**, by defining a template, you have the possibility of redefining the values of any attributes and the order in which the SNMP calls are made.

The following sections describe the typical configuration for each type of probe supported by the **IP-SLA Probe Configuration**.

3.1 DNS Lookup

```
<ptmp:probeTemplate>
  <ptmp:oidMappingType>dns_lookup</ptmp:oidMappingType>
  <ptmp:name>DEFAULT_dns_lookup</ptmp:name>
  <ptmp:type>8</ptmp:type> <!-- DNS -->
  <ptmp:protocol>26</ptmp:protocol> <!-- DNS_APPL -->
  <ptmp:frequency>240</ptmp:frequency>
  <ptmp:targetAddressString>watch4net.com</ptmp:targetAddressString>
  <ptmp:nameServer>192.168.1.2</ptmp:nameServer>
</ptmp:probeTemplate>
```

oidMappingType:

The default value for DNS lookup probes is dns_lookup. If the value is changed, the matching configuration should be created in the Probe OID Mapping File.

Name:

This is the identifier of the probe template. Every probe template must have a unique name.

type:

This represents the type of RTT operation to be performed. The SNMP default value for a DNS Lookup probe is 8.

protocol:

This specifies the protocol to be used to perform the RTT operation. The SNMP default value for a DNS Lookup probe is 26.

frequency:

This sets the rate at which a specified IP SLA operation repeats. It specifies the duration between initiating each RTT operation. It is an integer greater or equal to 0. Cisco recommends that the frequency should not be lower than 60. Furthermore, the frequency should not be lower than the timeout of the operation.

targetAddressString:

This is a string which specifies the address of the target. This string can be an IP address or a hostname. This object is applicable to DNS probes only.

nameserver:

This is a string which specifies the IP address of the name-server. This object is applicable to DNS probes only.

3.2 FTP Get Operation

```
<ptmp:probeTemplate>
  <ptmp:oidMappingType>ftp</ptmp:oidMappingType>
  <ptmp:name>DEFAULT_ftp</ptmp:name>
  <ptmp:type>12</ptmp:type> <!-- FTP -->
  <ptmp:protocol>30</ptmp:protocol> <!-- FTP_APPL -->
  <ptmp:frequency>240</ptmp:frequency>
  <ptmp:operation>3</ptmp:operation> <!-- FTP_GET -->
  <ptmp:url>ftp://www.test.com/ptmp:index.html</ptmp:url>
  <ptmp:sourceAddress>192.168.1.1</ptmp:sourceAddress>
  <ptmp:threshold>10000</ptmp:threshold>
</ptmp:probeTemplate>
```

oidMappingType:

The default value for FTP Get Operation probes is ftp. If the value is changed, the matching configuration should be created in the Probe OID Mapping File.

Name:

This is the identifier of the probe template. Every probe template must have a unique name.

type:

This represents the type of RTT operation to be performed. The SNMP default value for a FTP Get probe is 12.

protocol:

This specifies the protocol to be used to perform the RTT operation. The SNMP default value for a DNS FTP Get probe is 30.

frequency:

This sets the rate at which a specified IP SLA operation repeats. It specifies the duration between initiating each RTT operation. It is an integer greater or equal to 0. Cisco recommends that the frequency should not be lower than 60. Furthermore, the frequency should not be lower than the timeout of the operation.

operation:

This is a code that represents the specific type of RTT operation. This object is applicable to http and ftp probes only.

sourceAddress:

This represents a string which specifies the IP address of the source. This object is applicable to all probes except DNS.

threshold:

This value defines an administrative threshold limit. It is an integer greater or equal to 0.

url:

This is a string which represents the URL to which a HTTP probe should communicate with.

3.3 HTTP Get Operation

```
<ptmp:probeTemplate>
  <ptmp:oidMappingType>http</ptmp:oidMappingType>
  <ptmp:name>DEFAULT_http</ptmp:name>
  <ptmp:type>7</ptmp:type> <!-- HTTP -->
  <ptmp:protocol>25</ptmp:protocol> <!-- HTTP_APPL -->
  <ptmp:frequency>240</ptmp:frequency>
  <ptmp:operation>1</ptmp:operation> <!-- HTTP_GET -->
  <ptmp:url>http://www.watch4net.com</ptmp:url>
  <ptmp:sourceAddress>192.168.1.1</ptmp:sourceAddress>
</ptmp:probeTemplate>
```

oidMappingType:

The default value for HTTP Get Operation probes is http. If the value is changed, the matching configuration should be created in the Probe OID Mapping File.

Name:

This is the identifier of the probe template. Every probe template must have a unique name.

type:

This represents the type of RTT operation to be performed. The SNMP default value for an HTTP Get probe is 7.

protocol:

This specifies the protocol to be used to perform the RTT operation. The SNMP default value for an HTTP Get probe is 25.

frequency:

This sets the rate at which a specified IP SLA operation repeats. It specifies the duration between initiating each RTT operation. It is an integer greater or equal to 0. Cisco recommends that the frequency should not be lower than 60. Furthermore, the frequency should not be lower than the timeout of the operation.

operation:

This is a code that represents the specific type of RTT operation. This object is applicable to http and ftp probes only.

url:

This is a string which represents the URL to which a HTTP probe should communicate with.

sourceAddress:

This represents a string which specifies the IP address of the source. This object is applicable to all probes except DNS.

3.4 ICMP Path Echo

```
<ptmp:probeTemplate>
  <ptmp:oidMappingType>icmppathecho</ptmp:oidMappingType>
  <ptmp:name>DEFAULT_icmppathecho</ptmp:name>
  <ptmp:type>2</ptmp:type> <!-- PATH_ECHO -->
  <ptmp:protocol>2</ptmp:protocol> <!-- IP_ICMP_ECHO -->
  <ptmp:frequency>240</ptmp:frequency>
  <ptmp:sourceAddress>192.168.1.1</ptmp:sourceAddress>
  <ptmp:targetAddress>72.14.205.103</ptmp:targetAddress>
</ptmp:probeTemplate>
```

oidMappingType:

The default value for ICMP Path Echo probes is icmppathecho. If the value is changed, the matching configuration should be created in the Probe OID Mapping File.

Name:

This is the identifier of the probe template. Every probe template must have a unique name.

type:

This represents the type of RTT operation to be performed. The SNMP default value for an ICMP Path Echo probe is 2.

protocol:

This specifies the protocol to be used to perform the RTT operation. The SNMP default value for an ICMP Path Echo probe is 2.

frequency:

This sets the rate at which a specified IP SLA operation repeats. It specifies the duration between initiating each RTT operation. It is an integer greater or equal to 0. Cisco recommends that the frequency should not be lower than 60. Furthermore, the frequency should not be lower than the timeout of the operation.

sourceAddress:

This represents a string which specifies the IP address of the source. This object is applicable to all probes except DNS.

targetAddress:

This is a string which specifies the address of the target. An IP address is expected.

3.5 ICMP Echo

```
<ptmp:probeTemplate>
  <ptmp:oidMappingType>ping</ptmp:oidMappingType>
  <ptmp:name>DEFAULT_ping</ptmp:name>
  <ptmp:type>1</ptmp:type> <!-- ECHO (ping) -->
  <ptmp:protocol>2</ptmp:protocol> <!-- IP_ICMP_ECHO -->
  <ptmp:frequency>240</ptmp:frequency>
  <ptmp:sourceAddress>192.168.1.1</ptmp:sourceAddress>
  <ptmp:targetAddress>72.14.205.103</ptmp:targetAddress>
</ptmp:probeTemplate>
```

oidMappingType:

The default value for IP ICMP Echo probes is ping. If the value is changed, the matching configuration should be created in the Probe OID Mapping File.

Name:

This is the identifier of the probe template. Every probe template must have a unique name.

type:

This represents the type of RTT operation to be performed. The SNMP default value for an IP ICMP Echo probe is 1.

protocol:

This specifies the protocol to be used to perform the RTT operation. The SNMP default value for an IP ICMP Echo probe is 2.

frequency:

This sets the rate at which a specified IP SLA operation repeats. It specifies the duration between initiating each RTT operation. It is an integer greater or equal to 0. Cisco recommends that the frequency should not be lower than 60. Furthermore, the frequency should not be lower than the timeout of the operation.

sourceAddress:

This represents a string which specifies the IP address of the source. This object is applicable to all probes except DNS.

targetAddress:

This is a string which specifies the address of the target. An IP address is expected.

3.6 UDP Jitter

```
<ptmp:probeTemplate>
  <ptmp:oidMappingType>udpjitter</ptmp:oidMappingType>
  <ptmp:name>DEFAULT_udpjitter</ptmp:name>
  <ptmp:type>9</ptmp:type> <!-- JITTER -->
  <ptmp:protocol>27</ptmp:protocol> <!-- JITTER_APPL -->
  <ptmp:frequency>240</ptmp:frequency>
  <ptmp:sourceAddress>192.168.1.1</ptmp:sourceAddress>
  <ptmp:targetAddress>192.168.1.1</ptmp:targetAddress>
  <ptmp:targetPort>17000</ptmp:targetPort>
  <ptmp:interval>20</ptmp:interval>
  <ptmp:numPackets>20</ptmp:numPackets>
  <ptmp:enableResponder>1</ptmp:enableResponder>
</ptmp:probeTemplate>
```

oidMappingType:

The default value for UDP Jitter probes is udpjitter. If the value is changed, the matching configuration should be created in the Probe OID Mapping File.

Name:

This is the identifier of the probe template. Every probe template must have a unique name.

type:

This represents the type of RTT operation to be performed. The SNMP default value for an UDP Jitter probe is 9.

protocol:

This specifies the protocol to be used to perform the RTT operation. The SNMP default value for an UDP Jitter probe is 27.

frequency:

This sets the rate at which a specified IP SLA operation repeats. It specifies the duration between initiating each RTT operation. It is an integer greater or equal to 0. Cisco recommends that the frequency should not be lower than 60. Furthermore, the frequency should not be lower than the timeout of the operation.

sourceAddress:

This represents a string which specifies the IP address of the source. This object is applicable to all probes except DNS.

targetAddress:

This is a string which specifies the address of the target. An IP address is expected.

targetPort:

This object represents the target's port number. It is applicable to udpEcho, tcpConnect and jitter probes.

interval:

This value represents the inter-packet delay between packets and is in milliseconds. This object is applicable to the Jitter probes only.

numPackets:

This value represents the number of packets that need to be transmitted. This object is applicable to Jitter probes only.

enableResponder:

This optional attribute allows the enabling of the responder. The possible values are 1 for true and 2 for false. When no value is specified, it's considered as set to false. If set to true, an attempt to map the targetAdress to an entry in the Agent Configuration File will be done. In other words, the Agent Configuration File must have a configuration for the agent with targetAdress as an IP address.

3.7 TCP Connect

```
<ptmp:probeTemplate>
  <ptmp:oidMappingType>tcpconnect</ptmp:oidMappingType>
  <ptmp:name>DEFAULT_tcpconnect</ptmp:name>
  <ptmp:type>6</ptmp:type> <!-- TCP_CONNECT -->
  <ptmp:protocol>24</ptmp:protocol> <!-- IP_TCP_CONN -->
  <ptmp:frequency>60</ptmp:frequency>
  <ptmp:sourceAddress>192.168.1.1</ptmp:sourceAddress>
  <ptmp:targetAddress>72.14.205.103</ptmp:targetAddress>
  <ptmp:targetPort>80</ptmp:targetPort>
</ptmp:probeTemplate>
```

oidMappingType:

The default value for TCP Connect probes is tcpconnect. If the value is changed, the matching configuration should be created in the Probe OID Mapping File.

Name:

This is the identifier of the probe template. Every probe template must have a unique name.

type:

This represents the type of RTT operation to be performed. The SNMP default value for a TCP Connect probe is 6.

protocol:

This specifies the protocol to be used to perform the RTT operation. The SNMP default value for TCP Connect probe is 24.

frequency:

This sets the rate at which a specified IP SLA operation repeats. It specifies the duration between initiating each RTT operation. It is an integer greater or equal to 0. Cisco recommends that the frequency should not be lower than 60. Furthermore, the frequency should not be lower than the timeout of the operation.

sourceAddress:

This represents a string which specifies the IP address of the source. This object is applicable to all probes except DNS.

targetAddress:

This is a string which specifies the address of the target. An IP address is expected.

targetPort:

This object represents the target's port number. This object is applicable to udpEcho, tcp-Connect and jitter probes.

3.8 Cisco IOS Post 15.0 probe creation

Several changes, which affect IP-SLA probe creation, were introduced in Cisco IOS 15.0. Basically, the same oids must be set all at once, in the same PDU, instead of separately. Typically, this translates to probe oid mapping similar to this:

```
<sla:probeMapping>
  <sla:oidMappingType>tcpconnect-post15.0</sla:oidMappingType>

  <sla:possibleType>-1</sla:possibleType>
  <sla:possibleType>6</sla:possibleType>

  <sla:enableResponder key="enableResponder" group="999" order="1" oid="rttMonApplResponder"/>
  <sla:filter key="filter" group="1" order="14" oid="rttMonHistoryAdminFilter"/>
  <sla:frequency key="frequency" group="1" order="5" oid="rttMonCtrlAdminFrequency"/>

  <sla:numBuckets key="numBuckets" group="1" order="13" oid="rttMonHistoryAdminNumBuckets"/>
  <sla:numLives key="numLives" group="1" order="12" oid="rttMonHistoryAdminNumLives"/>
  <sla:owner key="owner" group="1" order="7" oid="rttMonCtrlAdminOwner"/>
  <sla:protocol key="protocol" group="1" order="2" oid="rttMonEchoAdminProtocol"/>
  <sla:rttLife key="rttLife" group="1" order="11" oid="rttMonScheduleAdminRttLife"/>
  <sla:sourceAddress key="sourceAddress" group="1" order="9" oid="
rttMonEchoAdminSourceAddress"/>
  <sla:sourcePort key="sourcePort" group="1" order="10" oid="rttMonEchoAdminSourcePort"/>
  <sla:startTime key="startTime" group="1" order="15" oid="rttMonScheduleAdminRttStartTime"
snmpValue="1"/>
  <sla:tag key="tag" group="1" order="8" oid="rttMonCtrlAdminTag"/>
  <sla:targetAddress key="targetAddress" group="1" order="4" oid="
rttMonEchoAdminTargetAddress"/>
  <sla:targetPort key="targetPort" group="1" order="3" oid="rttMonEchoAdminTargetPort"/>
  <sla:type key="type" group="1" order="1" oid="rttMonCtrlAdminRttType"/>

  <sla:nvgen key="nvgen" group="1" order="6" oid="rttMonCtrlAdminNvgen" snmpValue="1"/>
  <sla:status key="status" group="1" order="16" oid="rttMonCtrlAdminStatus" snmpValue="4"/>
</sla:probeMapping>
```

Note: All group attributes are set to 1: this indicates that all set on oids are sent in 1 pdu.

In terms of probe template, this translates to:

```
<ptmp:probeTemplate>
  <ptmp:oidMappingType>tcpconnect-post15.0</ptmp:oidMappingType>
  <ptmp:name>tcpconnect-post-15.0</ptmp:name>
  <ptmp:type>6</ptmp:type>
  <ptmp:protocol>24</ptmp:protocol>
  <ptmp:frequency>60</ptmp:frequency>
  <ptmp:sourceAddress>192.168.1.1</ptmp:sourceAddress>
  <ptmp:targetAddress>72.14.205.103</ptmp:targetAddress>
  <ptmp:targetPort>80</ptmp:targetPort>
</ptmp:probeTemplate>
```

Only the oidMappingType has changed.

4 Agent Configuration File

The Agent Configuration File is used mainly to set up an SNMP Session on an agent. The most important values that are set in this file are the community strings and the Agent IP address. As for probes, agents are also matched to an entry in the OID mapping File. There is a default value for the oidMappingType which is matched to the default way of saving the running configuration.

```
<agt:agent>
  <agt:oidMappingType/>
  <agt:index/>
  <agt:address>192.168.1.1</agt:address>
  <agt:port>161</agt:port>
  <agt:snmpVersion>2</agt:snmpVersion>
  <agt:readCommunity>public</agt:readCommunity>
  <agt:writeCommunity>private</agt:writeCommunity>
  <agt:interfaceAddress>0.0.0.0</agt:interfaceAddress>
  <agt:interfacePort>0</agt:interfacePort>
  <agt:saveRunningConfiguration>1</agt:saveRunningConfiguration>
</agt:agent>
```

oidMappingType:

This attribute is mandatory and the default value is defaultAgentType. As for probes, if the value is changed, the matching configuration should be created in the Agent Configuration OID Mapping file.

index:

This attribute is mandatory and specifies a unique entry in the ccCopyTable. The default value is 0. When the value is 0, a random value will be generated and used in the ccCopyTable, as suggested in the MIB.

address:

This attribute is mandatory. It is the host name or IP address of the agent. It's also considered as a unique key for the identification of an agent.

port:

This attribute is mandatory. It is the port number used by the agent for Simple Network Management Protocol. The default value is 161.

snmpVersion:

This attribute is optional. It represents the SNMP version supported by an agent. The possible values are 1, 2 and 3, the default value being 2.

readCommunity:

This attribute is mandatory. Its default value is "public".

writeCommunity:

This attribute is mandatory. Its default value is "private".

Note: readCommunity and writeCommunity values may be encrypted with the crypt-password tool provided in APG/bin directory.

interfaceAddress:

This attribute is optional. It represents the local address the socket is bound to. This is the physical interface used to reach the agent. The default value is "0.0.0.0". If the IP address is 0.0.0.0, the socket will be bound to the wildcard address, an IP address chosen by the kernel.

interfacePort:

This attribute is optional. It represents the local port the socket is bound to. The default value is 0. The local port must be between 0 and 65535 inclusive.

saveRunningConfiguration:

This attribute is optional. The possible values are 1 for true and 2 for false. When no value is specified, it's considered as set to false. If set to true, when the "save-running-config" script is run, the running configuration of the agent will be saved in the Starting Configuration.

6

Removing IP-SLA Probes

To use the IP-SLA probe removal tool, simply execute:
//IPSLA_Configuration_DIR/bin/delete-IPSLA-probes.sh

Note: A windows version is also available: *delete-IPSLA-probes.cmd*

CSV Example:

`delete-IPSLA-probes.sh -d csv -p probeConfigurationsFile -i interfaceIpAddress`

XML Example:

`delete-IPSLA-probes.sh -d xml -a agentConfigurationsFile -p probeConfigurationsFile`

The following is a list of all options available when using the creation script.

-d (Data Type):

This represents the data type (or mode) that will be used for the configuration files. This option is mandatory. There are 2 possible values: "csv" and "xml".

-a (Agent File):

This represents the path of the agent configuration file. This option is not needed when the data type is set to CSV. On the other hand, it is mandatory when removing probes in XML mode.

-p (Probe File):

This represents the path of the probe configuration file. This option is always mandatory.

-i (Interface):

This represents the agent interface. The option is optional and can only be used when in CSV mode. When it isn't set, the default value "0.0.0.0" is used. In XML mode, this value can be set in the agent configuration file.

-o (Owner):

This option represents the probe owner. Though it is possible to use this option, it has no effect when removing probes.

-h (Help):

This option prints a help message. It shows an example of usage for each mode and prints the list of options with a small explanation for each one.

-m (Probe OID Mapping File):

This represents the Probe OID Mapping File. It is optional and can be used in both modes. The only possible format for this file is XML.

1 Removing installed IP-SLA probes with CSV configuration files

The probeConfigurationsFile contains the probes to delete, identified by the agent address on which they are set up and the probe index number. The file format is described below:

[RouterIpAddress],[probeIndex],[snmpWriteCommunityString]

2 Removing installed IP-SLA probes with XML configuration files

The same probe file used to create probes can be used to remove them. Only probes with the delete field set to true are deleted.

3 Other methods of removing probes

IP-SLA probes may also be removed by setting the rttMonCtrlAdminStatus (.1.3.6.1.4.1.9.9.42.1.2.1.1.9.probeIndex) value to 6 (destroy), via an SNMP set command line tool.

Probes can also be deleted from the router via the IOS command line interface:
telnet routerIpAddress – > enable – > config terminal – > no rtr [probeIndex] — no ipsla [probeIndex]

7

Saving the Running Configuration

Packaged with the IP-SLA Configuration tools is the possibility to save an agent's running configuration. In fact, we provide a tool to save all outstanding (unsaved) configurations into the start-up configuration. Doing so insures that in case of an unexpected shutdown, the device will restart with the same configuration it had before the failure.

To use the IP-SLA probe creation tool, simply execute:
//IPSLA_Configuration_DIR/bin/save-running-config.sh

Note: A windows version is also available: *save-running-config.cmd*

Example:

The same Agent Configuration File used to create and remove the probes, can be used to save the running configuration. The "saveRunningConfiguration" field which is optional when creating or removing probes, is mandatory when saving the running configuration. As specified in the Agent Configuration File section, the possible values are 1 for true and 2 for false. Only XML files can be used to save the running configuration.

The following is a list of all options available when using the creation script.

-a (Agent File):

This represents the path of the agent configuration file.

-h (Help):

This option prints a help message. It shows an example of usage and prints the list of options with a small explanation for each one.

-m (Configuration OID Mapping File):

This represents the Configuration OID Mapping File. It is optional.

8

OID Mapping Files

OID mappings are used to associate probes, agents and their attributes to the SNMP OIDs. This allows for more flexibility and more compatibility with all types of agents.

There are two OID Mapping files. One is used to create the probes and the other is used to save the running configuration on an agent.

When creating probes or saving the agent running configuration, these files do not need to be specified. They should be specified only when the default behavior does not work on the agents or when another behavior is desired.

1 Available probe configuration parameters

The list of supported probe configuration parameters is provided in `jAPG\Tools/IP-SLA-Probe-Configuration-CLI/Default/conf/schemas/probeOidMapping.xsd` and `probeTemplate.xsd`.

2 oidProperty

In both mapping configuration files, every element is an oidProperty.

An oidProperty is a representation of a Varbind. A Varbind is a sequence of two fields, an Object ID (OID) and the value for/from that Object ID.

When defining the configurations of an OID Mapping file, oidProperties are put into groups and every oidProperty has an order within the group. Each group is a representation of a Varbind List and the order is actually the order of the Varbind within a Varbind List. Each one of these Varbind List is then used to create one SNMP PDU.

Within both OID Mapping Files, all elements are in alphabetical order. When the configuration files are read, the SNMP sets are not done in alphabetical order but are done by groups and by orders.

Example

```
<sla:confMapping>
  <sla:oidMappingType>defaultAgentType</sla:oidMappingType>
  <sla:copyDestFileType key="copyDestFileType" group="2" order="3" oid="ccCopyDestFileType"
snmpValue="3"/>
  <sla:copyEntryRowStatus key="copyEntryRowStatus" group="1" order="1" oid="
ccCopyEntryRowStatus" snmpValue="6"/>
  <sla:copyEntryRowStatus key="copyEntryRowStatus" group="2" order="1" oid="
ccCopyEntryRowStatus" snmpValue="4"/>
  <sla:copySourceFileType key="copySourceFileType" group="2" order="2" oid="
ccCopySourceFileType" snmpValue="4"/>
</sla:confMapping>
```

In the above example representing the default Agent Configuration OID Mapping, the elements are put in alphabetical order. But the SNMP operations are done in the following order:

A Varbind is built with OID copyEntryRowStatus and value 6. A PDU is created with this Varbind. Then a SNMP set is done with the PDU.

A Varbind is built with OID copyEntryRowStatus and value 4. A Varbind is built with OID copySourceFileType value 4. A Varbind is built with OID copyDestFileType and value 3. Then a PDU with the Varbind List is created. And then the SNMP set is done with the PDU

2.1 oidProperty.xsd

This file is the XML schema that describes the oidProperty. As mentioned earlier, all elements in both OID Mapping files are XSD Complex elements of type oidProperty.

```
<xs:complexType name="oidProperty">
  <xs:attribute name="key" type="xs:string" use="required"/>
  <xs:attribute name="group" type="xs:positiveInteger" use="required" />
  <xs:attribute name="order" type="xs:positiveInteger" use="required" />
  <xs:attribute name="oid" type="xs:string" use="optional"/>
  <xs:attribute name="snmpValue" type="xs:string" use="optional"/>
</xs:complexType>
```

key:

The key represents the oidProperty's name. In most cases, the value of key is the same as the name of the element. Example: sla:copyEntryRowStatus key="copyEntryRowStatus"

group:

The group number can be considered as a unique identifier for a Varbind List. Each Varbind List belongs to one PDU. The SNMP Sets are done with the PDUs in numerical order.

order:

The order can be considered as a unique identifier for a Varbind. Each Varbind is put within a Varbind List in numerical order.

oid:

This represents the Node Name of the OID which is defined in the Management Information Base (MIB) files.

snmpValue:

This represents the default value for an OID. In the case of probes, given that OID is mapped to a probe property, the default value is only used when no value is defined, for that property, in the Probe Configuration File.

2.2 Probe OID Mapping

Each probe attribute is mapped to an SNMP OID. By default, all probes are created using the "Create and GO" method and there are default values for the OIDs that work with most CISCO devices. In case the default values do not work on a device, or in case one should wish to use another method to create the probes, it is possible to change the default mapping by using a different OID Mapping File.

2.3 Agent Configuration OID Mapping

This file is used to save the running configuration. Unlike the probes, only one of the attributes found in the agent configuration file is mapped with the OID file: oidMappingType. Only one configuration exists in the default OID mapping meaning that all running configurations are saved the same way on every type of agent. This also means that all agents, by default, have the same oidMappingType. Of course, if one should wish to save the running configuration by using a different series of SNMP calls, it is possible to do so by creating a different mapping, with a different oidMappingType.