Configuring Attack Response Controller for Blocking and Rate Limiting

This chapter provides information for setting up the ARC to perform blocking and rate limiting on the sensor. It the following sections:

Blocking Notes and Caveats

The following notes and caveats apply to blocking:

- The ARC is formerly known as Network Access Controller. Although the name has been changed, the IDM, the IME, and the CLI contain references to Network Access Controller, **nac**, and **network-access**.
- Blocking is not supported on the FWSM in multiple mode admin context.
- Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

- Do not confuse blocking with the ability of the sensor to drop packets. The sensor can drop packets when the following actions are configured for a sensor in inline mode: deny packet inline, deny connection inline, and deny attacker inline.
- The ACLs that ARC makes should never be modified by you or any other system. These ACLs are temporary and new ACLs are constantly being created by the sensor. The only modifications that you can make are to the Pre- and Post-Block ACLs.
- Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.
- Two sensors cannot control blocking or rate limiting on the same device. If this situation is needed, configure one sensor as the master blocking sensor to manage the devices and the other sensors can forward their requests to the master blocking sensor.
- Pre-Block and Post-Block ACLS do not apply to rate limiting.
- When you add a master blocking sensor, you reduce the number of blocking devices per sensor. For example, if you want to block on 10 security appliances and 10 routers with one blocking interface/direction each, you can assign 10 to the sensor and assign the other 10 to a master blocking sensor.
- While blocking is disabled, the ARC continues to receive blocks and track the time on active blocks, but will not apply new blocks or remove blocks from the managed devices. After blocking is reenabled, the blocks on the devices are updated.
- We recommend that you do not permit the sensor to block itself, because it may stop communicating with the blocking device. You can configure this option if you can ensure that if the sensor creates a rule to block its own IP address, it will not prevent the sensor from accessing the blocking device.
- You MUST create a user profile before configuring the blocking device.

Understanding Blocking

The ARC is responsible for managing network devices in response to suspicious events by blocking access from attacking hosts and networks. The ARC blocks the IP address on the devices it is managing. It sends the same block to all the devices it is managing, including any other master blocking sensors. The ARC monitors the time for the block and removes the block after the time has expired.

The ARC completes the action response for a new block in no more than 7 seconds. In most cases, it completes the action response in less time. To meet this performance goal, you should not configure the sensor to perform blocks at too high a rate or to manage too many blocking devices and interfaces. We recommend that the maximum number of blocks not exceed 250 and the maximum number of blocking items not exceed 10. To calculate the maximum number of blocking items, a security appliance counts as one blocking item per blocking context. A router counts as one blocking item per blocking interface/direction. A switch running Catalyst software counts as one blocking item per blocking VLAN. If the recommended limits are exceeded, the ARC may not apply blocks in a timely manner or may not be able to apply blocks at all.

Caution Blocking is not supported on the FWSM in multiple mode admin context.

For security appliances configured in multi-mode, Cisco IPS does not include VLAN information in the block request. Therefore you must make sure the IP addresses being blocked are correct for each security appliance. For example, the sensor is monitoring packets on a security appliance customer context that

is configured for VLAN A, but is blocking on a different security appliance customer context that is configured for VLAN B. Addresses that trigger blocks on VLAN A may refer to a different host on VLAN B.

There are three types of blocks:

- Host block—Blocks all traffic from a given IP address.
- Connection block—Blocks traffic from a given source IP address to a given destination IP address and destination port. Multiple connection blocks from the same source IP address to either a different destination IP address or destination port automatically switch the block from a connection block to a host block.
- Network block—Blocks all traffic from a given network. You can initiate host and connection blocks manually or automatically when a signature is triggered. You can only initiate network blocks manually.

Note Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

Caution Do not confuse blocking with the ability of the sensor to drop packets. The sensor can drop packets when the following actions are configured for a sensor in inline mode: deny packet inline, deny connection inline, and deny attacker inline.

For automatic blocks, you must configure **request-block-host** or **request-block-connection** as the event action for particular signatures, and add them to any event action overrides you have configured, so that the SensorApp sends a block request to the ARC when the signature is triggered. When the ARC receives the block request from the SensorApp, it updates the device configurations to block the host or connection.

On Cisco routers and Catalyst 6500 series switches, ARC creates blocks by applying ACLs or VACLs. ACLs and VACLs permit or deny passage of data packets through interface directions or VLANs. Each ACL or VACL contains permit and deny conditions that apply to IP addresses. The security appliances do not use ACLs or VACLs. The built-in **shun** and **no shun** command is used.

Caution The ACLs that ARC makes should never be modified by you or any other system. These ACLs are temporary and new ACLs are constantly being created by the sensor. The only modifications that you can make are to the Pre- and Post-Block ACLs.

You need the following information for the ARC to manage a device:

- Login user ID (if the device is configured with AAA).
- Login password.
- Enable password (not needed if the user has enable privileges).
- Interfaces to be managed (for example, ethernet0, vlan100).
- Any existing ACL or VACL information you want applied at the beginning (Pre-Block ACL or VACL) or end (Post-Block ACL or VACL) of the ACL or VACL that will be created. This does not apply to the security appliances because they do not use ACLs to block.
- Whether you are using Telnet or SSH to communicate with the device.
- IP addresses (host or range of hosts) you never want blocked.

- How long you want the blocks to last.

Tip To check the status of the ARC, type **show statistics network-access** at the `sensor#`. The output shows the devices you are managing, any active blocks and rate limits, and the status of all devices..

Note Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.

For More Information

- For the procedure to add request-block-host or request-block-connection event actions to a signature, see Assigning Actions to Signatures, page 7-20.
- For the procedure for configuring overrides that add the **request-block-host** or **request-block-connection** event actions to alerts of a specific risk rating, see Adding, Editing, Enabling, and Disabling Event Action Overrides, page 8-18.
- For more information on Pre- and Post-Block ACLs, see How the Sensor Manages Devices, page 14-21.

Understanding Rate Limiting

The ARC is responsible for rate limiting traffic in protected networks. Rate limiting lets sensors restrict the rate of specified traffic classes on network devices. Rate limit responses are supported for the Host Flood and Net Flood engines, and the TCP half-open SYN signature. The ARC can configure rate limits on network devices running Cisco IOS 12.3 or later. Master blocking sensors can also forward rate limit requests to blocking forwarding sensors.

To add a rate limit, you specify the following:

- Source address and/or destination address for any rate limit
- Source port and/or destination port for rate limits with TCP or UDP protocol

You can also tune rate limiting signatures. You must also set the action to **request-rate-limit** and set the percentage for these signatures.

Note Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.

Table 14-1 lists the supported rate limiting signatures and parameters.

Table 14-1 Rate Limiting Signatures

Signature ID	Signature Name	Protocol	Destination IP Address Allowed	Data
2152	ICMP Flood Host	ICMP	Yes	echo-request
2153	ICMP Smurf Attack	ICMP	Yes	echo-reply

Table 14-1 Rate Limiting Signatures (continued)

Signature ID	Signature Name	Protocol	Destination IP Address Allowed	Data
4002	UDP Flood Host	UDP	Yes	none
6901	Net Flood ICMP Reply	ICMP	No	echo-reply
6902	Net Flood ICMP Request	ICMP	No	echo-request
6903	Net Flood ICMP Any	ICMP	No	None
6910	Net Flood UDP	UDP	No	None
6920	Net Flood TCP	TCP	No	None
3050	TCP HalfOpenSyn	TCP	No	halfOpenSyn

Tip To check the status of the ARC, type **show statistics network-access** at the `sensor#`. The output shows the devices you are managing, any active blocks and rate limits, and the status of all devices..

For More Information

- For the procedure for configuring rate limiting on a router, see Configuring Blocking and Rate Limiting Devices, page 14-21.
- For the procedure for configuring a sensor to be a master blocking sensor, see Configuring the Sensor to be a Master Blocking Sensor, page 14-28.

Understanding Service Policies for Rate Limiting

You must not apply a service policy to an interface/direction that is configured for rate limiting. If you do so, the rate limit action will fail. Before configuring rate limits, confirm that there is no service policy on the interface/direction, and remove it if one exists. The ARC does not remove the existing rate limit unless it is one that the ARC had previously added.

Rate limits use ACLs, but not in the same way as blocks. Rate limits use **acls** and **class-map** entries to identify traffic, and **policy-map** and **service-policy** entries to police the traffic.

Before Configuring ARC

Caution Two sensors cannot control blocking or rate limiting on the same device. If this situation is needed, configure one sensor as the master blocking sensor to manage the devices and the other sensors can forward their requests to the master blocking sensor.

Note When you add a master blocking sensor, you reduce the number of blocking devices per sensor. For example, if you want to block on 10 security appliances and 10 routers with one blocking interface/direction each, you can assign 10 to the sensor and assign the other 10 to a master blocking sensor.

Before you configure the ARC for blocking or rate limiting, make sure you do the following:

- Analyze your network topology to understand which devices should be blocked by which sensor, and which addresses should never be blocked.
- Gather the usernames, device passwords, enable passwords, and connections types (Telnet or SSH) needed to log in to each device.
- Know the interface names on the devices.
- Know the names of the Pre-Block ACL or VACL and the Post-Block ACL or VACL if needed.
- Understand which interfaces should and should not be blocked and in which direction (in or out). You do not want to accidentally shut down an entire network.

For More Information

For the procedure for configuring the master blocking sensor, see Configuring the Sensor to be a Master Blocking Sensor, page 14-28.

Supported Devices

Caution If the recommended limits are exceeded, the ARC may not apply blocks in a timely manner or may not be able to apply blocks at all.

By default, the ARC supports up to 250 devices in any combination. The following devices are supported for blocking by the ARC:

- Cisco series routers using Cisco IOS 11.2 or later (ACLs):
  - Cisco 1600 series router
  - Cisco 1700 series router
  - Cisco 2500 series router
  - Cisco 2600 series router
  - Cisco 2800 series router
  - Cisco 3600 series router
  - Cisco 3800 series router
  - Cisco 7200 series router
  - Cisco 7500 series router
- Catalyst 5000 switches with RSM with IOS 11.2(9)P or later (ACLs)
- Catalyst 6500 switches and 7600 routers with IOS 12.1(13)E or later (ACLs)
- Catalyst 6500 switches 7600 routers with Catalyst software version 7.5(1) or later (VACLs)
  - Supervisor Engine 1A with PFC
  - Supervisor Engine 1A with MSFC1
  - Supervisor Engine 1A with MFSC2
  - Supervisor Engine 2 with MSFC2
  - Supervisor Engine 720 with MSFC3

Note We support VACL blocking on the Supervisor Engine and ACL blocking on the MSFC.

- PIX Firewall with version 6.0 or later (**shun** command)
  - 501
  - 506E
  - 515E
  - 525
  - 535
- ASA with version 7.0 or later (**shun** command)
  - ASA 5510
  - ASA 5520
  - ASA 5540
- FWSM 1.1 or later (**shun** command)

You configure blocking using either ACLs, VACLS, or the **shun** command. All firewall and ASA models support the **shun** command.

The following devices are supported for rate limiting by the ARC:

- Cisco series routers using Cisco IOS 12.3 or later:
  - Cisco 1700 series router
  - Cisco 2500 series router
  - Cisco 2600 series router
  - Cisco 2800 series router
  - Cisco 3600 series router
  - Cisco 3800 series router
  - Cisco 7200 series router
  - Cisco 7500 series router

Caution The ARC cannot perform rate limits on 7500 routers with VIP. The ARC reports the error but cannot rate limit.

Configuring Blocking Properties

You can change the default blocking properties. It is best to use the default properties, but if you need to change them, use the following procedures:

- Allowing the Sensor to Block Itself, page 14-8
- Disabling Blocking, page 14-9
- Specifying Maximum Block Entries, page 14-11
- Specifying the Block Time, page 14-13
- Enabling ACL Logging, page 14-14

Allowing the Sensor to Block Itself

Caution We recommend that you do not permit the sensor to block itself, because it may stop communicating with the blocking device. You can configure this option if you can ensure that if the sensor creates a rule to block its own IP address, it will not prevent the sensor from accessing the blocking device.

Use the **allow-sensor-block {true | false}** command in the service network access submode to configure the sensor to block itself. To allow the sensor to block itself, follow these steps:

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

Step 2 Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
```

Step 3 Enter general submode.

```
sensor(config-net)# general
```

Step 4 Configure the sensor to block itself. By default, this value is false.

```
sensor(config-net-gen)# allow-sensor-block true
```

Step 5 Verify the settings.

```
sensor(config-net-gen)# show settings
   general
   ------------------------------------------------
      log-all-block-events-and-errors: true <defaulted>
      enable-nvram-write: false <defaulted>
      enable-acl-logging: false <defaulted>
      allow-sensor-block: true default: false
      block-enable: true default: true
      block-max-entries: 100 default: 250
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0)
      ------------------------------------------------
      ------------------------------------------------
      never-block-hosts (min: 0, max: 250, current: 1)
      ------------------------------------------------
         ip-address: 192.0.2.1
         ------------------------------------------
      ------------------------------------------------
      never-block-networks (min: 0, max: 250, current: 1)
      ------------------------------------------------
         ip-address: 209.165.200.224/27
         ---------------------------------------------
      ------------------------------------------------
      block-hosts (min: 0, max: 250, current: 0)
      ------------------------------------------------
--MORE--
```

Step 6 Configure the sensor not to block itself.

```
sensor(config-net-gen)# allow-sensor-block false
```

Step 7 Verify the setting.

```
sensor(config-net-gen)# show settings
   general
   -----------------------------------------------
      log-all-block-events-and-errors: true <defaulted>
      enable-nvram-write: false <defaulted>
      enable-acl-logging: false <defaulted>
      allow-sensor-block: false default: false
      block-enable: true default: true
      block-max-entries: 100 default: 250
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0)
      -----------------------------------------------
      -----------------------------------------------
      never-block-hosts (min: 0, max: 250, current: 1)
      -----------------------------------------------
         ip-address: 192.0.2.1
         -------------------------------------------
      -----------------------------------------------
      never-block-networks (min: 0, max: 250, current: 1)
      -----------------------------------------------
         ip-address: 209.165.200.224/27
         -------------------------------------------
      -----------------------------------------------
      block-hosts (min: 0, max: 250, current: 0)
      -----------------------------------------------
--MORE--
```

Step 8 Exit network access submode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:?[yes]:
```

Step 9 Press **Enter** to apply the changes or enter **no** to discard them.

Disabling Blocking

Note For blocking to operate, you must set up devices to do the blocking.

Use the **block-enable {true | false}** command in the service network access submode to enable or disable blocking on the sensor. By default, blocking is enabled on the sensor. If the ARC is managing a device and you need to manually configure something on that device, you should disable blocking first. You want to avoid a situation in which both you and the ARC could be making a change at the same time on the same device. This could cause the device and/or the ARC to crash.

Caution If you disable blocking for maintenance on the devices, make sure you enable it after the maintenance is complete or the network will be vulnerable to attacks that would otherwise be blocked.

Note While blocking is disabled, the ARC continues to receive blocks and track the time on active blocks, but will not apply new blocks or remove blocks from the managed devices. After blocking is reenabled, the blocks on the devices are updated.

To disable blocking or rate limiting, follow these steps:

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

Step 2 Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

Step 3 Enter general submode.

```
sensor(config-net)# general
```

Step 4 Disable blocking on the sensor. By default, this value is set to true.

```
sensor(config-net-gen)# block-enable false
```

Step 5 Verify the settings.

```
sensor(config-net-gen)# show settings
   general
   ------------------------------------------------
      log-all-block-events-and-errors: true <defaulted>
      enable-nvram-write: false <defaulted>
      enable-acl-logging: false <defaulted>
      allow-sensor-block: false default: false
      block-enable: false default: true
      block-max-entries: 100 default: 250
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0)
      ------------------------------------------------
      ------------------------------------------------
      never-block-hosts (min: 0, max: 250, current: 1)
      ------------------------------------------------
         ip-address: 192.0.2.1
         ---------------------------------------------
      ------------------------------------------------
      never-block-networks (min: 0, max: 250, current: 1)
      ------------------------------------------------
         ip-address: 209.165.200.224/27
         ---------------------------------------------
      ------------------------------------------------
      block-hosts (min: 0, max: 250, current: 0)
      ------------------------------------------------
--MORE--
```

Step 6 Enable blocking on the sensor.

```
sensor(config-net-gen)# block-enable true
```

Step 7 Verify that the setting has been returned to the default.

```
sensor(config-net-gen)# show settings
   general
   ------------------------------------------------
      log-all-block-events-and-errors: true <defaulted>
      enable-nvram-write: false <defaulted>
```

```
        enable-acl-logging: false <defaulted>
        allow-sensor-block: false default: false
        block-enable: true default: true
        block-max-entries: 100 default: 250
        max-interfaces: 250 <defaulted>
        master-blocking-sensors (min: 0, max: 100, current: 0)
        ------------------------------------------------
        ------------------------------------------------
        never-block-hosts (min: 0, max: 250, current: 1)
        ------------------------------------------------
           ip-address: 192.0.2.1
           ---------------------------------------------
        ------------------------------------------------
        never-block-networks (min: 0, max: 250, current: 1)
        ------------------------------------------------
           ip-address: 209.165.200.224/27
           ---------------------------------------------
        ------------------------------------------------
        block-hosts (min: 0, max: 250, current: 0)
        ------------------------------------------------
--MORE--
```

Step 8 Exit network access submode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:?[yes]:
```

Step 9 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

- For the procedure for configuring the sensor to manage Cisco routers, see Configuring the Sensor to Manage Cisco Routers, page 14-23.
- For the procedure for configuring the sensor to manage Cisco routers and switches, seeConfiguring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers, page 14-25.

Specifying Maximum Block Entries

Caution We do not recommend setting the maximum block entries higher than 250. Some devices have problems with larger numbers of ACL or shun entries. Refer to the documentation for each device to determine its limits before increasing this number.

Note The number of blocks will not exceed the maximum block entries. If the maximum is reached, new blocks will not occur until existing blocks time out and are removed.

Use the **block-max-entries** command in the service network access submode to configure the maximum block entries. You can set how many blocks are to be maintained simultaneously (1 to 65535). The default value is 250.

To change the maximum number of block entries, follow these steps:

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

Step 2 Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

Step 3 Enter general submode.

```
sensor(config-net)# general
```

Step 4 Change the maximum number of block entries.

```
sensor(config-net-gen)# block-max-entries 100
```

Step 5 Verify the setting.

```
sensor(config-net-gen)# show settings
   general
   -----------------------------------------------
      log-all-block-events-and-errors: true <defaulted>
      enable-nvram-write: false <defaulted>
      enable-acl-logging: false <defaulted>
      allow-sensor-block: false default: false
      block-enable: true <defaulted>
      block-max-entries: 100 default: 250
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0)
      -----------------------------------------------
      -----------------------------------------------
      never-block-hosts (min: 0, max: 250, current: 1)
      -----------------------------------------------
         ip-address: 192.0.2.1
         -----------------------------------------------
      -----------------------------------------------
      never-block-networks (min: 0, max: 250, current: 1)
      -----------------------------------------------
         ip-address: 209.165.200.224/27
         -----------------------------------------------
      -----------------------------------------------
      block-hosts (min: 0, max: 250, current: 0)
      -----------------------------------------------
--MORE--
```

Step 6 Return to the default value of 250 blocks.

```
sensor(config-net-gen)# default block-max-entries
```

Step 7 Verify the setting.

```
sensor(config-net-gen)# show settings
   general
   -----------------------------------------------
      log-all-block-events-and-errors: true <defaulted>
      enable-nvram-write: false <defaulted>
      enable-acl-logging: false <defaulted>
      allow-sensor-block: false default: false
      block-enable: true <defaulted>
      block-max-entries: 250 <defaulted>
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0
      -----------------------------------------------
```

```
      ------------------------------------------------
      never-block-hosts (min: 0, max: 250, current: 1)
      ------------------------------------------------
         ip-address: 192.0.2.1
         ---------------------------------------------
      ------------------------------------------------
      never-block-networks (min: 0, max: 250, current: 1)
      ------------------------------------------------
         ip-address: 209.165.200.224/27
         ---------------------------------------------
      ------------------------------------------------
      block-hosts (min: 0, max: 250, current: 0)
      ------------------------------------------------
--MORE--
```

Step 8 Exit network access submode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:?[yes]:
```

Step 9 Press **Enter** to apply the changes or enter **no** to discard them.

Specifying the Block Time

Note If you change the default block time, you are changing a signature parameter, which affects all signatures.

Note The time for manual blocks is set when you request the block.

Use the **global-block-timeout** command in the service event action rules submode to change the amount of time an automatic block lasts. The default is 30 minutes.

To change the default block time, follow these steps:

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

Step 2 Enter event action rules submode.

```
sensor# configure terminal
sensor(config)# service event-action-rules rules0
sensor(config-rul)#
```

Step 3 Enter general submode.

```
sensor(config-rul)# general
```

Step 4 Specify the block time. The value is the time duration of the block event in minutes (0 to 10000000).

```
sensor(config-rul-gen)# global-block-timeout 60
```

Step 5 Verify the setting.

```
sensor(config-rul-gen)# show settings
   general
```

```
   -----------------------------------------------
      global-overrides-status: Enabled <defaulted>
      global-filters-status: Enabled <defaulted>
      global-summarization-status: Enabled <defaulted>
      global-metaevent-status: Enabled <defaulted>
      global-deny-timeout: 3600 <defaulted>
      global-block-timeout: 60 default: 30
      max-denied-attackers: 10000 <defaulted>
   -----------------------------------------------
sensor(config-rul-gen)#
```

Step 6 Exit event action rules submode.

```
sensor(config-rul-gen)# exit
sensor(config-rul)# exit
Apply Changes:?[yes]:
```

Step 7 Press **Enter** to apply the changes or enter **no** to discard them.

Note There is a time delay while the signatures are updated.

Enabling ACL Logging

Use the **enable-acl-logging {true | false}** command in the service network access submode to enable ACL logging, which causes ARC to append the log parameter to block entries in the ACL or VACL. This causes the device to generate syslog events when packets are filtered. Enable ACL logging only applies to routers and switches. The default is disabled.

To enable ACL logging, follow these steps:

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

Step 2 Enter network access submode:

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

Step 3 Enter general submode.

```
sensor(config-net)# general
```

Step 4 Enable ACL logging.

```
sensor(config-net-gen)# enable-acl-logging true
```

Step 5 Verify that ACL logging is enabled.

```
sensor(config-net-gen)# show settings
   general
   -----------------------------------------------
      log-all-block-events-and-errors: true <defaulted>
      enable-nvram-write: false <defaulted>
      enable-acl-logging: true default: false
      allow-sensor-block: false <defaulted>
      block-enable: true <defaulted>
      block-max-entries: 250 <defaulted>
```

```
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0)
      -----------------------------------------------
```

Step 6 Disable ACL logging by using the **false** keyword.

```
sensor(config-net-gen)# enable-acl-logging false
```

Step 7 Verify that ACL logging is disabled.

```
sensor(config-net-gen)# show settings
   general
   -----------------------------------------------
      log-all-block-events-and-errors: true <defaulted>
      enable-nvram-write: false <defaulted>
      enable-acl-logging: false default: false
      allow-sensor-block: false <defaulted>
      block-enable: true <defaulted>
      block-max-entries: 250 <defaulted>
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0)
      -----------------------------------------------
```

Step 8 Exit network access mode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:?[yes]:
```

Step 9 Press **Enter** to apply the changes or enter **no** to discard them.

Enabling Writing to NVRAM

Use the **enable-nvram-write {true | false}** command to configure the sensor to have the router write to NVRAM when ARC first connects. If enable-nvram-write is enabled, NVRAM is written each time the ACLs are updated. The default is disabled.

Enabling NVRAM writing ensures that all changes for blocking are written to NVRAM. If the router is rebooted, the correct blocks will still be active. If NVRAM writing is disabled, a short time without blocking occurs after a router reboot. And not enabling NVRAM writing increases the life of the NVRAM and decreases the time for new blocks to be configured.

To enable writing to NVRAM, follow these steps:

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

Step 2 Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

Step 3 Enter general submode.

```
sensor(config-net)# general
```

Step 4 Enable writing to NVRAM.

```
sensor(config-net-gen)# enable-nvram-write true
```

Step 5 Verify that writing to NVRAM is enabled.

```
sensor(config-net-gen)# show settings
   general
   -----------------------------------------------
      log-all-block-events-and-errors: true <defaulted>
      enable-nvram-write: true default: false
      enable-acl-logging: false default: false
      allow-sensor-block: false <defaulted>
      block-enable: true <defaulted>
      block-max-entries: 250 <defaulted>
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0)
      -----------------------------------------------
```

Step 6 Disable writing to NVRAM.

```
sensor(config-net-gen)# enable-nvram-write false
```

Step 7 Verify that writing to NVRAM is disabled.

```
sensor(config-net-gen)# show settings
   general
   -----------------------------------------------
      log-all-block-events-and-errors: true <defaulted>
      enable-nvram-write: false default: false
      enable-acl-logging: false default: false
      allow-sensor-block: false <defaulted>
      block-enable: true <defaulted>
      block-max-entries: 250 <defaulted>
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0)
      -----------------------------------------------
```

Step 8 Exit network access submode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:?[yes]:
```

Step 9 Press **Enter** to apply the changes or enter **no** to discard them.

Logging All Blocking Events and Errors

Use the **log-all-block-events-and-errors {true | false}** command in the service network access submode to configure the sensor to log events that follow blocks from start to finish. For example, when a block is added to or removed from a device, an event is logged. You may not want all these events and errors to be logged. Disabling **log-all-block-events-and-errors** suppresses the new events and errors. The default is enabled.

To disable blocking event and error logging, follow these steps:

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

Step 2 Enter network access mode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

Step 3 Enter general submode.

```
sensor(config-net)# general
```

Step 4 Disable blocking event and error logging.

```
sensor(config-net-gen)# log-all-block-events-and-errors false
```

Step 5 Verify that logging is disabled.

```
sensor(config-net-gen)# show settings
   general
   ----------------------------------------------
      log-all-block-events-and-errors: false default: true
      enable-nvram-write: false default: false
      enable-acl-logging: false default: false
      allow-sensor-block: false <defaulted>
      block-enable: true <defaulted>
      block-max-entries: 250 <defaulted>
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0)
      ----------------------------------------------
```

Step 6 Enable blocking event and error logging.

```
sensor(config-net-gen)# log-all-block-events-and-errors true
```

Step 7 Verify that logging is enabled.

```
sensor(config-net-gen)# show settings
   general
   ----------------------------------------------
      log-all-block-events-and-errors: true default: true
      enable-nvram-write: false default: false
      enable-acl-logging: false default: false
      allow-sensor-block: false <defaulted>
      block-enable: true <defaulted>
      block-max-entries: 250 <defaulted>
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0)
      ----------------------------------------------
```

Step 8 Exit network access mode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:?[yes]:
```

Step 9 Press **Enter** to apply the changes or type **no** to discard them.

Configuring the Maximum Number of Blocking Interfaces

Use the **max-interfaces** command to configure the maximum number of interfaces for performing blocks. For example, a PIX Firewall counts as one interface. A router with one interface counts as one, but a router with two interfaces counts as two. At most you can configure 250 blocking interfaces on a router, switch, or firewall. You can configure up to 250 Catalyst 6K switches, 250 routers, and 250 firewalls.

The **max-interfaces** command configures the limit of the sum total of all interfaces and devices. In addition to configuring the limit on the sum total of interfaces and devices, there is a fixed limit on the number of blocking interfaces you can configure per device. Use the **show settings** command in network access mode to view the specific maximum limits per device.

To configure the maximum number of blocking interfaces, follow these steps:

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

Step 2 Enter network access mode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

Step 3 Enter general submode.

```
sensor(config-net)# general
```

Step 4 Specify the maximum number of interfaces.

```
sensor(config-net-gen)# max-interfaces 50
```

Step 5 Verify the number of maximum interfaces.

```
sensor(config-net-gen)# show settings
   general
   ----------------------------------------------
      log-all-block-events-and-errors: true default: true
      enable-nvram-write: false default: false
      enable-acl-logging: false default: false
      allow-sensor-block: false <defaulted>
      block-enable: true <defaulted>
      block-max-entries: 250 <defaulted>
      max-interfaces: 50 default: 250
      master-blocking-sensors (min: 0, max: 100, current: 0)
      ----------------------------------------------
```

Step 6 Return the setting to the default of 250.

```
sensor(config-net-gen)# default max-interfaces
```

Step 7 Verify the default setting.

```
sensor(config-net-gen)# show settings

   general
   ----------------------------------------------
      log-all-block-events-and-errors: true default: true
      enable-nvram-write: false default: false
      enable-acl-logging: false default: false
      allow-sensor-block: false <defaulted>
      block-enable: true <defaulted>
      block-max-entries: 250 <defaulted>
      max-interfaces: 250 <defaulted>
      master-blocking-sensors (min: 0, max: 100, current: 0)
      ----------------------------------------------
```

Step 8 Exit network access mode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:?[yes]:
```

Step 9 Press **Enter** to apply the changes or enter **no** to discard them.

Configuring Addresses Never to Block

Use the **never-block-hosts** and the **never-block-networks** commands in the service network access submode to configure hosts and network that should never be blocked.

The following commands apply:

- *ip_address*—Specifies the IP address of the device that should never be blocked.
- *ip_address/netmask*—Specifies the IP address of the network that should never be blocked. The format is A.B.C.D/nn.

You must tune your sensor to identify hosts and networks that should never be blocked, not even manually, because you may have a trusted network device whose normal, expected behavior appears to be an attack. Such a device should never be blocked, and trusted, internal networks should never be blocked. You can specify a single host or an entire network.

Note The **never-block-hosts** and the **never-block-networks** commands apply only to the Request Block Host and Request Block Connection event actions. It does not apply to the Deny Attacker Inline, Deny Connection Inline, or Deny Packet Inline event actions. Use event action rules to filter out the hosts that you do not want blocked, denied, or dropped.

Configuring Addresses Never to Be Blocked

To set up addresses never to be blocked by blocking devices, follow these steps:

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

Step 2 Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

Step 3 Enter general submode.

```
sensor(config-net)# general
```

Step 4 Specify the address that should never be blocked:

- For a single host

```
sensor(config-net-gen)# never-block-hosts 192.0.2.1
```

- For a network

```
sensor(config-net-gen)# never-block-networks 209.165.200.224/27
```

Step 5 Verify the settings.

```
sensor(config-net-gen)# show settings
   general
   -----------------------------------------------
      log-all-block-events-and-errors: true <defaulted>
      enable-nvram-write: false <defaulted>
      enable-acl-logging: false <defaulted>
```

```
        allow-sensor-block: false default: false
        block-enable: true default: true
        block-max-entries: 100 default: 250
        max-interfaces: 250 <defaulted>
        master-blocking-sensors (min: 0, max: 100, current: 0)
        ----------------------------------------------
        ----------------------------------------------
        never-block-hosts (min: 0, max: 250, current: 2)
        ----------------------------------------------
           ip-address: 192.0.2.1
           ---------------------------------------------
        ----------------------------------------------
        never-block-networks (min: 0, max: 250, current: 2)
        ----------------------------------------------
           ip-address: 209.165.200.224/27
--MORE--
```

Step 6 Exit network access submode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:?[yes]:
```

Step 7 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

For the procedure for configuring event action filters, see Configuring Event Action Filters, page 8-21.

Configuring User Profiles

Note If the username or password is not needed to log in to the device, do not set a value for it.

Note You MUST create a user profile before configuring the blocking device.

Use the **user-profiles** *profile_name* command in the service network access submode to set up user profiles for the other devices that the senor will manage. The user profiles contain userid, password, and enable password information. For example, routers that all share the same passwords and usernames can be under one user profile.

To set up user profiles, follow these steps:

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

Step 2 Enter network access mode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

Step 3 Create the user profile name.

```
sensor(config-net)# user-profiles PROFILE1
```

Step 4 Enter the username for that user profile.

```
sensor(config-net-use)# username username
```

Step 5 Specify the password for the user.

```
sensor(config-net-use)# password
Enter password[]: ********
Re-enter password ********
```

Step 6 Specify the enable password for the user.

```
sensor(config-net-use)# enable-password
Enter enable-password[]: ********
Re-enter enable-password ********
```

Step 7 Verify the settings.

```
sensor(config-net-use)# show settings
   profile-name: PROFILE1
   -----------------------------------------------
      enable-password: <hidden>
      password: <hidden>
      username: jsmith default:
   -----------------------------------------------
sensor(config-net-use)#
```

Step 8 Exit network access submode.

```
sensor(config-net-use)# exit
sensor(config-net)# exit
Apply Changes:?[yes]:
```

Step 9 Press **Enter** to apply the changes or enter **no** to discard them.

Configuring Blocking and Rate Limiting Devices

This section describes how to configure devices that the sensor uses to perform blocking or rate limiting. It contains the following topics:

How the Sensor Manages Devices

Note ACLs do not apply to rate limiting devices.

The ARC uses ACLs on Cisco routers and switches to manage those devices. These ACLs are built as follows:

1. A **permit** line with the sensor IP address or, if specified, the NAT address of the sensor.

   Note If you permit the sensor to be blocked, this line does not appear in the ACL.

2. Pre-Block ACL (if specified). This ACL must already exist on the device.

   Note The ARC reads the lines in the ACL and copies these lines to the beginning of the ACL.

3. Any active blocks.
4. Either specify a Post-Block ACL, which must already exist on the device, or specify **permit ip any any** (do not use if a Post-Block ACL is specified). The ARC reads the lines in the ACL and copies these lines to the end of the ACL.

   Note Make sure the last line in the ACL is **permit ip any any** if you want all unmatched packets to be permitted.

The ARC uses two ACLs to manage devices. Only one is active at any one time. It uses the offline ACL name to build the new ACL, then applies it to the interface. The ARC then reverses the process on the next cycle.

Caution The ACLs that the ARC makes should never be modified by you or any other system. These ACLs are temporary and new ACLs are constantly being created by the sensor. The only modifications that you can make are to the Pre- and Post-Block ACLs.

If you need to modify the Pre-Block or Post-Block ACL, do the following:

1. Disable blocking on the sensor.
2. Make the changes to the configuration of the device.
3. Reenable blocking on the sensor.

When blocking is reenabled, the sensor reads the new device configuration.

Caution A single sensor can manage multiple devices, but you cannot use multiple sensors to control a single device. In this case, use a master blocking sensor.

For More Information

- For the procedure for enabling blocking, see Configuring Blocking Properties, page 14-7.
- For the procedure for configuring the sensor to be a master blocking sensor, see Configuring the Sensor to be a Master Blocking Sensor, page 14-28.

Configuring the Sensor to Manage Cisco Routers

This section describes how to configure the sensor to manage Cisco routers. It contains the following topics:

Routers and ACLs

Note Pre-Block and Post-Block ACLS do not apply to rate limiting.

You create and save Pre-Block and Post-Block ACLs in your router configuration. These ACLs must be extended IP ACLs, either named or numbered. See your router documentation for more information on creating ACLs. Enter the names of these ACLs that are already configured on your router in the Pre-Block ACL and Post-Block ACL fields.

The Pre-Block ACL is mainly used for permitting what you do not want the sensor to ever block. When a packet is checked against the ACL, the first line that gets matched determines the action. If the first line matched is a permit line from the Pre-Block ACL, the packet is permitted even though there may be a deny line (from an automatic block) listed later in the ACL. The Pre-Block ACL can override the deny lines resulting from the blocks.

The Post-Block ACL is best used for additional blocking or permitting that you want to occur on the same interface or direction. If you have an existing ACL on the interface or direction that the sensor will manage, that existing ACL can be used as a Post-Block ACL. If you do not have a Post-Block ACL, the sensor inserts **permit ip any any** at the end of the new ACL.

When the sensor starts up, it reads the contents of the two ACLs. It creates a third ACL with the following entries:

- A **permit** line for the sensor IP address
- Copies of all configuration lines of the Pre-Block ACL
- A **deny** line for each address being blocked by the sensor
- Copies of all configuration lines of the Post-Block ACL

The sensor applies the new ACL to the interface and direction that you designate.

Note When the new ACL is applied to an interface or direction of the router, it removes the application of any other ACL to that interface or direction.

Configuring the Sensor to Manage Cisco Routers

To configure a sensor to manage a Cisco router to perform blocking and rate limiting, follow these steps:

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

Step 2 Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
```

```
sensor(config-net)#
```

Step 3 Specify the IP address for the router controlled by the ARC.

```
sensor(config-net)# router-devices ip_address
```

Step 4 Enter the logical device name that you created when you configured the user profile. The ARC accepts anything you enter. It does not check to see if the user profile exists.

```
sensor(config-net-rou)# profile-name user_profile_name
```

Step 5 Specify the method used to access the sensor. If unspecified, SSH 3DES is used.

```
sensor(config-net-rou)# communication {telnet | ssh-3des}
```

Note If you are using 3DES, you must use the command **ssh host-key** *ip_address* to accept the key or ARC cannot connect to the device.

Step 6 Specify the sensor NAT address.

```
sensor(config-net-rou)# nat-address nat_address
```

Note This changes the IP address in the first line of the ACL from the address of the sensor to the NAT address. This is not a NAT address configured on the device being managed. It is the address the sensor is translated to by an intermediate device, one that is between the sensor and the device being managed.

Step 7 Specify whether the router will perform blocking, rate limiting, or both.

Note The default is blocking. You do not have to configure response capabilities if you want the router to perform blocking only.

a. Rate limiting only

```
sensor(config-net-rou)# response-capabilities rate-limit
```

b. Both blocking and rate limiting

```
sensor(config-net-rou)# response-capabilities block|rate-limit
```

Step 8 Specify the interface name and direction.

```
sensor(config-net-rou)# block-interfaces interface_name {in | out}
```

Caution The name of the interface must either be the complete name of the interface or an abbreviation that the router recognizes with the **interface** command.

Step 9 (Optional) Add the pre-ACL name (blocking only).

```
sensor(config-net-rou-blo)# pre-acl-name pre_acl_name
```

Step 10 (Optional) Add the post-ACL name (blocking only).

```
sensor(config-net-rou-blo)# post-acl-name post_acl_name
```

Step 11 Verify the settings.

```
sensor(config-net-rou-blo)# exit
sensor(config-net-rou)# show settings
   ip-address: 192.0.2.1
   -----------------------------------------------
      communication: ssh-3des default: ssh-3des
      nat-address: 19.89.149.219 default: 0.0.0.0
      profile-name: PROFILE1
      block-interfaces (min: 0, max: 100, current: 1)
      -----------------------------------------------
         interface-name: GigabitEthernet0/1
         direction: in
         -----------------------------------------------
            pre-acl-name: <defaulted>
            post-acl-name: <defaulted>
         -----------------------------------------------
      -----------------------------------------------
      response-capabilities: block|rate-limit default: block
   -----------------------------------------------
sensor(config-net-rou)#
```

Step 12 Exit network access submode.

```
sensor(config-net-rou)# exit
sensor(config-net)# exit
sensor(config)# exit
Apply Changes:?[yes]:
```

Step 13 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

- For the procedure for configuring user profiles, see Configuring User Profiles, page 14-20.
- For the procedure for adding a device to the known hosts list, see Adding Hosts to the SSH Known Hosts List, page 4-47.

Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers

This section describes how to configure the sensor to manage Cisco switches. It contains the following topics:

Switches and VACLs

You can configure the ARC to block using VACLs on the switch itself when running Cisco Catalyst software, or to block using router ACLs on the MSFC or on the switch itself when running Cisco IOS software. This section describes blocking using VACLs. You cannot configure switches that use VACLs to perform rate limiting. You must configure the blocking interfaces on the Catalyst 6500 series switch and specify the VLAN of traffic you want blocked.

You create and save Pre-Block and Post-Block VACLs in your switch configuration. These VACLs must be extended IP VACLs, either named or numbered. See your switch documentation for more information on creating VACLs. Enter the names of these VACLs that are already configured on your switch in the Pre-Block VACL and Post-Block VACL fields.

The Pre-Block VACL is used mainly for permitting what you do not want the sensor to ever block. When a packet is checked against the VACL, the first line that gets matched determines the action. If the first line matched is a permit line from the Pre-Block VACL, the packet is permitted even though there may be a deny line (from an automatic block) listed later in the VACL. The Pre-Block VACL can override the deny lines resulting from the blocks.

The Post-Block VACL is best used for additional blocking or permitting that you want to occur on the same VLAN. If you have an existing VACL on the VLAN that the sensor will manage, the existing VACL can be used as a Post-Block VACL. If you do not have a Post-Block V ACL, the sensor inserts **permit ip any any** at the end of the new VACL.

When the sensor starts up, it reads the contents of the two VACLs. It creates a third VACL with the following entries:

- A **permit** line for the sensor IP address
- Copies of all configuration lines of the Pre-Block VACL
- A **deny** line for each address being blocked by the sensor
- Copies of all configuration lines of the Post-Block VACL

The sensor applies the new VACL to the VLAN that you designate.

Note When the new VACL is applied to a VLAN of the switch, it removes the application of any other VACL to that VLAN.

For More Information

For the procedure for configuring blocking using router ACLs, see Configuring Blocking and Rate Limiting Devices, page 14-21.

Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers

To configure the sensor to manage Catalyst 6500 series switches and Cisco 7600 series routers, follow these steps:

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

Step 2 Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

Step 3 Specify the IP address for the router controlled by the ARC.

```
sensor(config-net)# cat6k-devices ip_address
```

Step 4 Enter the user profile name that you created when you configured the user profile. The ARC accepts anything you type. It does not accept it, check to see if the logical device exists.

```
sensor(config-net-cat)# profile-name user_profile_name
```

Step 5 Specify the method used to access the sensor. If unspecified, SSH 3DES is used.

```
sensor(config-net-cat)# communication {telnet | ssh-3des}
```

Note If you are using 3DES, you must use the command **ssh host-key** *ip_address* to accept the key or ARC cannot connect to the device.

Step 6 Specify the sensor NAT address.

```
sensor(config-net-cat)# nat-address nat_address
```

Note This changes the IP address in the first line of the ACL from the IP address of the sensor to the NAT address. This is not a NAT address configured on the device being managed. It is the address the sensor is translated to by an intermediate device, one that is between the sensor and the device being managed.

Step 7 Specify the VLAN number.

```
sensor(config-net-cat)# block-vlans vlan_number
```

Step 8 (Optional) Add the pre-VACL name.

```
sensor(config-net-cat-blo)# pre-vacl-name pre_vacl_name
```

Step 9 (Optional) Add the post-VACL name.

```
sensor(config-net-cat-blo)# post-vacl-name post_vacl_name
```

Step 10 Exit network access submode.

```
sensor(config-net-cat-blo)# exit
sensor(config-net-cat)# exit
sensor(config-net)# exit
sensor(config)# exit
Apply Changes:?[yes]:
```

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

For More Information

- For the procedure for configuring user profiles, see Configuring User Profiles, page 14-20.
- For the procedure for adding a device to the known hosts list, see Adding Hosts to the SSH Known Hosts List, page 4-47.

Configuring the Sensor to Manage Cisco Firewalls

To configure the sensor to manage Cisco firewalls, follow these steps:

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

Step 2 Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
```

```
sensor(config-net)#
```

Step 3 Specify the IP address for the firewall controlled by the ARC.

```
sensor(config-net)# firewall-devices ip_address
```

Step 4 Enter the user profile name that you created when you configured the user profile. ARC accepts anything you type. It does not check to see if the logical device exists.

```
sensor(config-net-fir)# profile-name user_profile_name
```

Step 5 Specify the method used to access the sensor. If unspecified, SSH 3DES is used.

```
sensor(config-net-fir)# communication {telnet | ssh-3des}
```

Note If you are using 3DES, you must use the command **ssh host-key** *ip_address* to accept the key or the ARC cannot connect to the device.

Step 6 Specify the sensor NAT address.

```
sensor(config-net-fir)# nat-address nat_address
```

Note This changes the IP address in the first line of the ACL from the IP address of the sensor to the NAT address. This is not a NAT address configured on the device being managed. It is the address the sensor is translated to by an intermediate device, one that is between the sensor and the device being managed.

Step 7 Exit network access submode.

```
sensor(config-net-fir)# exit
sensor(config-net)# exit
sensor(config)# exit
Apply Changes:?[yes]:
```

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

For More Information

- For the procedure for configuring user profiles, see Configuring User Profiles, page 14-20.
- For the procedure for adding a device to the known hosts list, see Adding Hosts to the SSH Known Hosts List, page 4-47.

Configuring the Sensor to be a Master Blocking Sensor

Multiple sensors (blocking forwarding sensors) can forward blocking requests to a specified master blocking sensor, which controls one or more devices. The master blocking sensor is the ARC running on a sensor that controls blocking on one or more devices on behalf of one or more other sensors. The ARC on a master blocking sensor controls blocking on devices at the request of the ARCs running on other sensors. Master blocking sensors can also forward rate limits.

Caution Two sensors cannot control blocking or rate limiting on the same device. If this situation is needed, configure one sensor as the master blocking sensor to manage the devices and the other sensors can forward their requests to the master blocking sensor.

When you add a master blocking sensor, you reduce the number of blocking devices per sensor. For example, if you want to block on 10 firewalls and 10 routers with one blocking interface/direction each, you can assign 10 to the sensor and assign the other 10 to a master blocking sensor.

On the blocking forwarding sensor, identify which remote host serves as the master blocking sensor; on the master blocking sensor you must add the blocking forwarding sensors to its access list.

If the master blocking sensor requires TLS for web connections, you must configure the ARC of the blocking forwarding sensor to accept the X.509 certificate of the master blocking sensor remote host. Sensors by default have TLS enabled, but you can change this option.

Note Typically the master blocking sensor is configured to manage the network devices. Blocking forwarding sensors are not normally configured to manage other network devices, although doing so is permissible.

Even if you have no devices configured for blocking or rate limiting, a sensor that is configured for blocking or rate limiting can forward blocking and rate limiting requests to a master blocking sensor. When a signature fires that has blocking or rate limit requests configured as event actions, the sensor forwards the block or rate limit request to the master blocking sensor, which then performs the block or rate limit.

Caution Only one sensor should control all blocking interfaces on a device.

Use the **master-blocking-sensors** *master_blocking_sensor_ip_address* command in the service network access submode to configure a master blocking sensor.

The following commands apply:

- *master_blocking_sensor_ip_address*—Specifies the IP address of sensor for forward block requests.
- **password**—Specifies the account password of sensor for forward block requests.
- **port**—Specifies the port of sensor for forward block requests.
- **tls {true | false}** —Set to true if the remote sensor requires TLS; otherwise, set to false.
- **username**—Specifies the account name of sensor for forward block requests.

Configuring the Master Blocking Sensor

To configure ARC on a sensor to forward blocks to a master blocking sensor, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges on both the master blocking sensor and the blocking forwarding sensor.

Step 2 Enter configuration mode on both sensors.

```
sensor# configure terminal
```

Step 3 Configure TLS if necessary:

a. On the master blocking sensor, check to see if it requires TLS and what port number is used. If `enable-tls` is true, go to Step b.

```
sensor(config)# service web-server
sensor(config-web)# show settings
   enable-tls: true <defaulted>
   port: 443 <defaulted>
   server-id: HTTP/1.1 compliant <defaulted>
sensor(config-web)#
```

b. On the blocking forwarding sensor, configure it to accept the X.509 certificate of the master blocking sensor.

```
sensor(config-web)# exit
sensor(config)# tls trusted-host ip-address master_blocking_sensor_ip_address port
port_number
```

Example

```
sensor(config)# tls trusted-host ip-address 192.0.2.1 port 8080
Certificate MD5 fingerprint is
F4:4A:14:BA:84:F4:51:D0:A4:E2:15:38:7E:77:96:D8Certificate SHA1 fingerprint is
84:09:B6:85:C5:43:60:5B:37:1E:6D:31:6A:30:5F:7E:4D:4D:E8:B2
Would you like to add this to the trusted certificate table for this host?[yes]:
```

Note You are prompted to accept the certificate based on the certificate fingerprint. Sensors provide only self-signed certificates (instead of certificates signed by a recognized certificate authority). You can verify the master blocking sensor host sensor certificate by logging in to the host sensor and typing the **show tls fingerprint** command to see that the fingerprints of the host certificate match.

Step 4 Enter **yes** to accept the certificate from the master blocking sensor.

Step 5 Enter network access mode.

```
sensor(config)# service network-access
```

Step 6 Enter general submode.

```
sensor(config-net)# general
```

Step 7 Add a master blocking sensor entry.

```
sensor(config-net-gen)# master-blocking-sensors master_blocking_sensor_ip_address
```

Step 8 Specify the username for an administrative account on the master blocking sensor host.

```
sensor(config-net-gen-mas)# username username
```

Step 9 Specify the password for the user.

```
sensor(config-net-gen-mas)# password
Enter password []: *****
Re-enter mbs-password []: *****
sensor(config-net-gen-mas)#
```

Step 10 Specify the port number for the host HTTP communications. The default is 80/443 if not specified.

```
sensor(config-net-gen-mas)# port port_number
```

Step 11 Specify whether or not the host uses TLS/SSL.

```
sensor(config-net-gen-mas)# tls {true | false}
sensor(config-net-gen-mas)
```

Note If you set the value to true, you need to use the command **tls trusted-host ip-address** *master_blocking_sensor_ip_address*.

Step 12 Exit network access submode.

```
sensor(config-net-gen-mas)# exit
sensor(config-net-gen)# exit
sensor(config-net)# exit
sensor(config)# exit
Apply Changes:?[yes]:
```

Step 13 Press **Enter** to apply the changes or enter **no** to discard them.

Step 14 On the master blocking sensor, add the block forwarding sensor IP address to the access list.

For More Information

For the procedure for adding the blocking forward sensor IP address to the access list, see Changing the Access List, page 4-6.

Configuring Host Blocking

Note Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

Use the **block host** *ip-address* [**timeout** *minutes*] command in privileged EXEC mode to block a host. Use the **no** form of the command to remove a block on a host. You must have blocking configured before you can set up host blocks. You can also view a list of hosts that are being blocked. If you do not configure the amount of time for the host block, it is permanent.

The following commands apply:

- *ip-address*—Specifies the IP address of the host to be blocked.
- *minutes*—(Optional) Specifies the duration of the host block in minutes. The valid range is 0 to 70560 minutes.

Blocking a Host

To block a host, follow these steps:

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

Step 2 Configure the host block for 15 minutes, for example. The host block ends in 15 minutes.

```
sensor# block host 192.0.2.1 timeout 15
```

Step 3 Start a host block. The host block lasts until you remove it.

```
sensor# block host 192.0.2.1
```

Step 4 End the host block.

```
sensor# no block host 192.0.2.1
sensor#
```

Configuring Network Blocking

Note Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

Use the **block network** *ip-address/netmask* [**timeout** *minutes*] command in privileged EXEC mode to block a network. Use the **no** form of the command to remove a block on a network. You must have blocking configured before you can set up network blocks. You can also view a list of networks that are being blocked. If you do not configure the amount of time for the network block, it is permanent.

The following commands apply:

- *ip-address/netmask*—Specifies the network subnet to be blocked in *X.X.X.X/nn* format, where *X.X.X.X* specifies the sensor IP address as a 32-bit address written as 4 octets separated by periods where *X* = 0-255, and *nn* specifies the number of bits (1032) in the netmask.
- *minutes*—(Optional) Specifies the duration of the network block in minutes. The valid range is 0 to 70560 minutes.

Blocking a Network

To block a network, follow these steps:

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

Step 2 Configure the network block for 15 minutes, for example. The network block ends in 15 minutes.

```
sensor# block network 192.0.2.0/24 timeout 15
```

Step 3 Start a network block. The network block lasts until you remove it.

```
sensor# block network 192.0.2.0/24
```

Step 4 End the network block.

```
sensor# no block network 192.0.2.0/24
sensor#
```

Configuring Connection Blocking

Note Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

Use the **block connection** *source-ip-address destination-ip-address* [**port** *port-number*] [**protocol** *type*] [**timeout** *minutes*] command in privileged EXEC mode to block a connection between two IP addresses. Use the **no** form of the command to remove the connection block. You must have blocking configured before you can set up connection blocks. You can also view a list of connections that are being blocked. If you do not configure the amount of time for the connection block, it is permanent.

The following commands apply:

- *source-ip-address*—Specifies the source IP address in a connection block.
- *destination-ip-address*—Specifies the destination IP address in a connection block.
- *port-number*—(Optional) Specifies the destination port number. The valid range is 0 to 65535.
- *type*—(Optional) Specifies the protocol type. The valid types are **tcp** or **udp**.
- *minutes*—(Optional) Specifies the duration of the connection block in minutes. The valid range is 0 to 70560 minutes.

Blocking a Connection

To block a connection, follow these steps:

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

Step 2 Configure the connection block between a source IP address and a destination IP address specifying the port, protocol, and time, for example. The connection block ends in 30 minutes.

```
sensor# block connection 10.0.0.0 172.16.0.0 port 80 protocol tcp timeout 30
```

Step 3 Start a connection block. The connection block lasts until you remove it.

```
sensor# block connection 10.0.0.0 172.16.0.0
```

Step 4 End the connection block.

```
sensor# no block connection 10.0.0.0
sensor#
```

Obtaining a List of Blocked Hosts and Connections

Use the **show statistics** command to obtain a list of blocked hosts and blocked connections.

To obtain a list of blocked hosts and connections, follow these steps:

Step 1 Log in to the CLI.

Step 2 Check the statistics for the ARC. The `Host` entry indicates which hosts are being blocked and how long the blocks are.

```
sensor# show statistics network-access
Current Configuration
   LogAllBlockEventsAndSensors = true
   EnableNvramWrite = false
   EnableAclLogging = false
   AllowSensorBlock = false
   BlockMaxEntries = 250
   MaxDeviceInterfaces = 250
   NetDevice
      Type = Cisco
      IP = 10.1.1.1
      NATAddr = 0.0.0.0
      Communications = telnet
      BlockInterface
         InterfaceName = fa0/0
         InterfaceDirection = in
State
   BlockEnable = true
   NetDevice
      IP = 10.1.1.1
      AclSupport = uses Named ACLs
      Version = 12.2
      State = Active
   BlockedAddr
      Host
         IP = 192.168.1.1
         Vlan =
         ActualIp =
         BlockMinutes = 80
         MinutesRemaining = 76
```