



# CHAPTER 5

## Configuring Interfaces

---

This chapter describes how to configure interfaces on the sensor. It contains the following sections:

- [Understanding Interfaces, page 5-1](#)
- [Interface Support, page 5-2](#)
- [Promiscuous Mode, page 5-4](#)
- [Inline Mode, page 5-7](#)
- [Assigning Interfaces to the Virtual Sensor, page 5-8](#)
- [Bypass Mode, page 5-9](#)
- [Configuring Interface Notifications, page 5-10](#)

## Understanding Interfaces

The command and control interface is permanently mapped to a specific physical interface, which depends on the type of sensor you have. You can let the sensing interfaces operate in promiscuous mode, or you can pair the network sensing interfaces into logical interfaces called “inline pairs.” You must enable the interfaces or inline pairs before the sensor can monitor traffic.



### Note

---

On appliances, the sensing interfaces are disabled by default. On modules, the sensing interfaces are always enabled and cannot be disabled.

---

The sensing interface does not have an IP address assigned to it and is therefore invisible to attackers. This lets the sensor monitor the data stream without letting attackers know they are being watched. Promiscuous mode is contrasted by inline technology where all packets entering or leaving the network must pass through the sensor. For more information, see [Understanding Promiscuous Mode, page 5-4](#), and [Understanding Inline Mode, page 5-7](#).

The sensor monitors traffic on interfaces or inline pairs that are assigned to the default virtual sensor. For more information, see [Assigning Interfaces to the Virtual Sensor, page 5-8](#).

To configure the sensor so that traffic continues to flow through inline pairs even when SensorApp is not running, you can enable bypass mode. Bypass mode minimizes dataflow interruptions during reconfiguration, service pack installation, or software failure.

The sensor detects the interfaces of modules that have been installed while the chassis was powered off. You can configure them the next time you start the sensor. If a module is removed, the sensor detects the absence of the interfaces the next time it is started. Your interface configuration is retained, but the sensor ignores it if the interfaces are not present.

The following interface configuration events are reported as status events:

- Link up or down
- Traffic started or stopped
- Bypass mode auto activated or deactivated
- Missed packet percentage threshold exceeded

## Interface Support

Table 5-1 describes the interface support for appliances and modules running IPS 5.0:

**Table 5-1** *Interface Support*

Base Chassis	Added PCI Cards	Interfaces Supporting Inline	Possible Port Combinations	Interfaces Not Supporting Inline
IDS-4210	—	None	N/A	All
IDS-4215	—	None	N/A	All
IDS-4215	4FE	FastEthernet0/1 4FE FastEthernet1/0 FastEthernet1/1 FastEthernet1/2 FastEthernet1/3	0/1<->1/0 0/1<->1/1 0/1<->1/2 0/1<->1/3 1/0<->1/1 1/0<->1/2 1/0<->1/3 1/1<->1/2 1/1<->1/3 1/2<->1/3	FastEthernet0/0
IDS-4235	—	None	N/A	All
IDS-4235	4FE	4FE FastEthernet1/0 FastEthernet1/1 FastEthernet1/2 FastEthernet1/3	1/0<->1/1 1/0<->1/2 1/0<->1/3 1/1<->1/2 1/1<->1/3 1/2<->1/3	GigabitEthernet0/0 GigabitEthernet0/1
IDS-4235	TX (GE)	TX onboard + TX PCI GigabitEthernet0/0 + GigabitEthernet1/0 or GigabitEthernet2/0	0/0<->1/0 0/0<->2/0	GigabitEthernet0/1
IDS-4250	—	None	N/A	All

Table 5-1 Interface Support (continued)

Base Chassis	Added PCI Cards	Interfaces Supporting Inline	Possible Port Combinations	Interfaces Not Supporting Inline
IDS-4250	4FE	4FE FastEthernet1/0 FastEthernet1/1 FastEthernet1/2 FastEthernet1/3	1/0<->1/1 1/0<->1/2 1/0<->1/3 1/1<->1/2 1/1<->1/3 1/2<->1/3	GigabitEthernet0/0 GigabitEthernet0/1
IDS-4250	TX (GE)	TX onboard + TX PCI GigabitEthernet0/0 + GigabitEthernet1/0 or GigabitEthernet2/0	0/0<->1/0 0/0<->2/0	GigabitEthernet0/1
IDS-4250	SX	None	N/A	All
IDS-4250	SX + SX	2 SX GigabitEthernet1/0 GigabitEthernet2/0	1/0<->2/0	GigabitEthernet0/0 GigabitEthernet0/1
IDS-4250	XL	2 SX of the XL GigabitEthernet2/0 GigabitEthernet2/1	2/0<->2/1	GigabitEthernet0/0 GigabitEthernet0/1
IDS-2	—	port 7 and 8 GigabitEthernet0/7 GigabitEthernet0/8	0/7<->0/8	GigabitEthernet0/2
IPS-4240	—	4 onboard GE GigabitEthernet0/0 GigabitEthernet0/1 GigabitEthernet0/2 GigabitEthernet0/3	0/0<->0/1 0/0<->0/2 0/0<->0/3 0/1<->0/2 0/1<->0/3 0/2<->0/3	Management0/0
IPS-4255	—	4 onboard GE GigabitEthernet0/0 GigabitEthernet0/1 GigabitEthernet0/2 GigabitEthernet0/3	0/0<->0/1 0/0<->0/2 0/0<->0/3 0/1<->0/2 0/1<->0/3 0/2<->0/3	Management0/0
NM-CIDS	—	None	N/A	All
AIP-SSM-10	—	GigabitEthernet0/1	By security context	GigabitEthernet0/0
AIP-SSM-20	—	GigabitEthernet0/1	By security context	GigabitEthernet0/0

# Promiscuous Mode

This section describes promiscuous mode on the sensor, and contains the following topics:

- [Understanding Promiscuous Mode, page 5-4](#)
- [Understanding TCP Reset, page 5-4](#)
- [Configuring Promiscuous Mode, page 5-4](#)

## Understanding Promiscuous Mode

In promiscuous mode, packets do not flow through the IPS. The sensor analyzes a copy of the monitored traffic rather than the actual forwarded packet. The advantage of operating in promiscuous mode is that the IPS does not affect the packet flow with the forwarded traffic. The disadvantage of operating in promiscuous mode, however, is the IPS cannot stop malicious traffic from reaching its intended target for certain types of attacks, such as atomic attacks (single-packet attacks). The response actions implemented by promiscuous IPS devices are post-event responses and often require assistance from other networking devices, for example, routers and firewalls, to respond to an attack. While such response actions can prevent some classes of attacks, for atomic attacks, however, the single packet has the chance of reaching the target system before the promiscuous-based sensor can apply an ACL modification on a managed device (such as a firewall, switch, or router).

## Understanding TCP Reset

You need to designate an alternate TCP reset interface in the following situations:

- When a switch is being monitored with either SPAN or VACL capture and the switch does not accept incoming packets on the SPAN or VACL capture port.
- When a switch is being monitored with either SPAN or VACL capture for multiple VLANs, and the switch does not accept incoming packets with 802.1q headers.




---

**Note** The TCP resets need 802.1q headers to tell which VLAN the resets should be sent on.

---

- When a network tap is used for monitoring a connection.




---

**Note** Taps do not allow incoming traffic from the sensor.

---

## Configuring Promiscuous Mode

Use the **physical-interfaces** command in the service interface submode to configure promiscuous interfaces.




---

**Note** AIP-SSM is configured for promiscuous mode from the ASA CLI and not from the IPS CLI. For the procedure, see [Configuring ASA to Send IPS Traffic to AIP-SSM, page 14-3](#).

---

The following options apply:

- **physical-interfaces**—FastEthernet or GigabitEthernet. For a list of possible interfaces for your sensor, see [Interface Support, page 5-2](#).
- **admin-state [enabled | disabled]**—The administrative link state of the interface, whether the interface is enabled or disabled.




---

**Note** On all backplane sensing interfaces on all modules (IDSM-2 NM-CIDS, and AIP-SSM), **admin-state** is set to enabled and is protected (you cannot change the setting). The **admin-state** has no effect (and is protected) on the command and control interface. It only affects sensing interfaces. The command and control interface does not need to be enabled because it cannot be monitored.

---

- **alt-tcp-reset-interface**—Sends TCP resets out an alternate interface when this interface is used for promiscuous monitoring and the reset action is triggered by a signature firing. For more information on when to use the TCP reset interface, see [Understanding TCP Reset, page 5-4](#).




---

**Note** This option is not supported on modules (IDSM-2 NM-CIDS, and AIP-SSM) and appliances that only have one sensing interface (IDS-4210, IDS-4215, IDS-4235, and IDS-4250 without any additional NIC cards).

---

- **interface-name**—The name of the interface on which TCP resets should be sent when this interface is used for promiscuous monitoring and the reset action is triggered by a signature firing. This setting is ignored when this interface is a member of an inline interface.
- **none** —Disables the use of an alternate TCP reset interface. TCP resets triggered by the reset action when in promiscuous mode will be sent out of this interface instead.
- **default**—Sets the value back to the system default setting.
- **description**—Your description of the promiscuous interface.
- **duplex**—The duplex setting of the interface.
  - **auto**—Sets the interface to auto negotiate duplex.
  - **full**—Sets the interface to full duplex.
  - **half**—Sets the interface to half duplex.




---

**Note** The **duplex** option is protected on all modules.

---

- **no**—Remove an entry or selection setting.
- **speed**—The speed setting of the interface.
  - **auto**—Sets the interface to auto negotiate speed.
  - **10**—Sets the interface to 10 MB (for TX interfaces only).
  - **100**—Sets the interface to 100 MB (for TX interfaces only).
  - **1000**—Sets the interface to 1 GB (for Gigabit interfaces only).




---

**Note** The **speed** option is protected on all modules.

---

To configure the promiscuous interface settings on the sensor, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter interface submode:

```
sensor# configure terminal
sensor(config)# service interface
```

**Step 3** Display the list of available interfaces:

```
sensor(config-int)# physical-interfaces ?
GigabitEthernet0/0    GigabitEthernet0/0 physical interface.
GigabitEthernet0/1    GigabitEthernet0/1 physical interface.
GigabitEthernet0/2    GigabitEthernet0/2 physical interface.
GigabitEthernet0/3    GigabitEthernet0/3 physical interface.
Management0/0        Management0/0 physical interface.
sensor(config-int)# physical-interfaces
```

**Step 4** Enable the interface for promiscuous mode:

```
sensor(config-int)# physical-interfaces GigabitEthernet0/2
```

**Step 5** Enable the interface:

```
sensor(config-int-phy)# admin-state enabled
```

The interface must be assigned to the virtual sensor (see [Assigning Interfaces to the Virtual Sensor, page 5-8](#)) and enabled to monitor traffic.

**Step 6** Add a description of this interface:

```
sensor(config-int-phy)# description INT1
```

**Step 7** Configure the duplex settings:

```
sensor(config-int-phy)# duplex full
```

This option is not available on modules.

**Step 8** Configure the speed:

```
sensor(config-int-phy)# speed 1000
```

This option is not available on modules.

**Step 9** Verify the settings:

```
sensor(config-int-phy)# show settings
<protected entry>
name: GigabitEthernet0/2
-----
media-type: tx <protected>
description: INT1 default:
admin-state: enabled default: disabled
duplex: full default: auto
speed: 1000 default: auto
alt-tcp-reset-interface
-----
none
-----
-----
sensor(config-int-phy)#
```

**Step 10** Exit interface submode:

```
sensor(config-int-phy)# exit  
sensor(config-int)# exit  
Apply Changes:[yes]:
```

**Step 11** Press **Enter** to apply the changes or type **no** to discard them.

---

## Inline Mode

This section describes inline mode on the sensor, and contains the following topics:

- [Understanding Inline Mode, page 5-7](#)
- [Interface Support, page 5-2](#)
- [Configuring Inline Mode, page 5-7](#)

## Understanding Inline Mode

Operating in inline mode puts the IPS directly into the traffic flow and affects packet-forwarding rates making them slower by adding latency. An inline IPS sits in the fast-path, which allows the sensor to stop attacks by dropping malicious traffic before it reaches the intended target, thus providing a protective service. Not only is the inline device processing information on layers 3 and 4, but it is also analyzing the contents and payload of the packets for more sophisticated embedded attacks (layers 3 to 7). This deeper analysis lets the system identify and stop and/or block attacks that would normally pass through a traditional firewall device.

In inline mode, a packet comes in through the first interface of the pair of the sensor and out the second interface of the pair. The packet is sent to the second interface of the pair unless that packet is being denied or modified by a signature.



### Note

---

You can configure AIP-SSM to operate inline even though it has only one sensing interface.

---

## Configuring Inline Mode

Use the **inline-interfaces** command in the service interface submode to configure inline interfaces.



### Note

---

AIP-SSM is configured for inline mode from the ASA CLI and not from the IPS CLI. For the procedure, see [Configuring ASA to Send IPS Traffic to AIP-SSM, page 14-3](#).

---

The following options apply:

- **inline-interfaces**—Name of the logical inline interface pair.
- **default**—Sets the value back to the system default setting.
- **description**—Your description of the inline interface pair.
- **interface1**—The first interface in the inline interface pair.

- **interface2**—The second interface in the inline interface pair.
- **no**—Remove an entry or selection setting.

To configure the inline interface settings, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter interface submode:

```
sensor# configure terminal
sensor(config)# service interface
```

**Step 3** Name the inline pair:

```
sensor(config-int)# inline-interfaces PAIR1
```

**Step 4** Configure the two interfaces into a pair:

```
sensor(config-int-inl)# interface1 GigabitEthernet0/0
sensor(config-int-inl)# interface2 GigabitEthernet0/1
```

**Step 5** Add a description of the interface pair:

```
sensor(config-int-inl)# description PAIR1 = Gig0/0 & Gig0/1
```

**Step 6** Verify the settings:

```
sensor(config-int-inl)# show settings
name: PAIR1
-----
description: PAIR1 = Gig0/0 & Gig0/1 default:
interface1: GigabitEthernet0/0
interface2: GigabitEthernet0/1
-----
```

**Step 7** Exit inline interfaces submode:

```
sensor(config-int-inl)# exit
sensor(config-int)# exit
Apply Changes?[yes]:
```

**Step 8** Press **Enter** to apply the changes or type **no** to discard them.

## Assigning Interfaces to the Virtual Sensor

Use the **physical-interface** *interface\_name* command in the service analysis engine submode to assign the interface to the virtual sensor.

You can assign either a physical interface or a logical inline interface pair to the virtual sensor. Make sure that you have created any inline pairs before assigning them to the virtual sensor.

To assign the interface to the virtual sensor, follow these steps:

- 
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter analysis engine mode to assign the interfaces to the virtual sensor:
- ```
sensor# configure terminal
ssensor(config)# service analysis-engine
sensor(config-ana)# virtual-sensor vs0
sensor(config-ana-vir)# physical-interface GigabitEthernet0/1
```
- Step 3** Exit analysis engine mode:
- ```
sensor(config-ana-vir)# exit
sensor(config-ana)# exit
sensor(config)#
Apply Changes:[yes]:
```
- Step 4** Press **Enter** to apply the changes or type **no** to discard them.
- 

## Bypass Mode

This section describes bypass mode on the sensor, and contains the following topics:

- [Understanding Bypass Mode, page 5-9](#)
- [Configuring Bypass Mode, page 5-10](#)

## Understanding Bypass Mode

You can use the bypass mode as a diagnostic tool and a failover protection mechanism. You can set the sensor in a mode where all the IPS processing subsystems are bypassed and traffic is permitted to flow between the inline pairs directly. The bypass mode ensures that packets continue to flow through the sensor when the sensor's processes are temporarily stopped for upgrades or when the sensor's monitoring processes fail. There are three modes: on, off, and automatic. By default, bypass mode is set to automatic.



### Note

Bypass mode was originally intended to only be applicable to inline-paired interfaces. Because of a defect, it does affect promiscuous mode. A future version may address this defect. We recommend you configure bypass mode to automatic or off for promiscuous mode and not use the on mode.



### Caution

There are security consequences when you put the sensor in bypass mode. When bypass mode is on, the traffic bypasses the sensor and is not inspected, therefore, the sensor cannot prevent malicious attacks.



### Note

Bypass mode only functions when the operating system is running. If the sensor is powered off or shut down, bypass mode does not work—traffic is not passed to the sensor.

## Configuring Bypass Mode

Use the **bypass-option** command in the service interface submode to configure bypass mode.

The following options apply:

- **off**—Turns off inline bypassing. Packet inspection will be performed on inline data traffic. However, inline traffic will be interrupted if the analysis engine is stopped.
- **on**—Turns on inline bypassing. No packet inspection will be performed on the traffic. Inline traffic will continue to flow even if the analysis engine is stopped.
- **auto**—Automatically begins bypassing inline packet inspection if the analysis engine stops processing packets. This prevents data interruption on inline interfaces. This is the default.

---

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter interface submode:

```
sensor# configure terminal
sensor(config)# service interface
```

**Step 3** Configure bypass mode:

```
sensor(config-int)# bypass-mode off
```

**Step 4** Verify the settings:

```
sensor(config-int)# show settings
-----
bypass-mode: off default: auto
interface-notifications
-----
missed-percentage-threshold: 0 percent <defaulted>
notification-interval: 30 seconds <defaulted>
idle-interface-delay: 30 seconds <defaulted>
-----
sensor(config-int)#
```

**Step 5** Exit interface submode:

```
sensor(config-int)# exit
Apply Changes:[yes]:
```

**Step 6** Press **Enter** to apply the changes or type **no** to discard them.

---

## Configuring Interface Notifications

You can configure the sensor to monitor the flow of packets across an interface and send notification if that flow changes (starts/stops) during a specified interval. You can configure the missed packet threshold within a specific notification interval and also configure the interface idle delay before a status event is reported.

Use the **interface-notifications** command in the service interface submode to configure traffic notifications.

The following options apply:

- **default**—Sets the value back to the system default setting.
- **idle-interface-delay**—The number of seconds an interface must be idle before sending a notification. The valid range is 5 to 3600. The default is 30 seconds.
- **missed-percentage-threshold**—The percentage of packets that must be missed during a specified interval before notification will be sent. The valid range is 0 to 100. The default is 0.
- **notification-interval**—Interval to check for missed packet percentage. The valid range is 5 to 3600. The default is 30 seconds

To configure the interface notification settings, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter global configuration mode:

```
sensor# configure terminal
```

**Step 3** Enter interface submode:

```
sensor(config)# service interface
```

**Step 4** Enter interface notifications submode:

```
sensor(config-int)# interface-notifications
```

**Step 5** Configure the idle interface delay:

```
sensor(config-int-int)# idle-interface-delay 60
```

**Step 6** Configure the missed percentage threshold:

```
sensor(config-int-int)# missed-percentage-threshold 1
```

**Step 7** Configure the notification interval:

```
sensor(config-int-int)# notification-interval 60
```

**Step 8** Verify the settings:

```
sensor(config-int-int)# show settings
interface-notifications
-----
missed-percentage-threshold: 1 percent default: 0
notification-interval: 60 seconds default: 30
idle-interface-delay: 60 seconds default: 30
-----
sensor(config-int-int)#
```

**Step 9** Exit interface notifications submode:

```
sensor(config-int-int)# exit
sensor(config-int)# exit
Apply Changes:[yes]:
```

**Step 10** Press **Enter** to apply the changes or type **no** to discard them.

---

