Getting Started with IOS IPS — A Step-by-Step Guide

This guide is divided into two sections. The first section provides a detailed step-by-step process using IOS command-line interface (CLI) to get started in using the IOS IPS 5.x format signatures. It contains the following five steps:

Step 1: Downloading IOS IPS files

Step 2: Creating IOS IPS configuration directory on flash

Step 3: Configuring IOS IPS crypto key

Step 4: Enabling IOS IPS

Step 5: Loading IOS IPS signature package to router

Each step and the specific commands are described below. A section 'Additional Commands and References' under each step provides additional information. An example configuration is displayed in a box below each command.

The second section of the guide provides instructions and examples on advanced options for signature tuning. It contains the following options:

Retire/Unretire signatures

Enable/Disable signatures

Change signature actions

Pre-requisites

Before getting started with the above steps, ensure that you have the following:

- A Cisco Integrated Services Router (87x, 18xx, 28xx, or 38xx)
- 128MB or more DRAM and at least 2MB free flash memory
- Console or telnet connectivity to the router
- IOS Release 12.4(15)T3 or later
- A valid CCO (Cisco.com) login username and password
- A current Cisco IPS Service Contract for licensed signature update services

Section 1: Getting Started Configuration Steps

Step 1: Downloading IOS IPS files

The first step is to download IOS IPS signature package files and public crypto key from Cisco.com.

Step 1.1: Download the required signature files from Cisco.com to your PC

- Location:
  http://tools.cisco.com/support/downloads/go/Model.x?mdfid=281442967&mdfLevel=Software%20Family&treeName=Security&modelName=Cisco%20IOS%20Intrusion%20Prevention%20System%20Feature%20Software&treeMdfId=268438162
- Files to download:

  IOS-Sxxx-CLI.pkg: Signature package – download the latest signature package.

  realm-cisco.pub.key.txt: Public Crypto key – this is the crypto key used by IOS IPS

Step 2: Creating IOS IPS configuration directory on flash

The second step is to create a directory on your router's flash where you store the required signature files and configurations. Alternatively, you can use a Cisco USB flash drive connected to the router's USB port to store the signature files and configurations. The USB flash drive needs to remain connected to the router's USB port if it is used as the IOS IPS configuration directory location. IOS IPS also supports any IOS File System as its configuration location with proper write access.

Step 2.1: To create a directory, enter the following command at the router prompt.

```
mkdir <directory name>
```

```
router#mkdir ips
Create directory filename [ips]?
Created dir flash:ips
```

Additional Commands and References:

To verify the contents of the flash, enter the following command at the router prompt.

```
show flash:
```

```
router#dir flash:
Directory of flash:/

    5  -rw-    51054864   Feb 8 2008 15:46:14 -08:00  c2800nm-advipservicesk9-
mz.124-15.T3.bin
    6  drw-           0  Feb 14 2008 11:36:36 -08:00  ips

64016384 bytes total (12693504 bytes free)
```

To rename the directory name, use the following command example:

```
rename <current name> <new name>
```

```
router#rename ips ips_new
Destination filename [ips_new]?
```

Step 3: Configuring IOS IPS crypto key

The third step is to configure the crypto key used by IOS IPS. This key is located in the realm-cisco.pub.key.txt file that was downloaded in Step 1.

The crypto key is used to verify the digital signature for the master signature file (sigdef-default.xml) whose contents are signed by a Cisco private key to guarantee its authenticity and integrity at every release.

Step 3.1: Open the text file and copy the contents of the file

Step 3.2: Enter 'configure terminal' to enter router configure mode

Step 3.3: Paste the text file content at the '<hostname>(config)#' prompt

Step 3.4: Exit router configuration mode

Step 3.4: Enter the 'show run' command at the router prompt to confirm that the crypto key is configured. You should see the following in the configuration:

```
crypto key pubkey-chain rsa
 named-key realm-cisco.pub signature
  key-string
   30820122 300D0609 2A864886 F70D0101 01050003 82010F00 3082010A 02820101
   00C19E93 A8AF124A D6CC7A24 5097A975 206BE3A2 06FBA13F 6F12CB5B 4E441F16
   17E630D5 C02AC252 912BE27F 37FDD9C8 11FC7AF7 DCDD81D9 43CDABC3 6007D128
   B199ABCB D34ED0F9 085FADC1 359C189E F30AF10A C0EFB624 7E0764BF 3E53053E
   5B2146A9 D7A5EDE3 0298AF03 DED7A5B8 9479039D 20F30663 9AC64B93 C0112A35
   FE3F0C87 89BCB7BB 994AE74C FA9E481D F65875D6 85EAF974 6D9CC8E3 F0B08B85
   50437722 FFBE85B9 5E4189FF CC189CB9 69C46F9C A84DFBA5 7A0AF99E AD768C36
   006CF498 079F88F8 A3B3FB1F 9FB7B3CB 5539E1D1 9693CCBB 551F78D2 892356AE
   2F56D826 8918EF3C 80CA4F4D 87BFCA3B BFF668E9 689782A5 CF31CB6E B4B094D3
   F3020301 0001
  Quit
```

Step 3.5: Save the configuration

```
copy running-configure startup-configure
```

Additional Commands and References:

If the key is configured incorrectly, you need to remove the crypto key first and then reconfigure it.

To remove the key, enter the following commands in the order listed below:

```
router#configure terminal
router(config)#no crypto key pubkey-chain rsa
router(config-pubkey-chain)#no named-key realm-cisco.pub signature
```

```
router(config-pubkey-chain)#exit
router(config)#exit
```

Next verify that the key is removed from the configuration by using 'show run' at router prompt. Last, configure the key again by following Steps 3.1 through 3.5.

Step 4: Enabling IOS IPS

The fourth step is to configure IOS IPS using the following sequence of steps:

Step 4.1: Create a rule name (This will be used on an interface to enable IPS)

```
ip ips name <rule name> < optional ACL>
```

```
router#configure terminal
router(config)# ip ips name iosips
```

You can specify an optional extended or standard access control list (ACL) to filter the traffic that will be scanned by this rule name. All traffic that is permitted by the ACL is subject to inspection by the IPS. Traffic that is denied by the ACL is not inspected by the IPS.

```
router(config)#ip ips name ips list ?
  <1-199>  Numbered access list
  WORD     Named access list
```

Step 4.2: Configure IPS signature storage location, this is the directory 'ips' created in Step 2

```
ip ips config location flash:<directory name>
router(config)#ip ips config location flash:ips
```

Step 4.3: Enable IPS SDEE event notification

```
ip ips notify sdee
```

```
router(config)#ip ips notify sdee
```

To use SDEE, the HTTP server must be enabled (via the 'ip http server' command). If the HTTP server is not enabled, the router cannot respond to the SDEE clients because it cannot see the requests. SDEE notification is disabled by default and must be explicitly enabled.

IOS IPS also supports the use syslog to send event notification. SDEE and syslog can be used independently or enabled at the same time to send IOS IPS event notification. Syslog notification is enabled by default. If logging console is enabled, you will see IPS syslog messages. To enable syslog if it is not enabled:

```
router(config)#ip ips notify log
```

Step 4.4: Configure IOS IPS to use one of the pre-defined signature categories

IOS IPS with Cisco 5.x format signatures operates with signature categories, just like Cisco IPS appliances do. All signatures are pre-grouped into categories and the categories are hierarchical. This is so to help classifying signatures for easy grouping and tuning.

Warning: The “all” signature category contains ALL the signatures in a signature release. Since IOS IPS cannot compile and use all the signatures contained in a signature release at one time, DO NOT UNRETIRE the “all” category, otherwise the router will run out of memory.

Note: When configuring IOS IPS, it is required to FIRST RETIRE all the signatures in the “all” category, and then UNRETIRE selected signature categories.

Note: The order in which the signature categories are configured on the router is also important. IOS IPS will process the category commands in the order listed in the configuration. Some signatures belong to multiple categories, if multiple categories are configured and a signature belongs to more than one of them, the signature’s properties (e.g. retired/unretired, actions, etc.) in the last configured category will be used by IOS IPS.

In the following example, we first retire all the signatures in the “all” category, and then unretire the “IOS IPS Basic” category.

```
router(config)#ip ips signature-category
router(config-ips-category)# category all
router(config-ips-category-action)# retired true
router(config-ips-category-action)# exit
router(config-ips-category)# category ios_ips basic
router(config-ips-category-action)# retired false
router(config-ips-category-action)# exit
router(config-ips-category)# exit
Do you want to accept these changes? [confirm]y
router(config)#
```

Step 4.5: Enable IPS rule on the desired interface and specify the direction the rule will be applied to

```
interface <interface name>
  ip ips <rule name> <in | out>
```

```
router(config)#interface GigabitEthernet 0/1
router(config-if)#ip ips iosips in
router(config-if)#exit
router(config)#exit
router#
```

The direction ‘in’ means only traffic going into the interface will be inspected by IPS, and similarly ‘out’ means only traffic going out the interface will be inspected by IPS. To enable IPS to inspect both in and out traffic of the interface, enter the IPS rule name for ‘in’ and ‘out’ separately on the same interface:

```
router(config)#interface GigabitEthernet 0/1
router(config-if)#ip ips iosips in
router(config-if)#ip ips iosips out
router(config-if)#exit
router(config)#exit
router#
```

Step 5: Loading IOS IPS signature package to the router

The last step is to load the signature package downloaded in Step 1 to the router. The most common way to load the signature package to the router is to use either FTP or TFTP. The following example uses FTP. Please refer to Additional Commands and References section for alternative method for loading IOS IPS signature package.

Note: If using a telnet session, use the ‘terminal monitor’ command to view the console outputs.

Step 5.1: Copy the downloaded signature package from the FTP server to the router

```
copy ftp://<ftp_user:password@Server_IP_address>/<signature_package> idconf
```

Note: Please remember to use the ‘idconf’ parameter at the end of the copy command.

```
router#copy ftp://cisco:cisco@10.1.1.1/IOS-S310-CLI.pkg idconf
Loading IOS-S310-CLI.pkg !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 7608873/4096 bytes]
```

Immediately after the signature package is loaded to the router, signature compiling begins. You can see the logs on the router with logging level 6 or above enabled.

```
*Feb 14 16:44:47 PST: %IPS-6-ENGINE_BUILDS_STARTED:  16:44:47 PST Feb 14 2008
*Feb 14 16:44:47 PST: %IPS-6-ENGINE_BUILDING: multi-string - 8 signatures - 1 of 13
engines
*Feb 14 16:44:47 PST: %IPS-6-ENGINE_READY: multi-string - build time 4 ms - packets
for this engine will be scanned
*Feb 14 16:44:47 PST: %IPS-6-ENGINE_BUILDING: service-http - 622 signatures - 2 of
13 engines
*Feb 14 16:44:53 PST: %IPS-6-ENGINE_READY: service-http - build time 6024 ms -
packets for this engine will be scanned
|
output snipped
|
*Feb 14 16:45:18 PST: %IPS-6-ENGINE_BUILDING: service-smb-advanced - 35 signatures
- 12 of 13 engines
*Feb 14 16:45:18 PST: %IPS-6-ENGINE_READY: service-smb-advanced - build time 16 ms
- packets for this engine will be scanned
*Feb 14 16:45:18 PST: %IPS-6-ENGINE_BUILDING: service-msrpc - 25 signatures - 13 of
13 engines
*Feb 14 16:45:18 PST: %IPS-6-ENGINE_READY: service-msrpc - build time 32 ms -
packets for this engine will be scanned
*Feb 14 16:45:18 PST: %IPS-6-ALL_ENGINE_BUILDS_COMPLETE: elapsed time 31628 ms
```

Step 5.2: Verify the signature package is properly compiled

```
show ip ips signature count
```

```
router#show ip ips signature count
Cisco SDF release version S310.0                ← signature package release
version
Trend SDF release version V0.0

Signature Micro-Engine: multi-string: Total Signatures 8
      multi-string enabled signatures: 8
      multi-string retired signatures: 8
|
outpt snipped
|
Signature Micro-Engine: service-msrpc: Total Signatures 25
      service-msrpc enabled signatures: 25
      service-msrpc retired signatures: 18
      service-msrpc compiled signatures: 1
      service-msrpc inactive signatures - invalid params: 6

Total Signatures: 2136
   Total Enabled Signatures: 807
   Total Retired Signatures: 1779
   Total Compiled Signatures: 351               ← total compiled signatures for
the IOS IPS Basic category
   Total Signatures with invalid parameters: 6
   Total Obsoleted Signatures: 11
router#
```

Additional Commands and References:

If you see an error message during signature compilation such as:

```
%IPS-3-INVALID_DIGITAL_SIGNATURE: Invalid Digital Signature found (key not found
```

It means the public crypto key is invalid. Refer to Step 3 – Configuring IOS IPS Crypto Key to reconfigure the public crypto key.

If there is no access to a FTP or TFTP server, alternatively a USB flash drive can be used to load signature package to the router. First copy the signature package onto the USB drive, then connects it to one of the USB ports on the router, and finally use the ‘copy’ command with the ‘idconf’ parameter to copy the signature package to the router.

```
router#copy usbflash1:IOS-S310-CLI.pkg idconf
```

There are six files in the configured IOS IPS storage directory. These files have a name format of <router-name>-sigdef-xxx.xml or <router-name>-seap-xxx.xml.

```
router#dir ips
Directory of flash:/ips/

    7  -rw-      203419  Feb 14 2008 16:45:24 -08:00  router-sigdef-default.xml
    8  -rw-         271  Feb 14 2008 16:43:36 -08:00  router-sigdef-delta.xml
    9  -rw-        6159  Feb 14 2008 16:44:24 -08:00  router-sigdef-typedef.xml
   10  -rw-       22873  Feb 14 2008 16:44:26 -08:00  router-sigdef-category.xml
   11  -rw-         257  Feb 14 2008 16:43:36 -08:00  router-seap-delta.xml
   12  -rw-         491  Feb 14 2008 16:43:36 -08:00  router-seap-typedef.xml

64016384 bytes total (12693504 bytes free)
router#
```

These files are stored in compressed format and are not editable or viewable directly. The contents of each file are described below:

- **router-sigdef-default.xml:** contains all the factory default signature definitions
- **router-sigdef-delta.xml:** contains signature definitions that have been changed from the default
- **router-sigdef-typedef.xml:** is a file that has all the signature parameter definitions
- **router-sigdef-category.xml:** has all the signature category information, such as category ios_ips basic and advanced
- **router-seap-delta.xml:** contains changes made to the default SEAP parameters
- **router-seap-typedef.xml:** contains all the SEAP parameter definitions

Section II: Advanced Configuration Options

This section provides instructions and examples on advanced IOS IPS options for signature tuning.

Retire/Unretire signatures

Retire/unretire is to select/de-select which signatures are being used by IOS IPS to scan traffic.

Retiring a signature means IOS IPS will NOT compile that signature into memory for scanning.

Unretiring a signature instructs IOS IPS to compile the signature into memory and use the signature to scan traffic.

You can use IOS command-line interface (CLI) to retire or unretire individual signatures, or a group of signatures belong to a signature category in which case all signatures in that category will be retired or unretired.

Note: Some unretired signatures (either unretired as individual signature or within an unretired category) may not compile due to insufficient memory and/or invalid parameters or if the signature has been obsoleted.

The following example shows how to retire individual signature, signature 6130 with subsig ID of 10.

```
router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router(config)#ip ips signature-definition
router(config-sigdef)#signature 6130 10
router(config-sigdef-sig)#status
router(config-sigdef-sig-status)#retired true
router(config-sigdef-sig-status)#exit
router(config-sigdef-sig)#exit
router(config-sigdef)#exit
Do you want to accept these changes? [confirm]y
router(config)#
```

The following example shows how to unretire all signatures belong to the IOS IPS Basic category.

```
router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z
router(config)#ip ips signature-category
router(config-ips-category)# category ios_ips basic
router(config-ips-category-action)#retired false
router(config-ips-category-action)#exit
router(config-ips-category)#exit
Do you want to accept these changes? [confirm]y
```

Note: When signatures in categories other than “IOS IPS Basic” and “IOS IPS Advanced” are unretired as a category, compilation of some signatures/engines could fail. This is due to certain signatures in those categories are not supported by IOS IPS, see example below. All the other successfully compiled (unretired) signatures will be used by IOS IPS to scan traffic.

```
Router(config)#ip ips signature-category
router(config-ips-category)#category os
router(config-ips-category-action)#retired false
router(config-ips-category-action)#exit
router(config-ips-category)#exit
Do you want to accept these changes? [confirm]y


*Feb 14 18:10:46 PST: Applying Category configuration to signatures ...
*Feb 14 18:10:49 PST: %IPS-6-ENGINE_BUILDS_STARTED:  08:10:49 PST Feb 18 2008
*Feb 14 18:10:49 PST: %IPS-6-ENGINE_BUILDING: multi-string - 8 signatures - 1 of 13
engines
*Feb 14 18:10:49 PST: %IPS-6-ENGINE_READY: multi-string - build time 136 ms -
packets for this engine will be scanned
*Feb 14 18:10:49 PST: %IPS-6-ENGINE_BUILDING: service-http - 622 signatures - 2 of
13 engines
*Feb 14 18:10:50 PST: %IPS-4-META_ENGINE_UNSUPPORTED: service-http 5903:1 - this
signature is a component of the unsupported META engine
```

```
*Feb 14 18:24:42 PST: %IPS-4-SIGNATURE_COMPILE_FAILURE: service-http 5754:0 -
compilation of regular expression failed
*Feb 14 18:24:49 PST: %IPS-4-SIGNATURE_COMPILE_FAILURE: service-http 5729:1 -
compilation of regular expression failed
```

Enable/Disable signatures

Enable/disable is to enforce/disregard the action(s) associated with the signatures by IOS IPS when packet or packet flow matches the signatures.

Note: Enable/disable does NOT select/de-select signatures to be used by IOS IPS.

Enabling a signature means that when triggered by a matching packet (or packet flow), the signature takes the appropriate action associated with it. However, only unretired AND successfully compiled signatures will take the action when they are enabled. In other words, if a signature is retired, even though it is enabled, it will not be compiled (because it is retired) and it will not take the action associated with it.

Disabling a signature means that when triggered by a matching packet (or packet flow), the signature DOES NOT take the appropriate action associated with it. In other words, when a signature is disabled, even though it is unretired and successfully compiled, it will not take the action associated with it.

You can use IOS command-line interface (CLI) to enable or disable individual signatures or a group of signatures based on signature categories. The following example shows how to disable signature 6130 with subsig ID of 10.

```
router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router(config)#ip ips signature-definition
router(config-sigdef)#signature 6130 10
router(config-sigdef-sig)#status
router(config-sigdef-sig-status)#enabled false
router(config-sigdef-sig-status)#exit
router(config-sigdef-sig)#exit
router(config-sigdef)#exit
Do you want to accept these changes? [confirm]y
router(config)#
```

The following example shows how to enable all signatures belonging to the IOS IPS Basic category.

```
router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z
router(config)#ip ips signature-category
router(config-ips-category)# category ios_ips basic
router(config-ips-category-action)#enabled true
router(config-ips-category-action)#exit
router(config-ips-category)#exit
Do you want to accept these changes? [confirm]y
router(config)#
```

Change Signature Actions

You can use IOS command-line interface (CLI) to change signature actions for one signature or a group of signatures based on signature categories. The following example shows how to change signature action to alert, drop and reset for signature 6130 with subsig ID of 10.

```
router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router(config)#ip ips signature-definition
router(config-sigdef)#signature 6130 10
router(config-sigdef-sig)#engine
router(config-sigdef-sig-engine)#event-action produce-alert
router(config-sigdef-sig-engine)#event-action deny-packet-inline
router(config-sigdef-sig-engine)#event-action reset-tcp-connection
router(config-sigdef-sig-engine)#exit
router(config-sigdef-sig)#exit
router(config-sigdef)#exit
Do you want to accept these changes? [confirm]y
router(config)#
```

The following example shows how to change event actions for all signatures belonging to signature IOS IPS Basic category.

```
router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z
router(config)#ip ips signature-category
router(config-ips-category)# category ios_ips basic
router(config-ips-category-action)#event-action produce-alert
router(config-ips-category-action)#event-action deny-packet-inline
router(config-ips-category-action)#event-action reset-tcp-connection
router(config-ips-category-action)#exit
router(config-ips-category)#exit
Do you want to accept these changes? [confirm]y
router(config)#
```

Reference

- Cisco IOS IPS on Cisco.com: http://www.cisco.com/go/iosips
- Cisco IOS IPS Signature package: http://tools.cisco.com/support/downloads/go/Model.x?mdfid=281442967&mdfLevel=Software%20Family&treeName=Security&modelName=Cisco%20IOS%20Intrusion%20Prevention%20System%20Feature%20Software&treeMdfId=268438162
- Cisco IOS IPS Configuration Guide: http://www.cisco.com/en/US/products/ps6441/products_feature_guide09186a0080747eb0.html
- Cisco Router and Security Device Manager (SDM): http://www.cisco.com/cgi-bin/tablebuild.pl/sdm
- How to Use CCP to Configure IOS IPS in 12.4(15)T4 and later release: http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6586/ps6634/prod_white_paper0900aecd8066d265.html
- Cisco MySDN: http://tools.cisco.com/MySDN/Intelligence/home.x
- Cisco IDS Event Viewer: http://www.cisco.com/cgi-bin/tablebuild.pl/ids-ev

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV
Amsterdam, The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

Printed in USA C11-390389-04 11/09