



CHAPTER 6

Configuring Event Action Rules

This chapter explains how to configure event action rules. It contains the following sections:

- [About Event Action Rules, page 6-1](#)
- [Signature Event Action Processor, page 6-2](#)
- [Event Actions, page 6-3](#)
- [Task List for Configuring Event Action Rules, page 6-4](#)
- [Event Action Variables, page 6-4](#)
- [Calculating the Risk Rating, page 6-6](#)
- [Configuring Target Value Ratings, page 6-7](#)
- [Event Action Overrides, page 6-7](#)
- [Configuring Event Action Overrides, page 6-8](#)
- [Event Action Filters, page 6-9](#)
- [General Settings, page 6-14](#)
- [Event Action Rules Example, page 6-19](#)

About Event Action Rules

Event action rules are a group of settings you configure for the event action processing component of the sensor. These rules dictate the actions the sensor performs when an event occurs.

The event action processing component is responsible for the following functions:

- Calculating the risk rating
- Adding event action overrides
- Filtering event action
- Executing the resulting event action
- Summarizing and aggregating events
- Maintaining a list of denied attackers

Signature Event Action Processor

SEAP coordinates the data flow from the signature event in the alarm channel to processing through the SEAO, the SEAF, and the SEAH. It consists of the following components:

- Alarm channel
The unit that represents the area to communicate signature events from the Sensor App inspection path to signature event handling.
- Signature event action override (SEAO)
Adds actions based on the RR value. SEAO applies to all signatures that fall into the range of the configured RR threshold. Each SEAO is independent and has a separate configuration value for each action type. For more information, see [Calculating the Risk Rating, page 6-6](#).
- Signature event action filter (SEAF)
Subtracts actions based on the signature event's SIGID, addresses, and RR. The input to the SEAF is the signature event with actions possibly added by the SEAO.



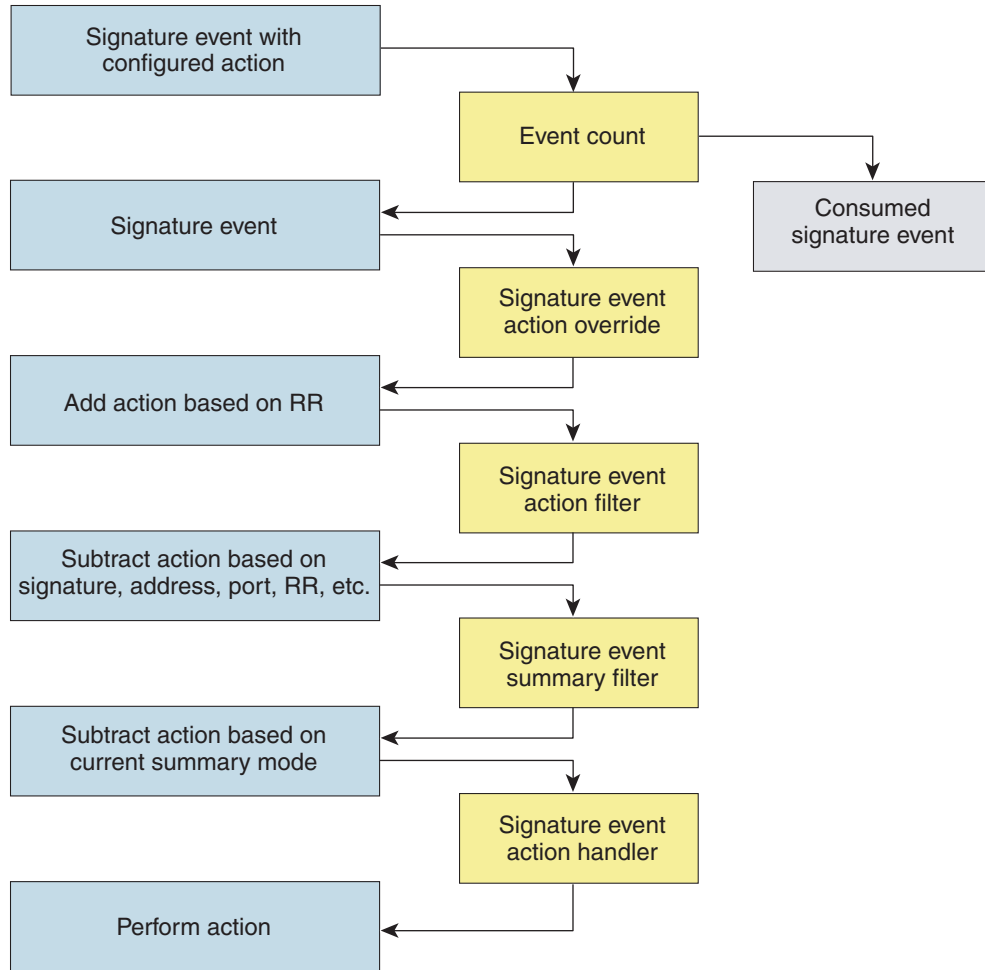
Note The SEAF can only subtract actions, it cannot add new actions.

The following parameters apply to the SEAF:

- Signature ID
- Subsignature ID
- Attacker address
- Attacker port
- Victim address
- Victim port
- RR threshold range
- Actions to subtract
- Sequence identifier (optional)
- Stop-or-continue bit
- Enable action filter line bit
- Signature event action handler (SEAH)
Performs the requested actions. The output from the SEAH is the actions being performed and possibly an evIdsAlert written to the Event Store.

[Figure 6-1 on page 6-3](#) illustrates the logical flow of the signature event through the SEAP and the operations performed on the action for this event. It starts with the signature event with configured action received in the alarm channel and flows top-to-bottom as the signature event passes through the functional components of the SEAP.

Figure 6-1 Signature Event Through SEAP



132188

Event Actions

Table 6-1 describes the event actions.

Table 6-1 Event Actions

Event Action Name	Description
Produce Alert	Writes the event to the Event Store as an evIdsAlert.
Produce Verbose Alert	Includes an encoded dump of the offending packet in the evIdsAlert.
Deny Attacker Inline	Does not transmit this packet and future packets originating from the attacker address for a specified period of time (inline mode only).
Deny Connection Inline	Does not transmit this packet and future packets on the TCP flow (inline mode only).
Deny Packet Inline	Does not transmit this packet (inline only).

Table 6-1 Event Actions (continued)

Event Action Name	Description
Log Attacker Packets	Starts IP logging packets containing the attacker address.
Log Pair Packets	Starts IP logging packets containing the attacker-victim address pair.
Log Victim Packets	Starts IP logging packets containing the victim address.
Reset TCP Connection	Sends TCP resets to hijack and terminate the TCP flow.
Request SNMP Trap	Sends request to Notification App to perform SNMP notification.
Request Block Connection	Sends request to Network Access Controller to block this connection.
Request Block Host	Sends request to Network Access Controller to block this host (the attacker).

**Caution**

The Produce Alert action is not automatic when you enable alerts for a signature. To have an alert created in the Event Store, you must select Produce Alert. If you add a second action, you must include Produce Alert if you want an alert sent to the Event Store. Also, every time you configure the event actions, a new list is created and it replaces the old list. Make sure you include all the event actions you need for each signature.

Task List for Configuring Event Action Rules

Follow these steps when configuring the event action rules component of the IPS:

1. Create any variables that you want to use in event action filters.
2. Create TVRs.

Assign TVRs to your network assets so that you can calculate the RR.
3. Create overrides to add actions based on the RR value.

Assign an RR to each event action type.
4. Create filters.

Assign filters to subtract actions based on the signature's SIGID, IP addresses, and RR.
5. Configure the general settings.

Specify whether you want to use the summarizer, the meta event generator, or configure denied attacker parameters.

Event Action Variables

This section describes event action variables, and contains the following topics:

- [About Event Action Variables, page 6-5](#)
- [Configuring Event Action Variables, page 6-5](#)

About Event Action Variables

You can create event action variables and then use those variables in event action filters. When you want to use the same value within multiple filters, use a variable. When you change the value of the variable, any filter that uses that variable is updated with the new value.



Note

You must preface the variable with a dollar (\$) sign to indicate that you are using a variable rather than a string.

Some variables cannot be deleted because they are necessary to the signature system. If a variable is protected, you cannot edit it. You receive an error message if you try to delete protected variables. You can edit only one variable at a time.

When configuring IP addresses, specify the full IP address or ranges or set of ranges. For example:

- 10.90.1.1
- 10.89.10.10-10.89.10.23
- 10.1.1.1-10.2.255.255, 10.89.10.10-10.89.10.23



Timesaver

For example, if you have an IP address space that applies to your engineering group and there are no Windows systems in that group, and you are not worried about any Windows-based attacks to that group, you could set up a variable to be the engineering group's IP address space. You could then use this variable to configure a filter that would ignore all Windows-based attacks for this group.

Configuring Event Action Variables

Use the **variables** *variable_name* **address** *ip_address* command in service event action rules submode to set up event action variables. The IP address can be one address, a range, or ranges separated by a comma.

To configure event action variables, 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
```

Step 3 Create a variable:

```
sensor(config-rul)# variables variable1 address 10.89.130.108
```

The valid values for **address** are A.B.C.D-A.B.C.D [,A.B.C.D-A.B.C.D].

Step 4 Check the variable you just made:

```
sensor(config-rul)# show settings
variables (min: 0, max: 256, current: 2)
```

```
-----
variableName: variable1
```

```
-----
address: 10.89.130.108 default: 0.0.0.0-255.255.255.255
-----
```

Step 5 Exit event action rules submode:

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

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

Calculating the Risk Rating

An RR is a value between 0 and 100 that represents a numerical quantification of the risk associated with a particular event on the network. The calculation takes into account the value of the network asset being attacked (for example, a particular server), so it is configured on a per-signature basis (ASR and SFR) and on a per-server basis (TVR).

RRs let you prioritize alerts that need your attention. These RR factors take into consideration the severity of the attack if it succeeds, the fidelity of the signature, and the overall value of the target host to you. The RR is reported in the `evIdsAlert`.

The following values are used to calculate the RR for a particular event:

- **Attack Severity Rating**—A weight associated with the severity of a successful exploit of the vulnerability.

The ASR is derived from the alert severity parameter of the signature.

- **Signature Fidelity Rating**—A weight associated with how well this signature might perform in the absence of specific knowledge of the target.

SFR is calculated by the signature author on a per-signature basis. The signature author defines a baseline confidence ranking for the accuracy of the signature in the absence of qualifying intelligence on the target. It represents the confidence that the detected behavior would produce the intended effect on the target platform if the packet under analysis were allowed to be delivered. For example, a signature that is written with very specific rules (specific Regex) has a higher SFR than a signature that is written with generic rules.

- **Target Value Rating**—A weight associated with the perceived value of the target.

TVR is a user-configurable value that identifies the importance of a network asset (through its IP address). You can develop a security policy that is more stringent for valuable corporate resources and looser for less important resources. For example, you could assign a TVR to the company web server that is higher than the TVR you assign to a desktop node. In this example, attacks against the company web server have a higher RR than attacks against the desktop node.



Note

RR is a product of ASR, SFR, TVR, and ARR with an optional PD (promiscuous delta) subtracted in promiscuous mode only.

Configuring Target Value Ratings

You can assign a TVR to your network assets. The TVR is one of the factors used to calculate the RR value for each alert. You can assign different TVRs to different targets. Events with a higher RR trigger more severe signature event actions.

Use the **target-value [zero value | low | medium | high | mission-critical] target-address ip_address range** command in service event action rules submode to set TVRs for your network assets. The default is medium.

To configure TVRs for your network assets, 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
```
- Step 3** Assign the TVR to the network asset:
- ```
sensor(config-rul)# target-value mission-critical target-address 10.89.130.108
```
- Step 4** Check the TVR setting you just configured:
- ```
sensor(config-rul)# show settings

target-value (min: 0, max: 5, current: 1)

target-value-setting: mission-critical
target-address: 10.89.130.108 default: 0.0.0.0-255.255.255.255

sensor(config-rul)#
```
- Step 5** Exit event action rules submode:
- ```
sensor(config-rul)# exit
Apply Changes:[yes]:
```
- Step 6** Press **Enter** to apply your changes or type **no** to discard them.
-

Event Action Overrides

This section describes event action overrides, and contains the following topics:

- [About Event Action Overrides, page 6-7](#)
- [Configuring Event Action Overrides, page 6-8](#)

About Event Action Overrides

You can add an event action override to change the actions associated with an event based on the RR of that event. Event action overrides are a way to add event actions globally without having to configure each signature individually. Each event action has an associated RR range. If a signature event occurs and the RR for that event falls within the range for an event action, that action is added to the event. For

example, if you want any event with an RR of 85 or more to generate an SNMP trap, you can set the RR range for Request SNMP Trap to 85-100. If you do not want to use action overrides, you can disable the entire event action override component.

Configuring Event Action Overrides

Use the **overrides** [**request-block-connection** | **request-block-host** | **deny-attacker-inline** | **deny-packet-inline** | **deny-connection-inline** | **log-attacker-packets** | **log-victim-packets** | **log-pair-packets** | **reset-tcp-connection** | **produce-alert** | **produce-verbose-alert** | **request-snmp-trap**] command in service event action rules submode to configure the parameters of event action overrides.

To add event action overrides, 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
```

Step 3 To configure how packets are treated for overrides:



Note The default RR range is 0 to 100. Set it to a different value, such as 85 to 100.

a. To deny packets from the source IP address of the attacker:

```
sensor(config-rul)# overrides deny-attacker-inline
```

b. To not transmit the single packet causing the alert:

```
sensor(config-rul-ove)# exit
sensor(config-rul)# overrides deny-packet-inline
```

c. To not transmit packets on the specified TCP connection:

```
sensor(config-rul-ove)# exit
sensor(config-rul)# overrides deny-connection-inline
```

d. To send TCP RST packets to terminate the connection:

```
sensor(config-rul-ove)# exit
sensor(config-rul)# overrides reset-tcp-connection
```

Step 4 To configure overrides to request blocks:

a. To request a block of the connection:

```
sensor(config-rul-ove)# exit
sensor(config-rul)# overrides request-block-connection
```

b. To request a block of the attacker host:

```
sensor(config-rul-ove)# exit
sensor(config-rul-ove)# exit
sensor(config-rul)# overrides request-block-host
```

Step 5 To log packets for overrides:

- a. To log the packets from the attacker IP address:

```
sensor(config-rul-ove) # exit
sensor(config-rul) # overrides log-attacker-packets
```

- b. To log the packets from the victim IP address:

```
sensor(config-rul-ove) # exit
sensor(config-rul) # overrides log-victim-packets
```

- c. To log packets from both the attacker and victim IP addresses:

```
sensor(config-rul-ove) # exit
sensor(config-rul) # overrides log-pair-packets
```

Step 6 To write alerts to the Event Store:

- a. To write an alert to the Event Store:

```
sensor(config-rul-ove) # exit
sensor(config-rul) # overrides produce-alert
```

- b. To write verbose alerts to the Event Store:

```
sensor(config-rul-ove) # exit
sensor(config-rul) # overrides produce-verbose-alert
```

- c. To write events that request an SNMP trap to the Event Store:

```
sensor(config-rul-ove) # exit
sensor(config-rul) # overrides request-snmp-trap
```

Step 7 Exit event action rules submode:

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

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

Event Action Filters

This section describes event action filters, and contains the following topics:

- [About Event Action Filters, page 6-9](#)
- [Configuring Event Action Filters, page 6-10](#)

About Event Action Filters

Event action filters are processed as an ordered list and you can move filters up or down in the list.

Filters let the sensor perform certain actions in response to the event without requiring the sensor to perform all actions or remove the entire event. Filters work by removing actions from an event. A filter that removes all actions from an event effectively consumes the event.

**Note**

When filtering sweep signatures, we recommend that you do not filter the destination addresses. If there are multiple destination addresses, only the last address is used for matching the filter.

**Caution**

Event action filters based on source and destination IP addresses do not function for the Sweep engine, because they do not filter as regular signatures. To filter source and destination IP addresses in sweep alerts, use the source and destination IP address filter parameters in the Sweep engine signatures.

Configuring Event Action Filters

You can configure event action filters to remove specific actions from an event or to discard an entire event and prevent further processing by the sensor. You can use event action variables that you defined to group addresses for your filters. For the procedure for configuring event action variables, see [Configuring Event Action Variables, page 6-5](#).

**Note**

You must preface the variable with a dollar sign (\$) to indicate that you are using a variable rather than a string. Otherwise, you receive the `Bad source and destination error`.

Use the **filters [edit | insert | move] name1 [begin | end | inactive | before | after]** command in service event action rules submode to set up event action filters.

To configure event action filters, 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
```

Step 3 Create the filter name:

```
sensor(config-rul)# filters insert name1 begin
```

Use *name1*, *name2*, and so forth to name your event action filters. Use the **begin | end | inactive | before | after** keywords to specify where you want to insert the filter.

Step 4 Configure the values for this filter:

- a. Set the signature ID range:

```
sensor(config-rul-fil)# signature-id-range 1000-1005
```

The default is 900 to 65535.

- b. Set the subsignature ID range:

```
sensor(config-rul-fil)# subsignature-id-range 1-5
```

The default is 0 to 255.

- c. Set the attacker address range:

```
sensor(config-rul-fil)# attacker-address-range 10.89.10.10-10.89.10.23
```

The default is 0.0.0.0 to 255.255.255.255.

- d. Set the victim address range:

```
sensor(config-rul-fil)# victim-address-range 192.56.10.1-192.56.10.255
```

The default is 0.0.0.0 to 255.255.255.255.

- e. Set the victim port range:

```
sensor(config-rul-fil)# victim-port-range 0-434
```

The default is 0 to 65535.

- f. Set the risk rating range:

```
sensor(config-rul-fil)# risk-rating-range 85-100
```

The default is 0 to 100.

- g. Set the actions to remove:

```
sensor(config-rul-fil)# actions-to-remove reset-tcp-connection
```

You can choose from the following actions:

request-block-connection	Request SHUN of connection
request-block-host	Request SHUN of attacker host
deny-attacker-inline	Do not transmit packets with source address of attacker
deny-packet-inline	Do not transmit the single packet causing alert
deny-connection-inline	Do not transmit packets on this TCP connection
log-attacker-packets	Activate packet logging for attacker address
log-victim-packets	Activate packet logging for victim address
log-pair-packets	Activate packet logging for attacker/victim address pair
reset-tcp-connection	Send TCP RST packets to terminate connection
produce-alert	Write evIdsAlert to EventStore
produce-verbose-alert	Write evIdsAlert to EventStore with triggerPacket
request-snmp-trap	Write evIdsAlert to EventStore with SNMP request in AlarmTraits

- h. Set the status of the filter to either disabled or enabled.

```
sensor(config-rul-fil)# filter-item-status [enable | disable]
```

The default is enabled

- i. Set the stop on match parameter.

```
sensor(config-rul-fil)# stop-on-match [true | false]
```

True tells the sensor to stop processing filters if this item matches. **False** tells the sensor to continue processing filters even if this item matches.

- j. Add any comments you want to explain this filter:

```
sensor(config-rul-fil)# user-comments
```

Step 5 Verify the settings for the filter:

```
sensor(config-rul-fil)# show settings
```

```
NAME: name1
-----
signature-id-range: 1000-10005 default: 900-65535
subsignature-id-range: 1-5 default: 0-255
attacker-address-range: 10.89.10.10-10.89.10.23 default: 0.0.0.0-255.255.255.255
victim-address-range: 192.56.10.1-192.56.10.255 default: 0.0.0.0-255.255.255.255
attacker-port-range: 0-65535 <defaulted>
victim-port-range: 1-343 default: 0-65535
risk-rating-range: 85-100 default: 0-100
actions-to-remove: reset-tcp-connection default:
filter-item-status: Enabled default: Enabled
stop-on-match: True default: False
user-comment: This is a new filter. default:
-----
```

Step 6 To edit an existing filter:

```
sensor(config-rul)# filters edit name1
```

Step 7 Edit the parameters (see Steps 4a through 4j).

Step 8 To move a filter up or down in the filter list:

```
sensor(config-rul-fil)# exit
sensor(config-rul)# filters move name5 before name1
```

Step 9 Verify that you have moved the filters:

```
sensor(config-rul-fil)# exit
sensor(config-rul)# show settings
-----
filters (min: 0, max: 4096, current: 5 - 4 active, 1 inactive)
-----
ACTIVE list-contents
-----
NAME: name5
-----
signature-id-range: 900-65535 <defaulted>
subsignature-id-range: 0-255 <defaulted>
attacker-address-range: 0.0.0.0-255.255.255.255 <defaulted>
victim-address-range: 0.0.0.0-255.255.255.255 <defaulted>
attacker-port-range: 0-65535 <defaulted>
victim-port-range: 0-65535 <defaulted>
risk-rating-range: 0-100 <defaulted>
actions-to-remove: <defaulted>
filter-item-status: Enabled <defaulted>
stop-on-match: False <defaulted>
user-comment: <defaulted>
-----
NAME: name1
-----
signature-id-range: 900-65535 <defaulted>
subsignature-id-range: 0-255 <defaulted>
attacker-address-range: 0.0.0.0-255.255.255.255 <defaulted>
victim-address-range: 0.0.0.0-255.255.255.255 <defaulted>
attacker-port-range: 0-65535 <defaulted>
victim-port-range: 0-65535 <defaulted>
risk-rating-range: 0-100 <defaulted>
actions-to-remove: <defaulted>
filter-item-status: Enabled <defaulted>
stop-on-match: False <defaulted>
user-comment: <defaulted>
-----
NAME: name2
-----
signature-id-range: 900-65535 <defaulted>
subsignature-id-range: 0-255 <defaulted>
attacker-address-range: 0.0.0.0-255.255.255.255 <defaulted>
victim-address-range: 0.0.0.0-255.255.255.255 <defaulted>
attacker-port-range: 0-65535 <defaulted>
victim-port-range: 0-65535 <defaulted>
risk-rating-range: 0-100 <defaulted>
actions-to-remove: <defaulted>
filter-item-status: Enabled <defaulted>
stop-on-match: False <defaulted>
user-comment: <defaulted>
-----
```

```

-----
INACTIVE list-contents
-----
sensor(config-rul)#

```

Step 10 To move a filter to the inactive list:

```
sensor(config-rul)# filters move name1 inactive
```

Step 11 Verify that the filter has been moved to the inactive list:

```

sensor(config-rul-fil)# exit
sensor(config-rul)# show settings
-----
INACTIVE list-contents
-----
NAME: name1
-----
signature-id-range: 900-65535 <defaulted>
subsignature-id-range: 0-255 <defaulted>
attacker-address-range: 0.0.0.0-255.255.255.255 <defaulted>
victim-address-range: 0.0.0.0-255.255.255.255 <defaulted>
attacker-port-range: 0-65535 <defaulted>
victim-port-range: 0-65535 <defaulted>
risk-rating-range: 0-100 <defaulted>
actions-to-remove: <defaulted>
filter-item-status: Enabled <defaulted>
stop-on-match: False <defaulted>
user-comment: <defaulted>
-----
sensor(config-rul)#

```

Step 12 Exit event action rules submode:

```

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

```

Step 13 Press **Enter** to apply your changes or type **no** to discard them.

General Settings

This section describes the general settings, and contains the following topics:

- [About General Settings, page 6-15](#)
- [Event Action Summarization, page 6-15](#)
- [Event Action Aggregation, page 6-15](#)
- [Deny Attackers, page 6-16](#)
- [Configuring the General Settings, page 6-16](#)
- [Clearing the Denied Attackers List, page 6-18](#)

About General Settings

You can configure the general settings that apply to event action rules, such as whether you want to use the summarizer and the meta event generator. The summarizer groups events into a single alert, thus decreasing the number of alerts the sensor sends out. The meta event generator processes the component events, which lets the sensor watch for suspicious activity transpiring over a series of events.

You can configure how long you want to deny attackers, the maximum number of denied attackers, and how long you want blocks to last.

Event Action Summarization

Summarization decreases the volume of alerts sent out from the sensor by providing basic aggregation of events into a single alert. Special parameters are specified for each signature and they influence the handling of the alerts. Each signature is created with defaults that reflect a preferred normal behavior. However, you can tune each signature to change this default behavior within the constraints for each engine type.

The non-alert generating actions (deny, block, TCP reset) go through the filters for each signature event unsummarized. The alert-generating actions are not performed on these summarized alerts; instead the actions are applied to the one summary alert and then put through the filters.

If you select one of the other alert-generating actions and do not have it filtered out, the alert is created even if you do not select Produce Alert. To prevent alerts from being created, you must have all alert-generating actions filtered out.

Summarization and event actions are processed after Engine. META has processed the component events. This lets the sensor watch for suspicious activity transpiring over a series of events.

Event Action Aggregation

Basic aggregation provides two operating modes. The simple mode involves configuring a threshold number of hits for a signature that must be met before the alert is sent. A more advanced mode is timed-interval counting. In this mode, the sensor tracks the number of hits per second and only sends alerts when that threshold is met. In this example, a *hit* is a term used to describe an event, which is basically an alert, but it is not sent out of the sensor as an alert until the threshold number of hits has been exceeded.

You can select from the following summarization options:

- **Fire All**—Fire All mode fires an alert each time the signature is triggered. If the threshold is set for summarization, the following happens: Alerts are fired for each execution until summarization occurs. After summarization starts only one alert every summary interval fires for each address set. Alerts for other address sets are either all seen or separately summarized. The signature reverts to Fire All mode after a period of no alerts of that signature.
- **Summary**—Summary mode fires an alert the first time a signature is triggered, and then additional alerts for that signature are summarized for the duration of the summary interval. Only one alert every summary interval should fire for each address set. If the global summary threshold is reached, the signature goes into Global Summarization mode.

- **Global Summarization**—Global Summarization mode fires an alert for every summary interval. Signatures can be preconfigured for global summarization.
- **Fire Once**—Fire Once mode fires an alert for each address set. You can upgrade this mode to Global Summarization mode.

Deny Attackers

You can configure certain aspects of the deny attackers inline event action. You can configure the number of seconds you want to deny attackers inline and you can limit the number of attackers you want denied in the system at any one time.

Configuring the General Settings

Use the following commands in service event action rules submode to configure general event action rules settings:

- **global-block-timeout** —Number of minutes to block a host or connection.
The valid range is 0 to 10000000. The default is 30 minutes.
- **global-deny-timeout**—Number of seconds to deny attackers inline.
The valid range is 0 to 518400. The default is 3600.
- **global-filters-status** [**enabled** | **disabled**]—Enables or disables the use of the filters.
The default is enabled.
- **global-metaevent-status** [**enabled** | **disabled**]—Enables or disables the use of the Meta Event Generator.
The default is enabled.
- **global-overrides-status** [**enabled** | **disabled**]—Enables or disables the use of the overrides.
The default is enabled.
- **global-summarization-status** [**enabled** | **disabled**]—Enables or disables the use of the summarizer.
The default is enabled.
- **max-denied-attackers**—Limits the number of denied attackers possible in the system at any one time.
The valid range is 0 to 100000000. The default is 10000.

To configure event action general settings, 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
```

Step 3 Enter general submode:

```
sensor(config)# general
```

Step 4 To enable or disable the meta event generator:

```
sensor(config-rul-gen)# global-metaevent-status [enabled | disabled]
```

The default is enabled.

Step 5 To enable or disable the summarizer:

```
sensor(config-rul-gen)# global-summarization-status [enabled | disabled]
```

The default is enabled.

Step 6 To configure the denied attackers inline event action:

a. To limit the number of denied attackers in the system at any given time:

```
sensor(config-rul-gen)# max-denied-attackers 100
```

The default is 1000.

b. To configure the amount of seconds to deny attackers in the system:

```
sensor(config-rul-gen)# global-deny-timeout 1000
```

The default is 3600 seconds.

Step 7 To configure the number of minutes to block a host or a connection:

```
sensor(config-rul-gen)# global-block-timeout 20
```

The default is 30 minutes.

Step 8 To enable or disable any overrides that you have set up:

```
sensor(config-rul-gen)# global-overrides-status [enabled | disabled]
```

The default is enabled.

Step 9 To enable or disable any filters that you have set up:

```
sensor(config-rul-gen)# global-filters-status [enabled | disabled]
```

The default is enabled.

Step 10 Check the settings for general submode:

```
sensor(config-rul-gen)# show settings
general
-----
global-overrides-status: Enabled default: Enabled
global-filters-status: Enabled default: Enabled
global-summarization-status: Enabled default: Enabled
global-metaevent-status: Enabled default: Enabled
global-deny-timeout: 1000 default: 3600
global-block-timeout: 20 default: 30
max-denied-attackers: 100 default: 10000
-----
sensor(config-rul-gen)#
```

Step 11 Exit event action rules submode:

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

Step 12 Press **Enter** to apply your changes or type **no** to discard them.

Clearing the Denied Attackers List

Use the **clear denied-attackers** command in service event action rules submode to delete the denied attackers list and clear the virtual sensor statistics.

If your sensor is configured to operate in inline mode, the traffic is passing through the sensor. You can configure signatures to deny packets, connections, and attackers while in inline mode, which means that single packets, connections, and specific attackers will be denied, that is, not transmitted, when the sensor encounters them.

When the signature fires, the attacker is denied and placed in a list. As part of sensor administration, you may want to delete the list or clear the statistics in the list.

To delete the list of denied attackers and clear the statistics, follow these steps:

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

Step 2 Display the list of denied IP addresses:

```
sensor# show statistics denied-attackers
Denied Attackers and hit count for each.
  10.20.4.2 = 9
  10.20.5.2 = 5
```

The statistics show that there are two IP addresses being denied at this time.

Step 3 Delete the denied attackers list:

```
sensor# clear denied-attackers
Warning: Executing this command will delete all addresses from the list of
attackers currently being denied by the sensor.
Continue with clear? [yes]:
```

Step 4 Type **yes** to clear the list.

Step 5 Verify that you have cleared the list:

```
sensor# show statistics virtual-sensor
Virtual Sensor Statistics
  Statistics for Virtual Sensor vs0
  Name of current Signature-Definition instance = sig0
  Name of current Event-Action-Rules instance = rules0
  List of interfaces monitored by this virtual sensor = mypair
  Denied Address Information
    Number of Active Denied Attackers = 0
    Number of Denied Attackers Inserted = 2
    Number of Denied Attackers Total Hits = 287
    Number of times max-denied-attackers limited creation of new entry = 0
    Number of exec Clear commands during uptime = 1
  Denied Attackers and hit count for each.
```

There is no longer any information under the Denied Attackers and hit count for each category.

Step 6 To clear only the statistics:

```
sensor# show statistics virtual-sensor clear
```

Step 7 Verify that you have cleared the statistics:

```
JWK-4255# show statistics virtual-sensor
Virtual Sensor Statistics
  Statistics for Virtual Sensor vs0
  Name of current Signature-Definition instance = sig0
  Name of current Event-Action-Rules instance = rules0
```

```
List of interfaces monitored by this virtual sensor = mypair
Denied Address Information
  Number of Active Denied Attackers = 2
  Number of Denied Attackers Inserted = 0
  Number of Denied Attackers Total Hits = 0
  Number of times max-denied-attackers limited creation of new entry = 0
  Number of exec Clear commands during uptime = 1
Denied Attackers and hit count for each.
  10.20.2.5 = 0
  10.20.5.2 = 0
```

The statistics have all been cleared except for the Number of Active Denied Attackers and Number of exec Clear commands during uptime categories. It is important to know if the list has been cleared.

Event Action Rules Example

The following example demonstrates how the individual components of your event action rules work together.

Risk Rating Ranges for Example 1

- **Produce Alert**—1-100
- **Produce Verbose Alert**—90-100
- **Request SNMP Trap**—50-100
- **Log Pair Packets**—90-100
- **Log Victim Packets**—90-100
- **Log Attacker Packets**—90-100
- **Reset TCP Connection**—90-100
- **Request Block Connection**—70-89
- **Request Block Host**—90-100
- **Deny Attacker Inline**—0-0
- **Deny Connection Inline**—90-100
- **Deny Packet Inline**—90-100

Event Action Filters for Example 1

1. SigID=2004, Attacker Address=*, Victim Address=20.1.1.1, Actions to Remove=ALL, Risk Rating Range=1-100, StopOnMatch=True
2. SigID=2004, Attacker Address=30.1.1.1, Victim Address=*, Actions to Remove=ALL, Risk Rating Range=1-100, StopOnMatch=True
3. SigID=2004, Attacker Address=*, Victim Address=*, Actions to Remove=None, Risk Rating Range=95-100, StopOnMatch=True

4. SigID=2004, Attacker Address=*, Victim Address=*, Actions to Remove=denyAttackerInline, requestBlockHost, requestBlockConnection, Risk Rating Range=56-94, StopOnMatch=True
5. SigID=2004, Attacker Address=*, Victim Address=*, Actions to Remove=denyAttackerInline, requestBlockHost, produceAlert, resetTcpConnection, logAttackerPackets, Risk Rating Range=1-55, StopOnMatch=True

Results for Example 1

When SIG 2004 is detected:

- If the attacker address is 30.1.1.1 or the victim address is 20.1.1.1, the event is consumed (ALL actions are subtracted).

If the attacker address is not 30.1.1.1 and the victim address is not 20.1.1.1:

- If the RR is 50, Produce Alert and Request SNMP Trap are added by the event action override component, but Produce Alert is subtracted by the event action filter. However, the event action policy forces the alert action because Request SNMP Trap is dependent on the <evIdsAlert>.
- If the RR is 89, Request SNMP Trap and Request Block Connection are added by the event action override component. However, Request Block Connection is subtracted by the event action filter.
- If the RR is 96, all actions except Deny Attacker Inline and Request Block Connection are added by the event action override component, and none are removed by the event action filter. The third filter line with the filter action NONE is optional, but is presented as a clearer way to define this type of filter.