

# Custom Attack Definitions

## Custom attacks

Trellix IPS uses a variety of methods to detect aberrant network activity. The main detection method used is attack definition. An attack defines either a rule or related signatures that specify a set of checks for capturing various attempts at exploiting a given vulnerability or creating other threat conditions.

### Note

Throughout this guide, the terms attack and attack definition are used interchangeably.

Trellix IPS regularly supplies you with its own attack definitions (signature set) to protect your network. Additionally, it also provides ad hoc signature sets in case of emergencies and zero-day vulnerabilities. Trellix's research team develops these signatures and tests them thoroughly before releasing them to its customers.

There could be unique security requirements that would not be possible to be covered in the Trellix IPS-supplied signature set. For such cases, you have the option of developing your own attack definitions. Such user-defined attacks are referred to as custom attack definitions or custom attacks.

## Reasons to create your own attack definitions

If the Trellix IPS-supplied signature set is adequate to protect your network, then why create one yourself?

The reasons you might want to create an attack definition will vary widely based on the usage scenario for your IPS Sensor and the environment in which it is deployed. The sections below attempt to describe some common scenarios and the actions that might be taken when they occur.

- **Emergency situations** - Trellix follows strict quality assurance processes and requires high-quality information before providing updates to customers. Thus, there may be situations where an attack covering particular vulnerabilities may be a very high priority for your network but might not be provided as swiftly as your security policy might require. In this case, you might need to write and deploy strategic attacks based on your knowledge of your network's vulnerabilities until an update from Trellix is available.
- **Forensic analysis** - It may be useful to deploy specialized attacks to investigate suspicious activity on your network and possibly to track intrusions.
- **Custom attacks for Your particular environment** - Writing custom attacks for your network environment may be necessary for a variety of reasons. The most common is policy enforcement. For example, your security policy may dictate that certain traffic or usage patterns are disallowed. In such cases, it may be useful to write and deploy attacks that can alert your IT or security staff.
- **To preserve legacy attack definitions** - If you are migrating from an open-source IPS or IDS solution, such as Snort, you may want to import the attacks in their current format into Trellix IPS.

### Types of custom attacks

In Trellix IPS, you have the flexibility to create custom attacks in two ways:

- You can create custom attacks in Trellix IPS's proprietary format. This type of custom attacks are signature-based. You can define one or more signatures per attack. Such attack definitions are called *Trellix IPS Custom Attacks* in this guide.

#### Note

In the earlier releases, the Trellix IPS Custom Attacks was called User-Defined Signatures (UDS). The interfaces of the UDS Editor are now available within the Custom Attack Editor.

- You can write rule-based custom attacks using Snort rules language, which is open-source. Such attack definitions are called *Snort Custom Attacks* in this guide.

#### Note

These two formats are not interchangeable. That is, you cannot convert a Trellix IPS Custom Attack to Snort Custom Attack or the other way around.

You can use Snort rules in your existing Trellix IPS setup without having to modify it in any way or install any Snort-related components. These rules could be your own or from sources such as the Snort user-community; the critical thing is that the rules should conform to the Snort syntax for you to use them in Trellix IPS.

A Snort Custom Attack is converted into Trellix IPS's format internally when you save it in the Manager server. This translation enables you to use Snort rules directly in Trellix IPS without the need for any Snort-related components.

### Trellix IPS signature terminology

To be able to create effective custom attacks, you need to understand how Trellix IPS detects attacks. To be able to understand this mechanism, you need to be familiar with some terms listed below:

- Signatures and rules
- Attacks
- Policies
- Alerts

#### Signatures and rules

A signature or rule is a set of checks (for example, string matches or IP-port comparisons) that are applied to network traffic seen by the Sensor. The term signature is relevant for Trellix IPS Custom Attacks (that is, custom attacks in Trellix IPS format) or the Trellix IPS-supplied signature set. The term rule applies to Snort Custom Attacks.

In case of Snort Custom Attacks, the Manager parses them to check the syntax. Then it converts the valid attack definitions to Trellix IPS's format and saves them in the Manager database. Once in the database, the converted Snort rules function like any other Trellix IPS signature. So, in Trellix IPS, both signatures and rules are functionally similar. What applies to a signature is also applicable to a rule unless stated otherwise.

#### Note

Throughout this guide, Snort Custom Attacks are also referred to as Snort rules or just rules. This is not to be confused with the Trellix IPS rule sets, which are a collection of attack definitions that meet certain criteria.

The attack detection mechanism in Trellix IPS relies on comparison of traffic to a database of signatures or rules. Trellix IPS enables you to define checks using combinations of string matches and checks for other anomalies such as excessive field lengths. When all the necessary conditions for a given rule or set of signatures are satisfied, an event is raised in the Sensor; all signatures and rules defined are checked against the traffic simultaneously and all matching events within the context of a given attack are correlated by the Sensor to generate a single alert.

### Attacks, policies, and alerts

In Trellix IPS, attack definitions are a mechanism used to identify and protect against malicious actions taken against your network. An attack definition is the aggregation of the signatures (or rule) and other supporting data that can identify a specific network event. When you select an attack, you are essentially selecting a group of conditions defined in a rule or signature(s).

#### Note

A rule in a Snort Custom Attack corresponds to the signatures in a Trellix IPS Custom Attack.

Policies are applied to the Sensor and consist of one or more attacks. Traffic passing through the Sensor is compared to the attacks enabled in the policies, and if any traffic is identified as malicious by an attack definition, an alert is triggered to notify you of the incident.

### Custom attack editor

Custom Attack Editor provided in the Manager is the tool that you use to create custom attacks. Custom Attack Editor enables you to create Trellix IPS Custom Attacks as well as in Snort Custom Attacks. Using this tool, you can also import custom attacks in bulk against defining them individually.

### Audience

Custom Attack Editor is designed for sophisticated users with expertise in networking and intrusion prevention. The ability to create a custom attack is a double-edged sword. It can be a very powerful detection and defense mechanism, but at the same time, when used without training and experience, a custom attack can cause harm to your network and the business that depends on it. In addition, without a significant amount of experience in both using and configuring detection mechanisms for network intrusion detection devices, it is possible to make mistakes that can render your detection device essentially useless. For example, a mistake in implementation of a signature on a high-traffic network could cause such a large number of alerts to be generated that it would render the Manager unusable. On the opposite end of the spectrum, without proper experience and expertise, it is likely that a user might create a signature that would never detect security incidents (due to errors in tests or detection window, for example), despite the fact that the signature might be intended to detect very important events specific to your network.

Since a poorly written custom attack can cause many more problems than it solves, Trellix recommends that an attack writer possesses the following knowledge:

- A strong understanding of computer networking
- Experience with networks running the protocol for which you intend to create a custom attack, including a good packet-level understanding of the protocol
- The ability to recognize the difference between "good" and "bad" traffic. That is, traffic that is correct and valid for your network and the devices that comprise it, and traffic that is anomalous to your network's configuration and security policies.
- A strong understanding of Snort rules language, if you plan to use Snort Custom Attacks

While the fact that Trellix IPS provides deep parsing of application protocols and supports complex attack-definition structures may seem overwhelming to first-time users, it is reasonably straightforward to create custom attacks after crossing this initial hurdle.

## Getting started with custom attacks

Creating a custom attack requires certain experience as well as a number of different pieces of information, both technical and policy-oriented. This chapter describes the information you should have before you start creating custom attacks in Trellix IPS.

See also

[Create custom attacks on page 1547](#)

### Before you create a custom attack

Creating attack definitions is a complex topic on which books have been written. It is highly recommended that you refer to one of these books if you are not an experienced attack definition author. Pay close attention to detail when you define your attacks, as there are many ways to shoot yourself (and your network!) in the foot as you develop expertise with attack definitions.

Before creating a custom attack, you should carefully consider the requirement that you are trying to address. This means that you should have a clear understanding of the attack's purpose, such as a specific business or policy requirement.

You should have a clear rationale for using a custom attack instead of another mechanism. In some cases, a custom attack may not be the appropriate solution. For this reason, you should consider whether you can address your need with other technical means, such as router ACLs, firewall rules, or a network sniffer.

Finally, you should verify that the custom attack you intend to create does not duplicate any attack provided in Trellix IPS. However, in case of a duplicate, you have the flexibility to use both or just the custom attack instead of the Trellix IPS-supplied attack. If you choose to use both, note that the Sensor may raise two alerts for the same attack traffic.

### Required information for creating a custom attack

The following is a list of the information you should have in hand as you create your custom attacks and the constituent signatures or rule. A signature or rule can range from very simple (for example, checking the value of a header field) to highly complex checks of different information in a specific order. You must have good bit of data to aid yourself in creating an accurate attack definition, such as the following:

- Reason for creating this custom attack
- Technical information references for this custom attack
- Protocol in which this custom attack will search the traffic (also known as the impact protocol)
- Specific hardware or software platforms affected by this traffic (also known as impact packages)
- Severity of this event
- The direction in which the "traffic to be watched for" occurs
- Specific criteria that comprise the attack, such as field values and patterns to match
- A method, data, or tool to be used for testing the attack before you use it in your production environment



### Understanding impact packages and protocols

Trellix IPS policies define what hardware and software platforms are present on a network and what should be monitored. The platforms can be as generic as the HTTP protocol or as specific as Microsoft Internet Explorer running on Windows Server 2016.

After a policy is configured and saved, the Manager searches through the database and generates a list of attacks that will be enabled when that policy is applied to an interface. Attacks specify what platforms they affect as part of their impact construct.

In case of Trellix IPS Custom Attacks, the impact construct of an attack can contain references to protocols, and the packages that provide support for those protocols. Protocol references refer to protocols defined by Trellix as well as the custom-defined ones. Package references refer to particular software or hardware platforms, which can include a specific OS.

The platforms that can be selected are those for which Trellix IPS has specific support in some form. If you do not see a particular platform, you can choose the one which is similar, or specify `tcpip-machine` as the package, or specify just the impact protocol for the attack.

For Snort Custom Attacks, you cannot choose an impact package. By default, `tcpip-machine` is selected as the package, and this cannot be modified. Additionally, the Manager identifies the impact protocol for the attack definition.

### How Trellix IPS prevents intrusions

The attack prevention mechanism of Trellix IPS is very powerful, but may also be difficult to understand. A good analogy is to that of a DNA test. DNA testing allows biology experts to obtain a DNA sample from a member of a species, and use the sample to determine the individual from which the particular sample came. Trellix IPS provides much of the same capability, but oriented toward detecting and accurately identifying network events. As an example, Trellix IPS's detection mechanisms can allow your signature to identify every HTTP traffic flow, every HTTP traffic flow using the GET mechanism, every HTTP traffic flow using GET with `/cgi-bin/calendar.pl` as the path, and even every GET with that path and a parameter named **month** with a value of **February**.

This is why Trellix IPS supports the aggregation of multiple conditions into every attack. Each signature or rule within an attack can be more or less specific so as to identify everything from generic network activity that affects a given platform in a particular way to a specific piece of code that has very specific and identifiable effects. Based on their specificity and severity, signatures and rules are assigned different confidence and severity values.

When a network event occurs that matches an existing attack definition, the rule or signature(s) (generic and specific) within that attack definition may be triggered. When alert throttling is enabled, the Sensor correlates the multiple triggering events automatically to raise a single alert with the highest confidence level.

### How attack detection works

The Sensor performs different levels of traffic processing and analysis. Signatures and rules operate on traffic that has passed through these higher-level processing phases. Understanding how attack detection works can enable you to create effective Custom Attacks.

#### Flows

At the highest level, the Sensor addresses UDP and TCP traffic based on the concept of a flow. Flows are defined by their protocol (UDP/TCP), the source and destination ports, and IP addresses of their endpoints. As you might be aware, UDP does not contain the concept of "state" that TCP does. So the Sensor implements a timer-based flow context for UDP traffic. After dividing traffic into flows, the Sensor makes use of port mappings or, in the case of traffic running on non-standard ports, intelligent protocol identification, to pass each flow to the appropriate protocol parsing mechanism.

For a custom attack, you can specify whether the Sensor should look at the complete flow, one direction of the flow, or restrict itself to data occurring within single packets of the flow. Precise control of this detection window is necessary for accurate detection of attacks.

### Protocol parsing specifications

Protocol specifications (Trellix IPS's protocol parsing mechanisms) parse through network flows to validate traffic and divide it into protocol fields which may then be actively tested against Trellix IPS-supplied attacks or Custom Attacks. By dividing protocol traffic into the appropriate fields, a Sensor can perform matches against the most specific field or subfield pertinent to an effective attack, thus resulting in very low false-positive rates. Since the parsing process is fully stateful, it allows detection of anomalies in the protocol's behavior. Additionally, this parsing makes it possible to provide an additional benefit to Trellix IPS Custom Attack writers in the form of qualifiers. Qualifiers are tests that are embodied in the name of a particular protocol field. For example, rather than specifying that an HTTP request method must be "GET", the Trellix IPS system allows you to use "http-get-req-uri" as the name of the field, saving you the requirement of providing that test in the Custom Attack, and the Sensor from having to perform an extra pattern match.

### Packet searches

Traffic flows that are not identified as belonging to any particular protocol are passed to the packet search protocol specification engine for further parsing. Trellix IPS presents each direction of the flow to Trellix IPS-supplied attacks and any Custom Attacks. Tests against packet search traffic typically take the form of specific ordered pattern matches so as to prevent false positives and performance problems.

### Attack definitions

Attack definitions tie together elements of the above-described framework to derive specific "fingerprints" for network traffic from smaller building blocks.

In essence, attack definitions are like DNA tests. They can identify both specific people and relatives of that person. In the IPS case, the relatives may be a collection of buffer overflow attacks against a certain piece of software, and the particular person would be a specific piece of exploit code.

While the two are not greatly different, Trellix IPS adopts a convention of differentiating between attacks based on abnormality and attacks based on specific traffic. The main difference is while anomaly-based attack detection process examines the network for unexpected or non-conforming behavior, specific attack definitions will often look for a very particular indicator, such as a flag with a particular value, or a specific string's presence. Anomaly attacks know what to expect in normal traffic, and are triggered when they get something else. Normal attack definitions look for specific misbehavior. The custom attacks that you define must check for behavioral anomalies as well as specific exploit strings. Thus, all possible attempts to exploit a vulnerability can be detected.

### Technical information references

A Custom Attack definition is generally based on an advisory or some other description of a known vulnerability. You should have in hand whatever information you can find regarding the attack definition. This can include traffic dumps of an attack in progress or the exploit code itself. You can use this information to determine the characteristics of the vulnerability.

You should know the specific criteria that the attack definition should comprise of, such as field values and patterns to match. Your research may lead to a long list of characteristics specific to the exploit traffic. However, bear in mind that an attack definition based on all suspicious characteristics may be too specific. Although it would be precise, it may impact a Sensor's throughput or lead to detection problems. On the other hand, an attack definition based on only one of the characteristics may be too broad and generate false positives.

### Importance of testing custom attack definitions

It is imperative that you thoroughly test your attack definitions before deploying them in a production environment. Incorrect definitions can lead to false positives, false negatives, and performance problems, any of which could be very detrimental to the security and reliability of your network. The best way to avoid these problems is to make use of a comprehensive test plan that puts the attack definition through the full deployment process and verifies that it performs as expected.

Trellix recommends that at a minimum you include the following verification exercises in your test plan:

- Use traffic generation tools or packet dumps to verify that your attack definitions match the traffic they are supposed to detect.
- If possible, verify that any custom attack definition is not duplicating functionality already available in Trellix IPS. For example, check if there is Trellix IPS-supplied attack definition for the same condition. You can do this by examining whether your test traffic raises duplicate alerts - one for Trellix IPS-supplied attack and another for custom attack.
- Deploy the custom attacks on a non-production Sensor connected to either a test network that mirrors your production network traffic or a non-production Sensor connected to your production network in SPAN or Tap mode.

### Quick tour of the custom attack editor

To create and import Custom Attacks, you use the Custom Attack Editor - a powerful tool available in the Manager. Using this tool, you can define both Trellix IPS Custom Attacks and Snort Custom Attacks. This section explains the user interfaces and features of the Custom Attack Editor.

### Basics of the custom attack editor interface

There could be unique security requirements that would not be possible to be covered in the Trellix IPS-supplied signature set. For such cases, you have the option of developing your own attack definitions. Such user-defined attacks are referred to as Custom Attack definitions or Custom Attacks.

Custom Attack Editor enables you to create Trellix IPS Custom Attacks as well as Snort Custom Attacks. Using this tool, you can also import custom attacks in bulk against defining them individually. You use the Custom Attack Editor to manage custom attacks. You can launch it from the **Custom Attacks** page of the Manager. From the Manager, select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click **Custom Attacks**.

#### Note

The Custom Attacks menu option is available only for the root admin domain.

The Custom Attacks page provides the following options:

- Native Trellix IPS Format:** You can create custom attacks in Trellix IPS's proprietary format. This type of custom attacks are signature-based. You can define one or more signatures per attack. Such attack definitions are called Native Trellix IPS Format Custom Attacks.
- Snort Format:** You can write rule-based custom attacks using Snort rules language, which is open-source. Such attack definitions are called Snort Custom Attacks. In case of Snort Custom Attacks, the Manager parses them to check the syntax. It then converts all valid attack definitions to Trellix IPS's format and saves them in the Manager database. Once in the database, the converted Snort rules function like any other Trellix IPS signature. So, in Trellix IPS, both signatures and rules are functionally similar implying that what applies to a signature also applies to a rule unless stated otherwise.

Intrusion Prevention > Policy Types > IPS

Custom Attacks

Native Trellix IPS Format

Snort Format

Quick Search

Clear All Filters

|    | State     | Name ↑                                              | Severity   | BTP      | Attack Category | Test Compile | Trellix IPS ID | Last Update Time   |
|----|-----------|-----------------------------------------------------|------------|----------|-----------------|--------------|----------------|--------------------|
| 1  | Published | UDS- HTTP: Apache Struts 2 Remote Code execution... | High (8)   | Low (1)  | Exploit         | Success      | 0xc0002200     | Jul 02, 2021 19:27 |
| 2  | Published | UDS-2k_limit_test                                   | High (9)   | High (6) | Exploit         | Success      | 0xc0009700     | Jul 02, 2021 19:27 |
| 3  | Published | UDS-76_char_border                                  | High (9)   | Low (1)  | Exploit         | Success      | 0xc0009b00     | Jul 02, 2021 19:27 |
| 4  | Published | UDS-Apache Struts2 Remote Code Execution (CVE-2...  | High (8)   | Low (1)  | Exploit         | Success      | 0xc0003700     | Jul 02, 2021 19:27 |
| 5  | Published | UDS-Apache Struts2-CVE-2017-9805                    | High (7)   | Low (1)  | Exploit         | Success      | 0xc0005500     | Jul 02, 2021 19:27 |
| 6  | Staged    | UDS-ATT-Testing                                     | Low (3)    | High (6) | Exploit         | Failed       | 0xc0099000     | Sep 22, 2016 2:06  |
| 7  | Published | UDS-Attack-DESTIP2000                               | Medium (6) | High (6) | Exploit         | Success      | 0xc1434300     | Jul 02, 2021 19:27 |
| 8  | Published | UDS-Attack-Dns10                                    | High (9)   | High (6) | Exploit         | Success      | 0xc1396300     | Jul 02, 2021 19:27 |
| 9  | Published | UDS-Attack-Dns100                                   | High (9)   | High (6) | Exploit         | Success      | 0xc1255100     | Jul 02, 2021 19:27 |
| 10 | Published | UDS-Attack-Dns1000                                  | High (9)   | High (6) | Exploit         | Success      | 0xc1009300     | Jul 02, 2021 19:27 |
| 11 | Published | UDS-Attack-Dns1001                                  | High (9)   | High (6) | Exploit         | Success      | 0xc1303200     | Jul 02, 2021 19:27 |

+ - Other Actions

8798 (of 8798) attacks showing

Custom Attacks

Save Cancel

Default page of the Custom Attack Editor

This section explains the default page that is displayed when you launch the Custom Attack Editor.

Intrusion Prevention > Policy Types > IPS

Custom Attacks

Native Trellix IPS Format

Short Format

Quick Search

Clear All Filters

| State | Name ↑    | Severity                                            | BTP        | Attack Category | Test Compile | Trellix IPS ID | Last Update Time |                    |
|-------|-----------|-----------------------------------------------------|------------|-----------------|--------------|----------------|------------------|--------------------|
| 1     | Published | UDS- HTTP: Apache Struts 2 Remote Code execution... | High (8)   | Low (1)         | Exploit      | Success        | 0xc0002200       | Jul 02, 2021 19:27 |
| 2     | Published | UDS-2k_limit_test                                   | High (9)   | High (6)        | Exploit      | Success        | 0xc0009700       | Jul 02, 2021 19:27 |
| 3     | Published | UDS-76_char_border                                  | High (9)   | Low (1)         | Exploit      | Success        | 0xc0009b00       | Jul 02, 2021 19:27 |
| 4     | Published | UDS-Apache Struts2 Remote Code Execution (CVE-2...  | High (8)   | Low (1)         | Exploit      | Success        | 0xc0003700       | Jul 02, 2021 19:27 |
| 5     | Published | UDS-Apache Struts2-CVE-2017-9805                    | High (7)   | Low (1)         | Exploit      | Success        | 0xc0005500       | Jul 02, 2021 19:27 |
| 6     | Staged    | UDS-ATT-Testing                                     | Low (3)    | High (6)        | Exploit      | Failed         | 0xc0099000       | Sep 22, 2016 2:06  |
| 7     | Published | UDS-Attack-DESTIP2000                               | Medium (6) | High (6)        | Exploit      | Success        | 0xc1434300       | Jul 02, 2021 19:27 |
| 8     | Published | UDS-Attack-Dns10                                    | High (9)   | High (6)        | Exploit      | Success        | 0xc1396300       | Jul 02, 2021 19:27 |
| 9     | Published | UDS-Attack-Dns100                                   | High (9)   | High (6)        | Exploit      | Success        | 0xc1255100       | Jul 02, 2021 19:27 |
| 10    | Published | UDS-Attack-Dns1000                                  | High (9)   | High (6)        | Exploit      | Success        | 0xc1009300       | Jul 02, 2021 19:27 |
| 11    | Published | UDS-Attack-Dns1001                                  | High (9)   | High (6)        | Exploit      | Success        | 0xc1303200       | Jul 02, 2021 19:27 |

+ - Other Actions

8798 (of 8798) attacks showing

Custom Attacks

Save Cancel

Figure 4-1 Default page of Custom Attack Editor

| Item | Description               |
|------|---------------------------|
| 1    | Tabs                      |
| 2    | Regular expression search |
| 3    | Viewing options           |
| 4    | Menu options              |
| 5    | Save/Cancel buttons       |

The default page of the Custom Attack Editor has the following areas:

- **Tabs** — All the custom attacks are displayed in tabbed regions in the Custom Attack Editor. There are two tabs in the Custom Attack Editor:
  - **Native Trellix IPS Format** tab is the default tab that is displayed when you launch the Custom Attack Editor. It lists all the Native Trellix IPS Format custom attacks that are currently stored in the Manager server as well as newly created custom attacks.

You can use this tab to perform the following actions on Native Trellix IPS Format Custom Attacks:

- Add, copy, or delete
- Other actions such as:
  - Export
  - Test Compile
  - Import
  - Check Attack Counts
  - Export All
  - Manage Grepping Protocols
  - Save as CSV

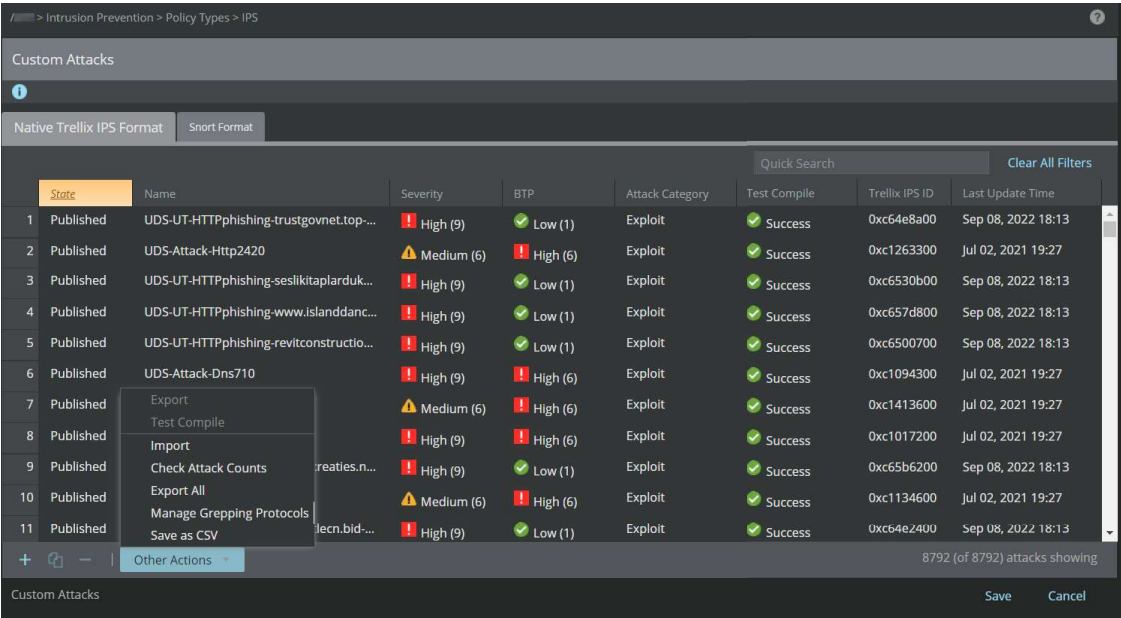


Figure 4-2 Native Trellix IPS Format tab

- **Snort Format** tab displays all the Snort Custom Attacks.

You can use this tab to perform the following actions on Snort Format Custom Attacks:

- Add, copy, or delete
- View Snort variables
- Other actions such as:
  - Export
  - Test Compile
  - Import
  - Check Attack Counts
  - Export All
  - Manage Grepping Protocols
  - Save as CSV

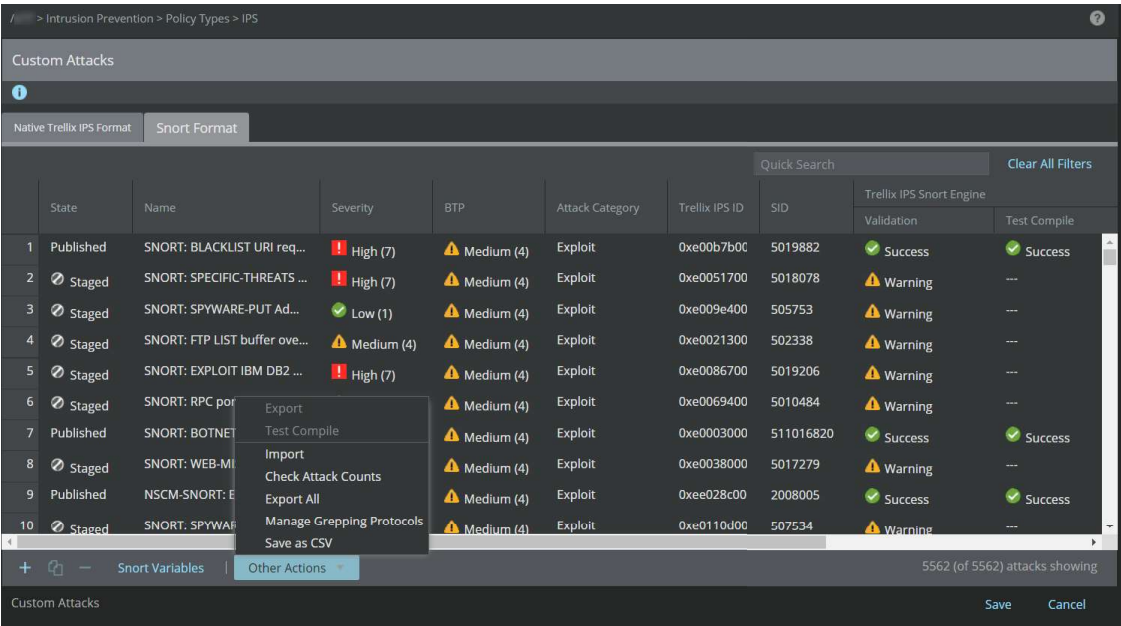


Figure 4-3 Snort Format tab

- **Regular expression search** — You can use Java regular expressions to ensure a quicker search. The following table shows some of the important expressions that can be used for quick filter option.

| Regular expression | Description                              |
|--------------------|------------------------------------------|
| ^                  | Matches beginning of line                |
| \$                 | Matches end of line                      |
| abc*xyz            | String starts with abc and ends with xyz |
| \A                 | Beginning of entire string               |
| \z                 | End of entire string                     |
| .                  | Matches any single character             |

- **Viewing options** — This area contains the options you can use to list just the custom attacks that you wish to view. These options can be helpful when you are trying to locate specific custom attacks from a large set.  
You can hide a column. You can also modify the way the content is aligned in each of the columns.



- **Menu Options** — The main actions that you can perform in the Custom Attack Editor.
- **Save\Cancel buttons** — These options are used to Save or Cancel the changes that you have made.

Menu items

This section describes the menu options that are available in Custom Attack Editor.

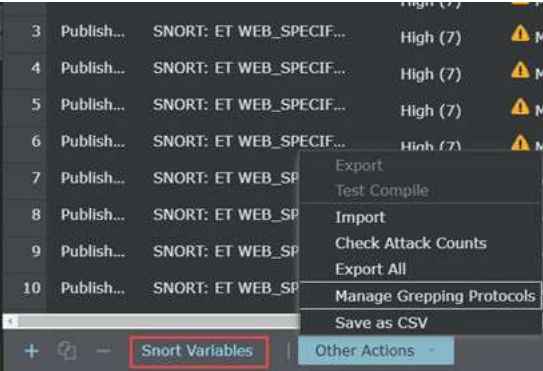


Figure 4-4 File menu

| Option                                                          | Definition                                                                                                                                                            |
|-----------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Applicable to Native Trellix IPS Format and Snort Format</b> |                                                                                                                                                                       |
|                                                                 | Click to define a custom attack. You can either create an exploit attack or a reconnaissance attack. You need to create an attack before adding signatures to it.     |
|                                                                 | Click to copy an existing custom attack.                                                                                                                              |
|                                                                 | Click to delete the selected attack from the Manager client. To delete it from the Manager server, you need to click <b>Save</b> after you delete it from the client. |
| Export                                                          | Select this option to export the custom attacks in the Manager to a file.                                                                                             |
| Test Compile                                                    | Click to Test Compile the selected custom attacks.                                                                                                                    |
| Import                                                          | Select this menu to import Native Trellix IPS Format or Snort Format custom attack from files to the Manager.                                                         |
| Check Attack Counts                                             | Click to view the count of published custom attacks.                                                                                                                  |
| Export All                                                      | Click to export all the custom attacks in the Manager to a file.                                                                                                      |
| Manage Grepping Protocols                                       | Click to Manage Grepping Protocols.                                                                                                                                   |
| Save as CSV                                                     | Click to save the custom attacks in CSV format.                                                                                                                       |
| <b>Applicable to Snort Format</b>                               |                                                                                                                                                                       |
| Snort Variables                                                 | Click to manage Snort Variables.                                                                                                                                      |

Important

Before you delete a custom-defined protocol, make sure it is not used in any of the Trellix IPS Custom Attacks.



Tab regions

To enable easy viewing, Custom Attacks are listed in tabbed regions in the **Custom Attack Editor**. The **Native Trellix IPS Format** tab is the default tab that is displayed when you launch the **Custom Attack Editor**. It displays all the Native Trellix IPS Format custom attacks. The **Snort Format** tab displays all the Snort Custom Attacks.

You can click on a column heading and drag-and-drop it to suit your viewing preference. Click on the arrow at the end of the column heading to:

- Hide the column
- Display any hidden columns
- Sort or filter the contents in the column

|   | State     | Name ↑             |
|---|-----------|--------------------|
| 1 | Published | UDS-Attack-Dns1225 |
| 2 | Published | UDS-Attack-Dns1229 |
| 3 | Published | UDS-Attack-Dns1379 |
| 4 | Published | UDS-Attack-Dns1454 |
| 5 | Published | UDS-Attack-Dns199  |
| 6 | Published | UDS-Attack-Dns369  |
| 7 | Published | UDS-Attack-Dns409  |

Note

What you see in the figure below are the default column headings.

| Native Trellix IPS Format |           | Snort Format                            |          |                 |              |                |                               |
|---------------------------|-----------|-----------------------------------------|----------|-----------------|--------------|----------------|-------------------------------|
| State                     | Name ↑    | Severity                                | BTP      | Attack Category | Test Compile | Trellix IPS ID | Last Update Time              |
| 1                         | Published | UDS- HTTP: Apache Struts 2 Remote Co... | High (8) | Low (1)         | Exploit      | Success        | 0xc0002200 Jul 02, 2021 19:27 |
| 2                         | Published | UDS-2k_limit_test                       | High (9) | High (6)        | Exploit      | Success        | 0xc0009700 Jul 02, 2021 19:27 |
| 3                         | Published | UDS-76_char_border                      | High (9) | Low (1)         | Exploit      | Success        | 0xc0009b00 Jul 02, 2021 19:27 |
| 4                         | Published | UDS-Apache Struts2 Remote Code Exec...  | High (8) | Low (1)         | Exploit      | Success        | 0xc0003700 Jul 02, 2021 19:27 |
| 5                         | Published | UDS-Apache Struts2-CVE-2017-9805        | High (7) | Low (1)         | Exploit      | Success        | 0xc0005500 Jul 02, 2021 19:27 |

Figure 4-5 Tab regions

Table 4-1 Tab regions

| Item           | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Trellix IPS ID | <div>When you save the added/imported custom attacks to the database, the Manager assigns a unique attack ID to each rule. For native Trellix IPS attacks, the ID starts with 0xc and for Snort attacks, it starts with 0xe.</div> <div> <b>Note:</b> Until you save the newly-added attacks to the Manager database, they are assigned the same temporary ID.</div>               |
| SID            | <div>Snort rule ID (SID) is the ID assigned to a Snort rule by you or the party that provided the rule. The Snort attack definitions that you want to save in the Trellix IPS database must have a unique SID. Make sure that the SIDs of the attacks that you are writing or importing have not been used by the definitions that are already in the database.</div> <div>For attack definitions that failed to import, the Trellix IPS assigns -1 as the SID.</div> |

Table 4-1 Tab regions (continued)

| Item                            | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Trellix IPS Snort Engine</b> | <p>The imported rules are available for both the Trellix IPS Snort and the Suricata Snort. This column indicates whether the rule was validated successfully.</p> <p>Some rules may have been converted but with warnings. For example, rules for which an equivalent Trellix IPS signature exists are converted with warnings.</p> <p>Some rules may have failed to convert. For example, rules that use undefined variables fail to convert.</p> <p>The <b>Validation</b> field displays status of attack validation from the Trellix IPS Snort Engine. Validation results can be Success, Failed, or Warning.</p> <p>The <b>Test Compile</b> column displays the status of test compilation.</p>                                                                                                                                                                                                                                                                                                                                                                                                              |
| <b>Suricata Snort Engine</b>    | <p>The <b>Validation</b> column displays status of attack validation from the Suricata Snort Engine. Validation results for the Suricata Snort rules can be Success or Failed.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>Last Updated Time</b>        | This is the time stamp when a rule was imported or modified.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>State</b>                    | <p>This column indicates whether a rule will be published and therefore made available for inclusion in the IPS policies. You can click on a rule to publish or stage it.</p> <div>  <p><b>Note:</b></p> <ul style="list-style-type: none"> <li>• If you publish a Snort rule that overlaps with a native Trellix IPS attack definition, you can expect two alerts to be generated for the same rule.</li> <li>• Only the <b>Published</b> rules are considered for inclusion in the IPS policies. For more information, see the tables <i>Rules for determining the state of a Custom Attack in Native Trellix IPS format</i> and <i>Rules for determining the state of a Custom Attack in Snort format</i> below.</li> </ul> </div>                                                                                                                                                                                                                                                                                           |
| <b>Name</b>                     | <p>The name for the attack that the Manager assigns post-import. The format that the Manager uses for the name is SNORT:&lt;the text specified for the msg rule option in the rule&gt; (&lt;SID&gt;). So this name is modified accordingly if you modify the msg text or the SID of the rule.</p> <p>Note that msg rule option and a unique SID are required in a rule for you to import it into the Manager.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>Severity</b>                 | <p>The Manager assigns a severity of Low, Medium, or High based on the priority value of the rule.</p> <ul style="list-style-type: none"> <li>• A priority 1 Snort attack definition is assigned a severity of High.</li> <li>• A priority 2 Snort attack definition is assigned a severity of Medium.</li> <li>• A priority of 3 or higher is assigned a severity of Low.</li> </ul> <p>You can set the priority for a rule by using the priority keyword in the rule definition. If the priority keyword is not present in the rule, the Manager derives the priority value based on the classtype of the rule. However, if the classtype is not available for the rule and there is no priority set in the rule definition, the Manager assigns a priority value of 0 to the rule.</p> <div>  <p><b>Tip:</b> Severity is often used in attack set profiles to include/exclude attacks from IPS policies. To ensure inclusion of a rule in a policy, assign a Severity value as per the policy profile settings.</p> </div> |

Table 4-1 Tab regions (continued)

| Item                                    | Description                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|-----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>BTP</b> (Benign Trigger Probability) | <p>This column indicates the possibility of producing false positives. The lower the BTP value, higher is the accuracy of the rule.</p> <div>  <p><b>Tip:</b> BTP is often used in attack set profiles to include/exclude attacks from IPS policies. To ensure inclusion of a rule in a policy, assign a BTP value as per the policy profile settings.</p> </div> |
| <b>Attack Target</b>                    | This column indicates the type of attack target such as client, server, and so on.                                                                                                                                                                                                                                                                                                                                                                 |
| <b>Test Compile</b>                     | This column indicates the compilation status of the attack.                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>Attack Category</b>                  | This column indicates the category of the attack such as malware, exploit, policy violation, and so on.                                                                                                                                                                                                                                                                                                                                            |
| <b>Protection Category</b>              | Indicates the Protection Category to which the Trellix IPS Custom Attack belongs. You specify this when you create a Trellix IPS Custom Attack.                                                                                                                                                                                                                                                                                                    |
| <b>Supported Device Types</b>           | This column indicates the supported devices for the Snort Rule.                                                                                                                                                                                                                                                                                                                                                                                    |

 **Note**

The columns **SID**, **Trellix IPS Snort Engine**, and **Suricata Snort Engine** are available only on the **Snort Format** tab.

Table 4-2 Rules for determining the state of a Custom Attack in Native Trellix IPS format

| State of the Custom Attack | Trellix IPS Snort Engine Validation Status | Test Compile Status |
|----------------------------|--------------------------------------------|---------------------|
| Published                  | Success                                    | Success             |
| Published                  | Warning                                    | Success             |
| Staged                     | Warning                                    | Failed              |
| Staged                     | Failed                                     | Pending             |
| Staged                     | Success                                    | Failed              |

Table 4-3 Rules for determining the state of a Custom Attack in Snort format

| State of the Custom Attack | Trellix IPS Snort Engine Validation Status | Test Compile Status | Suricata Engine Validation Status |
|----------------------------|--------------------------------------------|---------------------|-----------------------------------|
| Published                  | Success                                    | Success             | Success                           |
| Published                  | Success                                    | Failed              | Success                           |
| Published                  | Success                                    | Success             | Failed                            |
| Published                  | Failed                                     | Pending             | Success                           |
| Published                  | Warning                                    | Success             | Success                           |
| Published                  | Warning                                    | Failed              | Success                           |
| Staged                     | Warning                                    | Success             | Failed                            |
| Staged                     | Warning                                    | Failed              | Failed                            |

**Table 4-3** Rules for determining the state of a Custom Attack in Snort format *(continued)*

| State of the Custom Attack | Trellix IPS Snort Engine Validation Status | Test Compile Status | Suricata Engine Validation Status |
|----------------------------|--------------------------------------------|---------------------|-----------------------------------|
| Staged                     | Failed                                     | Pending             | Failed                            |
| Staged                     | Success                                    | Failed              | Failed                            |

## View options

Various options are available on the tabs of the Custom Attack Editor to enable you to easily locate the information that you require. The options are:

- Sorting based on column values
- Group By
- Display Filter

| State       | Name               | Severity | BTP      | Attack Category | Test Compile |
|-------------|--------------------|----------|----------|-----------------|--------------|
| 1 Published | UDS-Attack-Dns1225 | High (6) | High (6) | Exploit         | Success      |
| 2 Published | UDS-Attack-Dns1229 | High (6) | High (6) | Exploit         | Success      |
| 3 Published | UDS-Attack-Dns1379 | High (6) | High (6) | Exploit         | Success      |
| 4 Published | UDS-Attack-Dns1454 | High (6) | High (6) | Exploit         | Success      |
| 5 Published | UDS-Attack-Dns199  | High (9) | High (6) | Exploit         | Success      |
| 6 Published | UDS-Attack-Dns369  | High (9) | High (6) | Exploit         | Success      |

**Sorting Ascending/Descending** — You can sort the list of Custom Attacks based on the values of one or more columns. This can be in the ascending or descending order for the selected columns.

To sort the display information based on a column, click on the column heading. To base it on multiple columns, click on the required column headings (in the order that you want).

**Group By this field** — This option enables you to view the aggregate attack count based on certain values. For example, out of the attacks listed in a tab view, you can view the number of high, medium, and low-severity attacks.

**Filter** — To locate custom attacks based on values such as severity and conversion result, you can just sort the listed attack definitions based on these values by clicking on the corresponding column heading. For advanced search capabilities, you can use the Display Filter feature. You can create a display filter with a set of criteria to view only those attack definitions that meet the criteria. For example, you can locate attack definitions based on what is contained in the attack names. This way you can easily locate specific attack definitions from a larger set.

### Note

A Filter that you define is always applied to the attacks listed on the both the tabs and not just to the attacks on the currently selected tab.

## Attack creation interfaces

As discussed earlier, there are two types of custom attack definitions: Native Trellix IPS Format custom attack and Snort Format custom attack. The methods to create these two types of attacks are different. So, the interfaces where you define them also differ accordingly. This section describes these interfaces in detail.

Trellix IPS custom attack creation interfaces

To access the Trellix IPS Custom Attack creation interface:

- 1 In the Custom Attack Editor, Click .

The New Custom Attack creation interface presents information related to naming, describing, and categorizing the attack. You can either create an Exploit Attack or a Reconnaissance Attack.

See also  
[Create custom attacks on page 1547](#)

Attack tab

The **Attack** tab enables you to name your attack and type a description.

| Option                     | Definition                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| State                      | Select the state of the custom attack. The choices are <b>Published</b> and <b>Staged</b> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| Name                       | The name you assign to the attack. "UDS" is automatically prefixed before every created attack name. For example, if you name the new attack "HTTP Attack XYZ", it appears as "UDS-HTTP Attack XYZ" in the Custom Attack Editor, as well as in the attack database when you subsequently save the attack in the Manager server.                                                                                                                                                                                                                                              |
| Description                | Use this area for notes and other pertinent information. <div> <b>Tip:</b> We recommend that you enter useful information in the <b>Description</b> field for easy future reference.</div>                                                                                                                                                                                                                                                                                                |
| Severity                   | Select a severity from the drop-down list. Choices are as follows: <ul style="list-style-type: none"><li>• High (most severe): High severity is divided into three categories — High 9, High 8, and High 7.</li><li>• Medium: Medium severity is divided into three categories — Medium 6, Medium 5, Medium 4.</li><li>• Low (least severe): Low severity is divided into three categories — Low 3, Low 2, and Low 1.</li></ul>                                                                                                                                              |
| Protection Category        | You must choose a Protection Category from the available options. The Protection Category indicates the intent of the attack and the intended target. For example, you can choose <b>Client Protection/Operating Systems</b> for an attack targeting vulnerabilities in client operating systems. In this example, Client Protection is the category and Operating Systems is a subcategory. The list of Protection Categories is pre-defined and provided by Trellix Advance Research Center. You cannot modify it. This list is updated when you update the Signature Set. |
| Detection Type             | Select the type of detection from the options that relisted.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| Attack Target              | Select the appropriate attack target.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Blocking                   | Select the appropriate blocking. You can either block only the attack packet or the entire flow.                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <b>Non-editable Fields</b> |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Benign Trigger Probability | This is an indication of the probability that the Snort Custom Attack will alert on traffic that may not be an attack. The default value is Medium, which you cannot modify.                                                                                                                                                                                                                                                                                                                                                                                                 |
| Attack Category            | This column indicates the type of attack.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Trellix IPS ID             | The numeric ID assigned for the attack by the Manager for database archival. The Manager assigns the ID after you save it in the Manager server. For Snort Custom Attacks, the IDs begin with 0xe. For Snort Custom Attacks created in the Central Manager, the IDs begin with 0xee.                                                                                                                                                                                                                                                                                         |

| Option                 | Definition                                                                                                                                                                                                                                                                                          |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Supported Device Types | You can apply a Custom Attack signature for just the available device types. The value for this field depends on what you select for the corresponding rule. You cannot edit this field at the attack level, but the Manager modifies it accordingly when you change it for the corresponding rule. |
| Last Updated           | Displays the time at which the signature was last updated                                                                                                                                                                                                                                           |

Signature creation interface

After you define the details for a Trellix IPS Custom Attack, you need to define one or more signatures for that attack. To access the signature creation interface, in the **New Custom Attack** interface, click on the **Signature-<signature name>** tab.

The signature creation interface presents the details related to the signature you are creating. These details are used to provide further information about the suspicious activity for which you are attempting to capture and analyze through Custom Attacks. By configuring each field, you are further refining the signature's search, which raises the probability your attack will successfully detect the desired activity, thus preventing false positives.

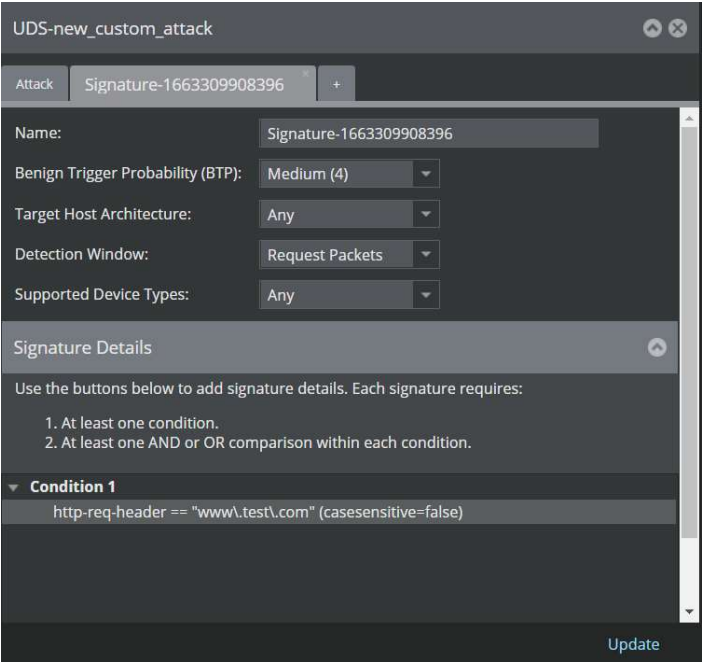


Figure 4-6 New Signature window

The signature detail fields are as follow:

| Option                           | Definition                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Name                             | The name you give to the signature. If the attack is detected, the Attack Log displays the detecting signature in the attack's details window.                                                                                                                                                                                                                                                                                                                                                                                                            |
| Benign Trigger Probability (BTP) | This is an indication of the probability that your signature will alert on traffic that may not be an attack. The choices are High, Medium, and Low, with Low representing approximately a 0% to 33% chance of your signature raising a false positive. For example, your signature may be a generic string search, such as "Confidential", thus the BTP would be best graded as High. If you know a specific file name that is infected or sensitive, such as "program.exe", you could set your BTP to Low to reflect your confidence in your signature. |
| Target Host Architecture         | You can define a specific machine architecture targeted by an attack with shellcode. For example, to detect a Intel-specific shellcode, you can select i386 as the architecture. The default is <b>any</b> .                                                                                                                                                                                                                                                                                                                                              |



| Option                        | Definition                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Detection Window</b>       | <p>This field describes where in a flow you want your signature to actively check traffic. The choices are as follows:</p> <ul style="list-style-type: none"> <li>• <b>Single Packet</b> — Check each packet for your signature condition(s). If your condition is xyz, and the Sensor sees xy in one packet and z in the next, no alert is generated. Each condition you specify must be found in one packet.</li> <li>• <b>Request Packet</b> — Check only the request direction of a flow.</li> <li>• <b>Response Packet</b> — Check only the response direction of a flow.</li> <li>• <b>Entire Flow</b> — Check the entire flow (between two hosts) for your condition(s). If your condition is xyz, and the Sensor sees xy in one packet and z in the next of the same flow, an alert is generated.</li> </ul>                                                                                 |
| <b>Supported Device Types</b> | <p>You can apply a Trellix IPS Custom Attack signature for any of the following device type options:</p> <ul style="list-style-type: none"> <li>• <b>NS-Series Only</b></li> <li>• <b>VM-Series Only</b></li> <li>• <b>Any</b></li> </ul> <p>The value for this field depends on what you select for the constituent signatures. For example, if the attack contains signatures with <b>Supported Device Types</b> set only to NS-series, then the value displayed here is <b>NS-series Only</b>. You cannot edit this field at the attack level, but the Manager modifies it accordingly when you change it for the corresponding signatures.</p>                                                                                                                                                                                                                                                   |
| <b>Conditions</b>             | <p>A condition is the test or group of tests that, if met, raises an alert. Conditions are made up of protocol field tests, or comparisons. A signature may have multiple conditions, each with multiple comparison tests. If you configure multiple conditions for a signature, each condition must be met in order (ANDTHEN logic) before an alert is raised.</p> <ul style="list-style-type: none"> <li>•  — Add a new condition.</li> <li>•  — Delete a selected condition.</li> </ul>                                                                                                                                                                                                                                     |
| <b>Comparisons</b>            | <p>Within a condition, you add one or more comparison tests. Comparisons can be AND or OR in nature. AND comparisons must be met for an alert to be generated for malicious traffic. OR comparisons allow for multiple comparisons within a condition, of which only one of the OR tests must be met to raise an alert. You must start a condition with an AND comparison. You may have up to 32 comparisons per condition.</p> <ul style="list-style-type: none"> <li>•  — Add an AND comparison.</li> <li>•  — Add an OR comparison. You must have already added an AND comparison.</li> <li>•  — Delete a selected condition.</li> </ul> |

See also

[Create a signature on page 1559](#)

[Create the signature for the example on page 1663](#)

[Create the signature on page 1655](#)

## Snort custom attack interfaces

You create Snort Format custom attacks using the Snort Rules Language. When you write a Snort rule in the Custom Attack Editor and save it in the Manager server, an attack definition is automatically created and the rule associated with this attack.

### Interface to write Snort rules

The interface for writing a Snort rule is a free-text editor within the Custom Attack Editor. To access the Add Snort Rule editor in the Custom Attack Editor, click the **Snort Format** tab.

Enter the Snort rule in the Add Snort Rule editor and click **Save**. The rule and the associated Custom Attack are saved in the Manager client. You can view this attack on the currently selected tab as well as on the **All Custom Attacks** tab.

Interface to edit snort custom attacks

To access the **Edit Snort Attack** interface, click a Snort Custom Attack .

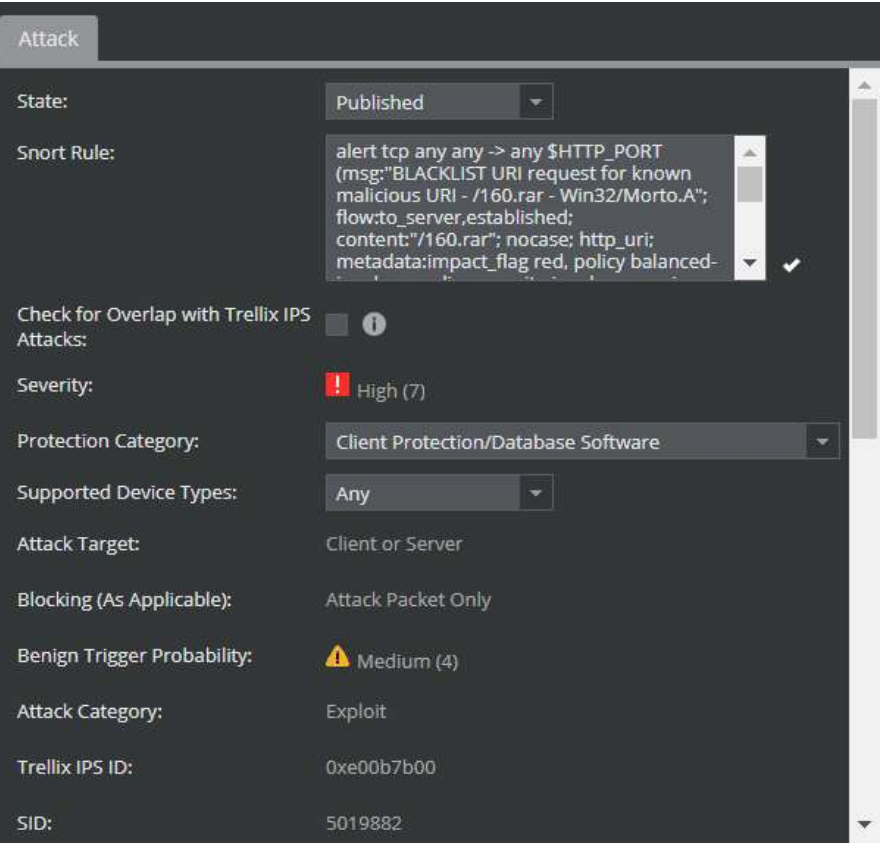


Figure 4-7 Edit Snort Attack window

When you create a Snort rule in the Manager, based on the elements of the rule, the Manager assigns values to some of the fields. If you modify the rule in the **Raw Snort Rule Text** section, the Manager modifies the field values accordingly. For example, if you change the msg in the rule, the attack and signature names also change because the Manager assigns the attack and rule names as per the msg keyword in the rule.

 **Note**

Every time you modify the rule in the **Raw Snort Rule Text** section, you need to check if the rule is conforming to Snort rules syntax by clicking . After the Manager translates the rule to Trellix IPS's format, click **Save** to save the changes in the Manager client. The changes are saved in the Manager server only when you click **Save** in the Custom Attack Editor.

The following the fields displayed on the **General** tab for a Snort Custom Attack:

| Option                                            | Definition                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|---------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Editable Fields</b>                            |                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <b>State</b>                                      | <p>Select the state of the custom attack. the choices are <b>Published</b> and <b>Staged</b>.</p> <ul style="list-style-type: none"> <li>Published — An attack in the Published state means that it is published in the policies of Trellix IPS.</li> <li>Staged — An attack in the Staged state means that it is not published in the policies of Trellix IPS.</li> </ul>                                                                 |
| <b>Snort Rule</b>                                 | You can modify the rule in this section.                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>Check for Overlap with Trellix IPS Attacks</b> | Verifies if the rule matches an existing Trellix IPS attack definition. If the rule matches then the Snort Rule will be <b>Staged</b> .                                                                                                                                                                                                                                                                                                    |
| <b>Protection Category</b>                        | Displays the Protection Category assigned to the attack. The Protection Category indicates the intent of the attack and the intended target. The list of Protection Categories is pre-defined and provided by Trellix Advance Research Center. You cannot modify it. This list is updated when you update the Signature Set.                                                                                                               |
| <b>Attack Target</b>                              | Indicates the target that is being exploited                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>Supported Device Types</b>                     | <p>You can apply a Snort Custom Attack signature for just the NS-series Sensors or Virtual IPS Sensors, or for both of them. The value for this field depends on what you select for the corresponding rule. You cannot edit this field at the attack level, but the Manager modifies it accordingly when you change it for the corresponding rule.</p> <p>By default, you can apply up to 5000 per NS-series and Virtual IPS Sensors.</p> |
| <b>Non-editable Fields</b>                        |                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <b>Severity</b>                                   | This priority is derived from the classtype or the priority tag.                                                                                                                                                                                                                                                                                                                                                                           |
| <b>Attack Target</b>                              | This column indicates the target that is being exploited.                                                                                                                                                                                                                                                                                                                                                                                  |
| <b>Blocking (As Applicable)</b>                   | In case of Snort Custom Attacks, the Sensor drops just the packet that matches the rule. You cannot edit this field.                                                                                                                                                                                                                                                                                                                       |
| <b>Benign Trigger Probability</b>                 | This is an indication of the probability that the Snort Custom Attack will alert on traffic that may not be an attack. The default value is Medium.                                                                                                                                                                                                                                                                                        |
| <b>Attack Category</b>                            | This column indicates the type of attack.                                                                                                                                                                                                                                                                                                                                                                                                  |
| <b>Trellix IPS ID</b>                             | The numeric ID assigned for the attack by the Manager for database archival. The Manager assigns the ID after you save it in the Manager server. For Snort Custom Attacks, the IDs begin with 0xe. For Snort Custom Attacks created in the Central Manager, the IDs begin with 0xee.                                                                                                                                                       |
| <b>SID</b>                                        | <p>Snort rule ID (SID) is the ID assigned to a Snort rule by you or the party that provided the rule. The Snort attack definitions that you want to save in the Trellix IPS database must have a unique SID. Make sure that the SIDs of the attacks that you are writing or importing have not been used by the definitions that are already in the database.</p> <p>For attack definitions that failed to import, SID is set to -1.</p>   |
| <b>Trellix IPS Snort Engine</b>                   | <p>The <b>Validation</b> field displays status of attack validation from the Trellix IPS Snort Engine.</p> <p>The <b>Test Compile</b> field displays the status of test compilation.</p>                                                                                                                                                                                                                                                   |
| <b>Suricata Snort Engine</b>                      | The <b>Validation</b> field displays status of attack validation from the Suricata Snort Engine.                                                                                                                                                                                                                                                                                                                                           |
| <b>Last Updated</b>                               | This is the time stamp when a rule was imported or modified.                                                                                                                                                                                                                                                                                                                                                                               |

To access the **Bulk Edit Snort Rules** interface, select multiple snort rules by pressing the **Ctrl** key.

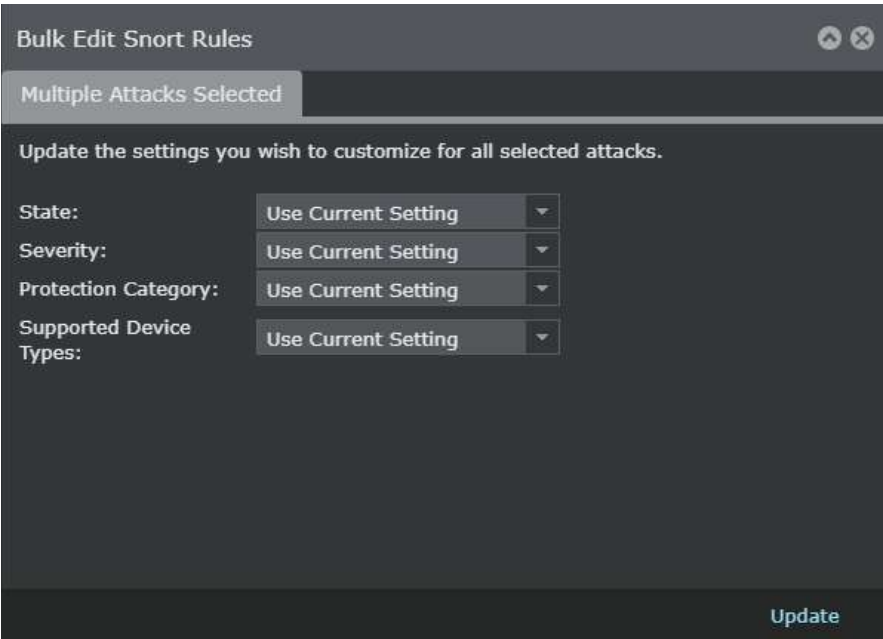


Figure 4-8 Bulk Edit Snort Rules window

The following the fields displayed on the **General** tab for a Snort Custom Attack:

| Option   | Definition                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| State    | <p>This column indicates whether a custom attack is published in the Trellix IPS policies. This column can have one of the following values:</p> <ul style="list-style-type: none"><li>Published — An attack in the Published state means that it is published in the policies of Trellix IPS.</li><li>Staged — An attack in the Staged state means that it is not published in the policies of Trellix IPS.</li><li>Use Current Setting — Retains the current value</li></ul>                                                                                |
| Severity | <p>Select a severity from the drop-down list. This column can have one of the following values:</p> <ul style="list-style-type: none"><li>High (most severe): High severity is divided into three categories — High 9, High 8, and High 7.</li><li>Medium: Medium severity is divided into three categories — Medium 6, Medium 5, Medium 4.</li><li>Low (least severe): Low severity is divided into three categories — Low 3, Low 2, and Low 1.</li><li>Informational: Informational has 0.</li><li>Use Current Setting: Retains the current value</li></ul> |

| Option                        | Definition                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Protection Category</b>    | You must choose a Protection Category from the available options. The Protection Category indicates the intent of the attack and the intended target. For example, you can choose <b>Client Protection/Operating Systems</b> for an attack targeting vulnerabilities in client operating systems. In this example, Client Protection is the category and Operating Systems is a subcategory. The list of Protection Categories is pre-defined and is provided by Trellix Advance Research Center. You cannot modify it. This list is updated when you update the Signature Set. |
| <b>Supported Device Types</b> | <p>You can apply a Snort Custom Attack signature for just NS-series Sensors or Virtual IPS Sensors, or for both of them. The value for this field depends on what you select for the corresponding rule.</p> <p>By default, you can apply up to 5000 Snort signatures per NS-series Sensors and Virtual IPS Sensors.</p> <p>This column can have one of the following values:</p> <ul style="list-style-type: none"> <li>• <b>NS-Series Only</b></li> <li>• <b>VM-Series Only</b></li> <li>• <b>Any</b></li> </ul>                                                              |

## Matching Criteria section

The **Matching Criteria** section enables you to categorize your attack for eventual submission to your Manager's attack database. Once exported to the attack database, selection of your attack is published by one of several rule sets, which are then added to policies for enforcement.

A rule set is defined as a set of ordered rules used to determine what attacks or conditions are of interest, and thus should be monitored when applied as part of a policy. A rule set is configured based on attack category (Exploit, Reconnaissance), operating system (Windows, UNIX), protocol (HTTP, DNS), application (SendMail, Apache), severity (High, Low), and benign trigger probability (High, Low) options. Except for benign trigger probability, you need to indicate these options when defining at the attack level. You can indicate the value for benign trigger probability when creating the signatures for an attack definition. Based on the values that you set for these options, the attack is available for policy enforcement.

- **Protocol** — This is the identified impact protocol for the Snort Custom Attack.
- **Software Package (OS)** — Manager assigns tcpip-machine to all Snort Custom Attacks. You cannot edit this value.

## Packet Grep protocol interface

Not all traffic can be supported by standardized protocol specifications. There are cases that require a different means of identifying an attack based on packet grepping. Trellix IPS has a pre-defined list of packet grep protocols. You can view this list of protocols in the Packet Grep interface. Using this interface, you can also create packet grep protocol in case the one that you need is not in the list of Trellix IPS-defined packet grep protocols.

To access the Packet Grep protocol interface in the Custom Attack Editor, select **Other Actions | Manage Grepping Protocols**.

## Trellix IPS default protocols

The Default section displays applications which support TCP, UDP, and IP that are commonly used/allowed, but do not use a standardized protocol and may be used for malicious purposes. The Custom Attack Editor enables you to monitor usage of programs such as NetMeeting and PCAnywhere in order to prevent external attacks. For example, your security policy may allow the use of PCAnywhere to resolve desktop/server issues remotely, but the program can also be used to infiltrate your network and perform malicious acts. The Custom Attack Editor enables you to create a pattern-matching signature for responses or requests to/from any of the listed programs for monitoring purposes.

To create a signature for any of these instances, simply create a signature instance, and select Packet Grep Protocol Match as a comparison for a condition.

Manage Grepping Protocols

Default

Custom

|    | Name              | Protocol | Ports     |
|----|-------------------|----------|-----------|
| 1  | Adobe-Flash-Media | TCP      | 1935      |
| 2  | AlphaStor         | TCP      | 41025     |
| 3  | AsteriskIAX2      | UDP      | 4569      |
| 4  | Bopup             | TCP      | 19810     |
| 5  | Borland           | TCP      | 3050      |
| 6  | FW1-mc            | TCP      | 256       |
| 7  | FujitsuSystemcast | UDP      | 4011      |
| 8  | IBM-Tivoli        | TCP      | 1582      |
| 9  | IBMInformix       | TCP      | 1526,9088 |
| 10 | IBMTivoliOSAgent  | TCP      | 10110     |
| 11 | IBMTivoliService  | TCP      | 18302     |

Save



Custom-defined packet grep protocols

Custom-defined packet grep protocols section is where you can view the list of custom-defined packet grep protocols and also create new ones. Custom-defined packet grep protocols can be used when creating a Native Trellix IPS Format custom attack with **Text in Custom Application** template.

-  — Add a new packet grep protocol instance.
-  — Delete a created instance.

 **Note**

You cannot modify a default custom-defined packet grep protocol.

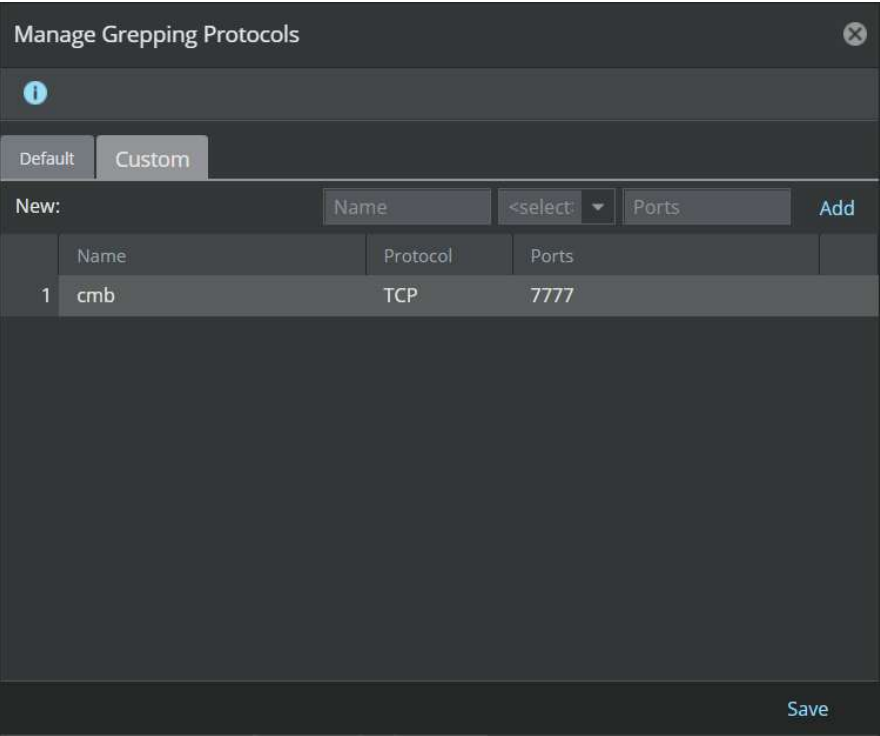


Figure 4-9 Custom-defined Packet Grep Protocols window

When you add a packet grep protocol, you need to specify the transport layer protocol and the identifying port(s) for locating those attacks that use an application protocol unknown to Trellix IPS.

The interface fields and options are as follows:

- **Name** — Name displayed during selection. For example, type "Custom-Adobe-Flash" for the Name.
- **Transport Protocol** — Specify the transport protocol that will be used by the application whose protocol details that you are defining.
  - TCP
  - UDP
- **Ports** — Target ports that have been added for the current custom protocol. Type the number in this field, and then click **Add** to add a port that can be used to identify the unknown traffic.



### Note

Do not enter any standard port number.

## Other Actions

Following are the other actions that can be performed using the custom attack editor.

### Work with packet search protocol

This section deals with creating a packet search protocol and a signature using your new packet search protocol.

#### Create a packet search protocol

Not all traffic can be supported by previously defined protocol specifications. Thus, in some cases, you may need to define a different means of separating out certain protocol traffic. The Packet Grep Protocol feature (Packet Search > Manage Packet Search Protocol) enables you to create pattern matching searches in a particular protocol traffic that is not separately parsed.

Packet searching is commonly known as packet grepping. The term grep refers to the Unix command "g/re/p," which executes a global search for a regular expression, and prints the lines that contain pattern matches. In essence, a protocol packet grep is a search for a fixed pattern in either the request or response of an unsupported protocol flow.

Protocols, such as HTTP and FTP, have defined specifications and destination ports. For example, HTTP primarily uses port 80, but is also common on port 8080. A Sensor monitoring traffic identifies the HTTP traffic by the protocol's common traits, and checks the traffic against standardized (RFC) HTTP traffic specifications. If packets are determined to be suspicious through signature tests, an alert is sent from the Sensor to the Manager. In the event that a Sensor does not recognize the protocol of a monitored transmission through signature, anomaly, or packet search tests, the packets are sent to a general state machine and examined against user-configured pattern-matching checks. If a pattern is matched, an alert is raised.

For example, port 1387 on one of your Payroll servers is receiving undetermined protocol connection requests atop TCP. You have confidential information on this server, so protecting your data from suspicious searches or requests is crucial. You can create a packet search instance in requests via TCP to port 1387, then create a Trellix IPS Custom Attack that defines a regular expression pattern that alerts upon being matched.

Several packet search protocol instances are provided with Trellix IPS. These applications use protocols that are not defined by an RFC nor by a defined Trellix IPS protocol specification. Some of these instances, such as pcAnywhere and NetMeeting, represent legitimate applications that are allowed by network policy within/into some networks, but which may be used for malicious purposes.

 **Note**

You cannot delete any of the Trellix IPS-defined packet grep protocols.

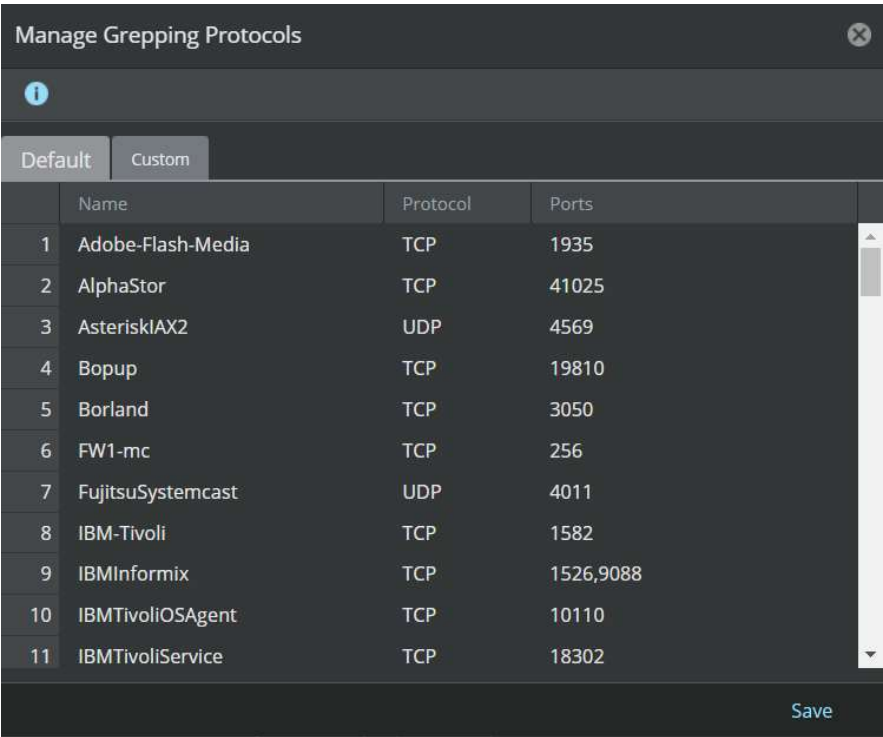


Figure 4-10 Manage Grepping Protocols window

See also  
*Create a signature* on page 1559

Create a packet search protocol instance

Several packet search protocol instances are provided with Trellix IPS. You can create a packet search protocol instance. These applications use protocols that are neither defined by an RFC nor by any Trellix-defined protocol specification.

Task

- 1 In the Custom Attack Editor, select **Other Actions | Manage Grepping Protocols**.  
The **Manage Grepping Protocols** dialog box opens.
- 2 Go to **Custom** tab.
- 3 Type a name. This name is listed in the "Select Protocol" step during signature creation.  
Type a verbose name. This name is for reference purposes.
- 4 Select the transport protocol as either **TCP** or **UDP**.

- 5 Type a port number in the **Ports** field, then click **Add**. Repeat for multiple port numbers.

 **Note**

You cannot add a port number that is already used by a defined protocol, such as 21 (FTP) or 80 (HTTP). Also, if you create a packet grep instance for port 888, you cannot create another packet grep instance for that port.

Manage Grepping Protocols

Default

Custom

New:

Name

<select>

Ports

Add

|   | Name | Protocol | Ports |
|---|------|----------|-------|
| 1 | cmb  | TCP      | 7777  |

Save

- 6 Click **Save** when finished.
- 7 Create a new signature instance that utilizes your created packet grep instance.

Create a signature using your new packet search protocol

Before you begin

Create a Trellix IPS Custom Attack definition and in the **Matching Criteria** section and select **tcpip-machine** as the **Software Package**.

Task

- 1 Click **Signature-<Signature ID>** tab.
- The **Signature Details** section is visible.

- 2 Add a comparison with at least one condition. Do the following:
  - a Click **Condition 1** so that it is highlighted.

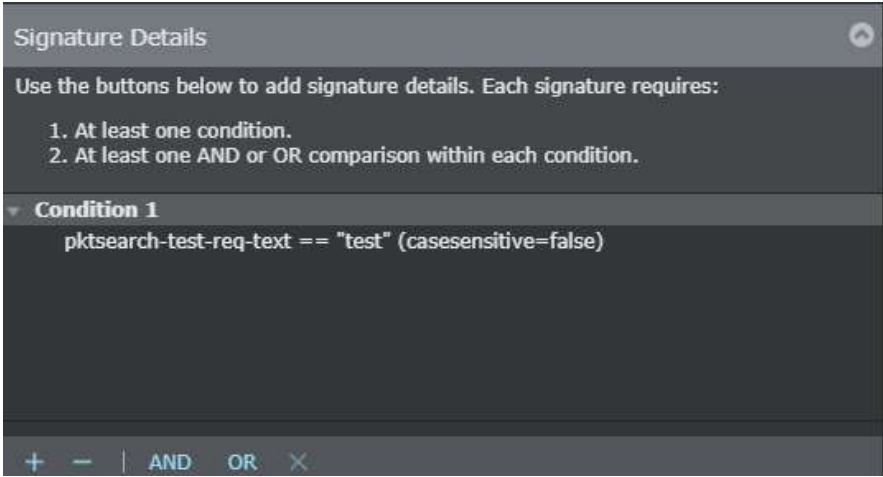


Figure 4-11 Creating a condition

- b Click **AND**.  
The Add ADD Comparison dialog box opens.

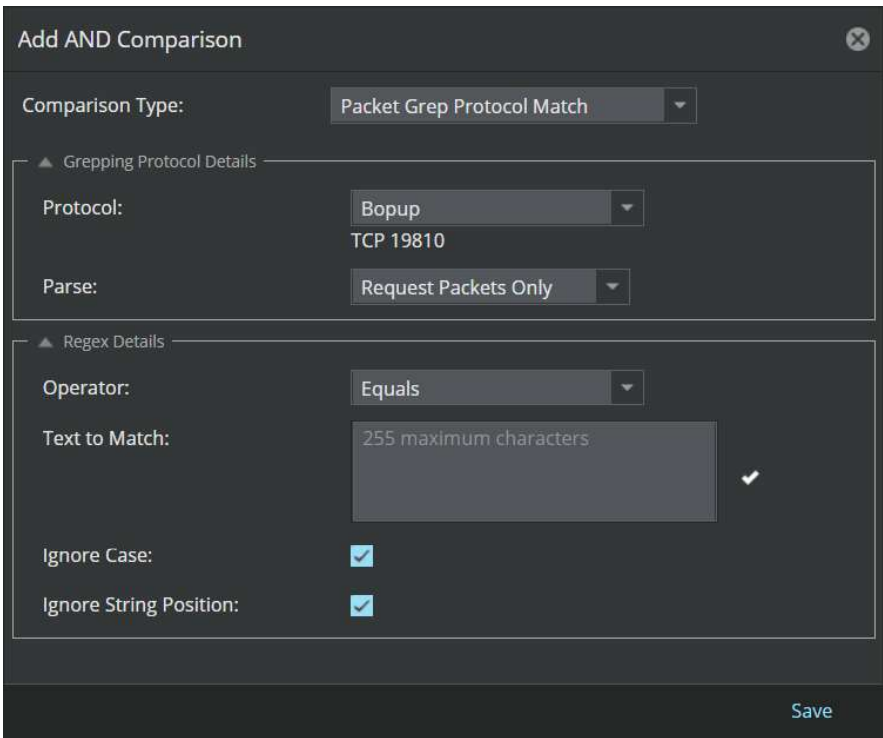


Figure 4-12 Adding comparators

- c Select **Packet Grep Protocol Match** from the **Comparison Type** drop-down menu.
    - d Select the protocol name that you defined from the **Protocol** drop-down menu.

- e Select the signature search direction from the **Parse** drop-down menu. The choices are as follows:
    - Request Packets Only
    - Response Packets Only
  - f Select the regular expression matching criteria as **Equals** or **Does NOT equal** from the **Operator** drop-down menu.
  - g Add a pattern to match in the **Text to Match** text box.
  - h Optionally, click the **Ignore Case** check box if you want the pattern to be matched regardless of [letter] case.
  - i Optionally, click the **Ignore String Position** check box.
  - j Click **Save** after adding the condition details. Your comparison appears under **Condition 1**.
- 3 Click **Update** in the attack editor window.
  - 4 Click **Save** in the **Custom Attacks** window.

### Import previously exported custom attacks

If you had exported any Trellix IPS Custom Attacks, you can import it back into the Manager. Note that this should be the same ZIP file that was created at the time of export. If this ZIP file contains any Snort Custom Attacks, those are imported as well. This feature enables you to import custom attack definitions created on a different Manager as well as the Trellix IPS-defined custom attacks.

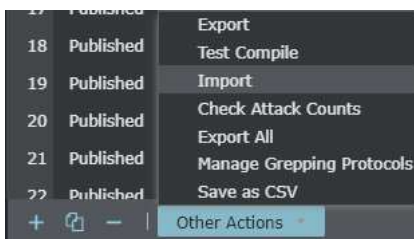
#### Note

When you import the ZIP file containing the exported attack definitions, only the attacks not present in the Manager are imported.

To import the previously exported Trellix IPS Custom Attacks:

#### Task

- 1 In the Custom Attack Editor, go to **Other Actions | Import**.



- 2 Browse to the location of your saved Trellix IPS Custom Attack ZIP file.
- 3 Click **Open**.

The imported attacks are listed in a new tab in the Custom Attack Editor. If the ZIP contains Snort Custom Attacks, those two types of attacks are listed on separate tabs.

- 4 If you are importing Native Custom Attacks:
  - a Select **Import Native Trellix IPS Attacks**.

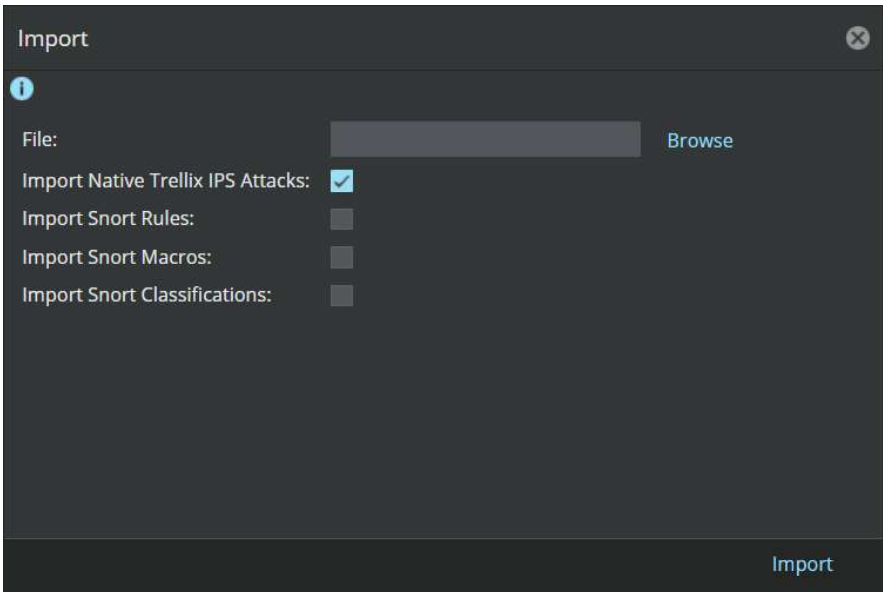


Figure 4-13 Import Native Trellix IPS Attacks

- b Deselect **Import Snort Rules**.
  - c Deselect **Import Snort Macros**.
  - d Deselect **Import Snort Classification**.
  - e Click **Import**.

If you are importing Snort Rules:

- a Deselect **Import Native Trellix IPS Attacks**.
  - b Select **Import Snort Rules**.

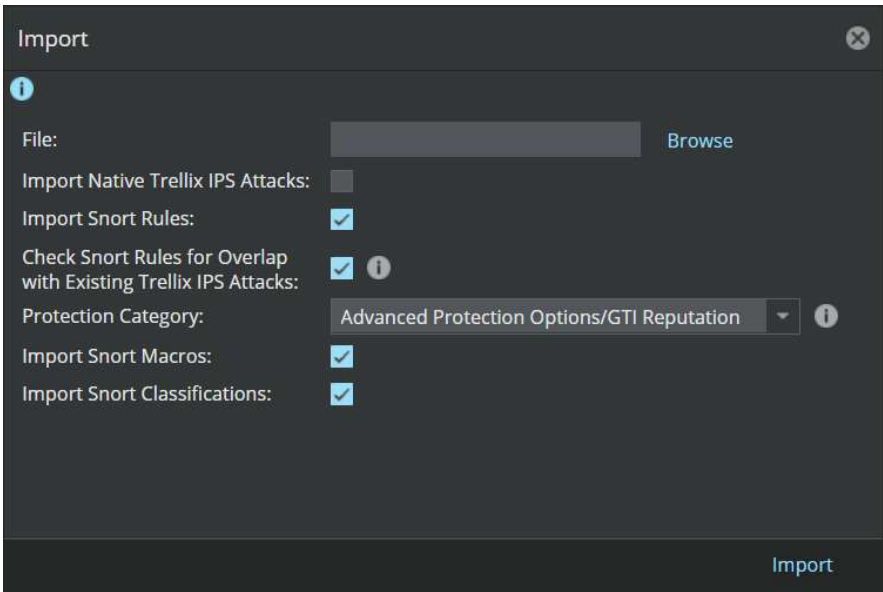


Figure 4-14 Import Snort Rules



- c** Select **Check For Overlaps With Existing Trellix IPS Attacks**.
  - d** Select a **Protection Category** value.
  - e** Select **Import Snort Macros**.
  - f** Select **Import Snort Classification**.
  - g** Click **Import**.
- 5** Verify if the attack is published in the policies.
- 6** Publish the attack with its signatures to the Sensors for attack detection.

See also

*[Viewing a policy to verify inclusion of the attack on page 1604](#)*

### Importing Trellix IPS-defined custom attacks

To quickly address the latest threats, Trellix IPS periodically provides attack definitions before they are incorporated in official signature sets. These Trellix IPS-defined custom attacks (ID-UDS) are automatically deprecated when they are incorporated in a regular signature set.

#### **Note**

When an ID-UDS is deprecated, the Manager generates an **Informational** alert that you can view in the Attack Log.

#### **Note**

You cannot edit the attributes of an ID-UDS using the Custom Attack Editor.

You can export and import an ID-UDS just like you would with any other Custom Attacks.

### Importing snort custom attacks

Using the Custom Attack Editor, you can import Snort rules from a file against constructing them one by one. Sources such as the Snort user community distribute files containing the rules which you can directly import. The import feature is helpful when you want to create a large number of Snort Custom Attacks in one go.

You can use files to import Snort rules into the Manager following any one of the two methods described below:

- Import a .zip file that contains all the required files.
- Import files individually in a specific order.

If you are importing the files individually, they have to be imported in the following order:

- 1** Import the Snort Configuration file (.conf)
- 2** Import the Snort variable files
- 3** Import the Snort rules files (.rules)

Check for overlaps with Trellix IPS attacks

When you create a Snort rule in the Custom Attack Editor or when you import Snort rules, the Custom Attack Editor checks if there is an equivalent Trellix IPS attack signature based on the CVE IDs. If a Trellix IPS attack signature exists, the corresponding Snort rule is referred to as a *duplicate Snort custom attack* or *duplicate Snort rule*. Such rules are imported into the Custom Attack Editor but staged in the policies by default when you save them. You can manually change the **State** of such rules to **Published**.

Alternatively, you can specify your preference in the Custom Attack Editor so that, going forward, the duplicate Snort custom attacks are published in the corresponding policies by default.

When you publish duplicate Snort custom attacks and enable alerting for the corresponding Trellix IPS signature as well, two alerts might be raised for the same attack traffic — one triggered by the Snort custom attack and the other by the Trellix IPS attack.

Task

- 1 In the Custom Attack Editor, click .
- 2 Select Check for Overlap with Trellix IPS Attacks.

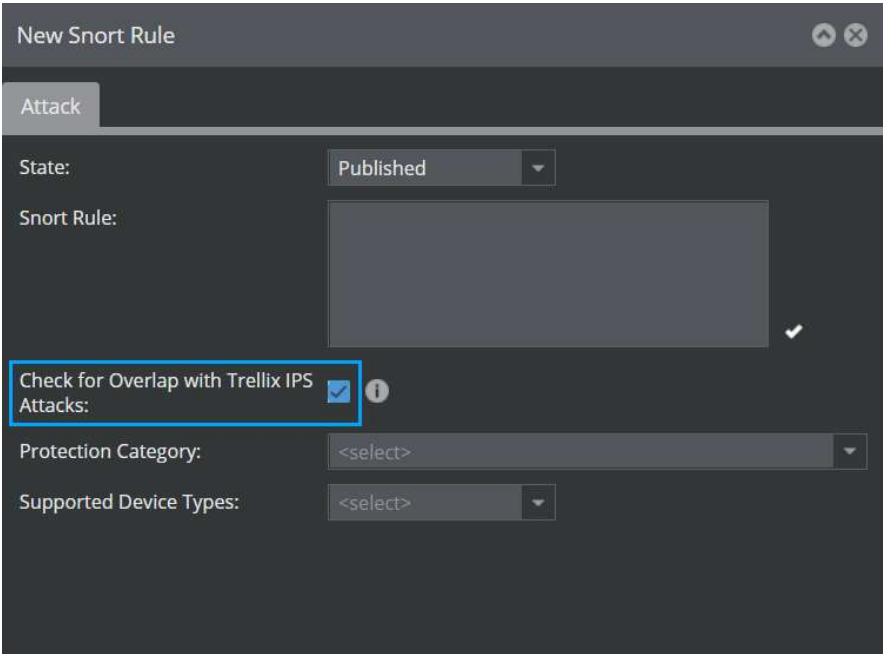


Figure 4-15 Publish duplicate Snort custom attacks

- This selection applies only to the Snort custom attacks that you will create or import from now on and not to the already staged duplicate Snort custom attacks. You can only manually publish the previously staged duplicate Snort custom attacks by changing the **State** of the corresponding attack.
- This feature compares only the CVE IDs in Snort rules with the CVE IDs in the Trellix IPS attack signatures.
- The CVE ID that you mention in the Snort rule must be of the universal format for this feature to work.

| Correct                      | Incorrect                                                                                                                                                                                         |
|------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| reference:cve,<br>2010-0249; | reference:cve, CVE-2010-0249; The rule is published even though there is an equivalent Trellix IPS signature because the Custom Attack Editor looked for CVE-2010-0249 instead of just 2010-0249. |

Import snort rules through a conf file

Before you begin

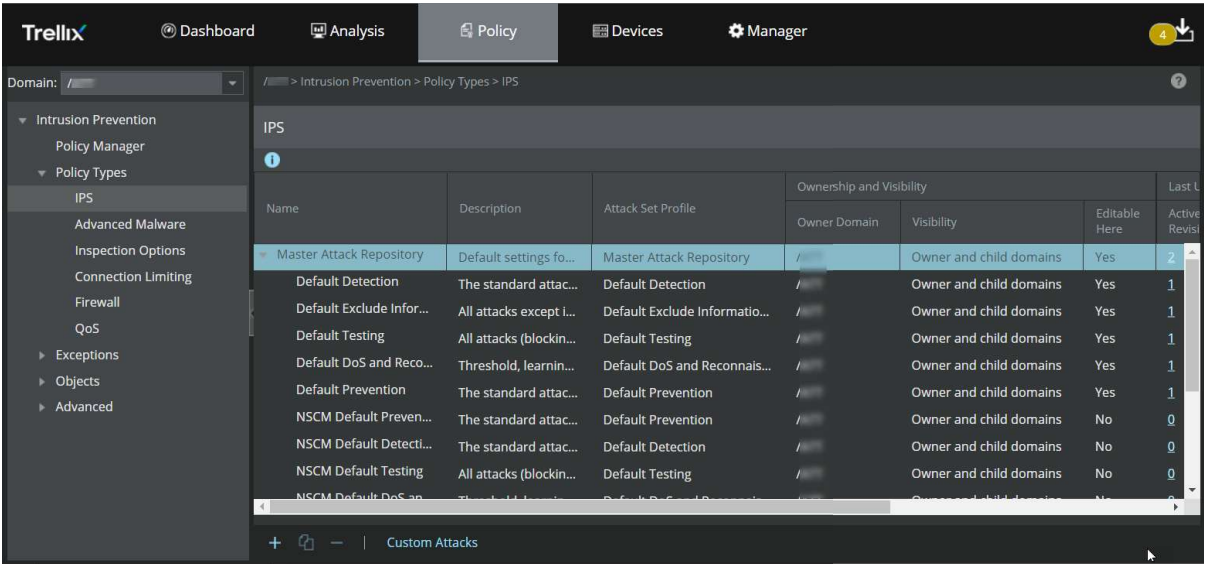
Make sure of the following before you begin importing the conf file:

- You have all the rules files containing the rules that you want to import.  
The conf file has references to each rules file that you want to import. That is, the rules files are called from the config file using the include keyword and the absolute or relative path to the files. You can also use a variable to denote the path.
- All the variables, classifications, and references used in the rules are either defined in the conf, or the rules files up front, or available already in the Manager.
- Depending on how the rules have been constructed, the conf file may be referencing some files in addition to the rules files. Make sure all the files called by the conf file are in place.
- Each rule must have a globally unique Snort rule ID (SID) for it to be converted and saved in the Manager database. A rule is not considered for import, if the SID and revision number are same as that of a rule imported earlier. If the SID is same but not the revision number, then the rule with the highest revision number is retained.

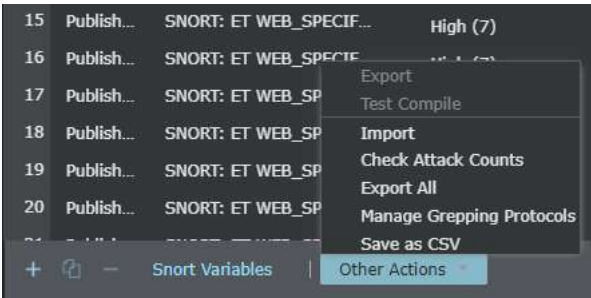
This section provides the steps for importing Snort rules into the Manager using a conf file.

Task

- 1 Select Policy | <Admin Domain Name> | Intrusion Prevention | Policy Types | IPS. Click Custom Attacks.



2 In the Custom Attack Editor of the Snort Format tab, click Other Actions | Import .



3 Navigate to the .conf file to be imported.

4 Click Open.

All the rules are imported into the Manager and the valid ones are converted to Trellix IPS's format. There could be some rules that were successfully converted to Trellix IPS's format, some converted with warnings, and some that failed to convert.

5 Select a Protection Category value.

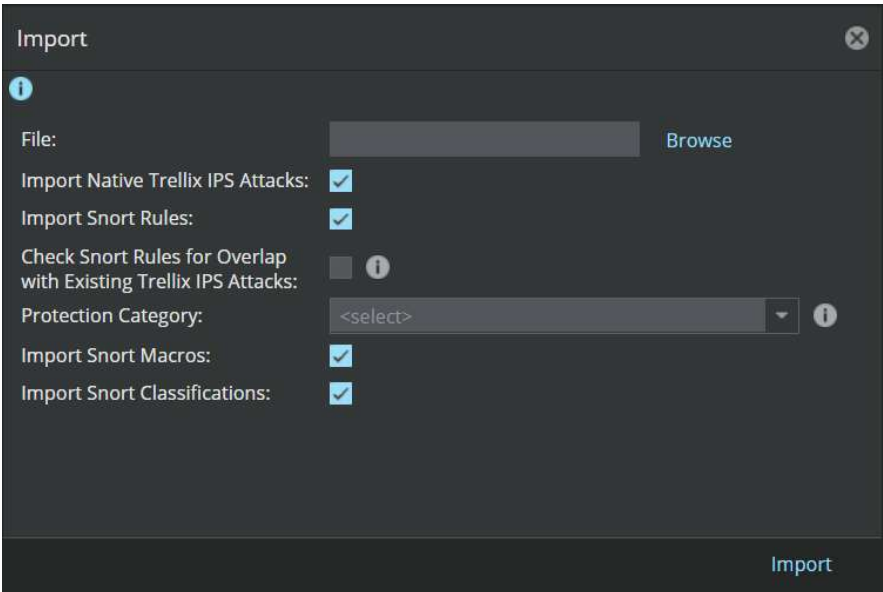


Figure 4-16 Import rules window

6 Click Import.

- 7 You can refer to the section [Managing Snort rules](#) on page 1539 to:
- View the details of the imported Snort rules
  - Know which rules converted successfully, which converted with warnings, and which failed to convert

See also  
[Variables](#) on page 1595  
[Managing Snort rules](#) on page 1539

Import snort rules through a rules file

Before you begin

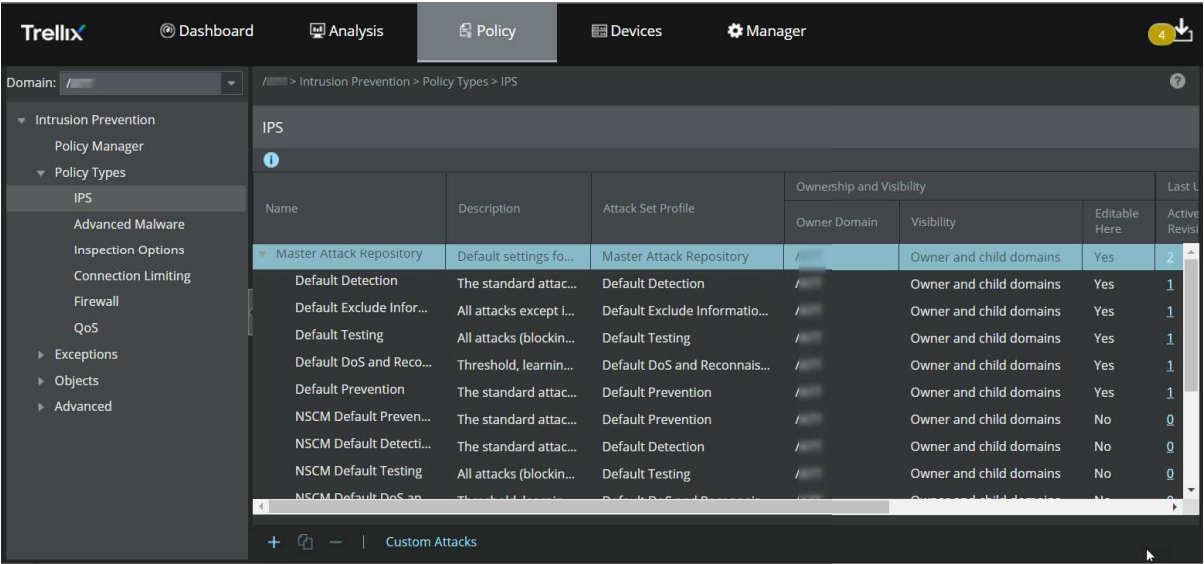
Make sure of the following before you begin importing a rules file:

- All the variables, classifications, and references used in the rules are either defined in the rules files up front or already available in the Manager.
- Just like a conf file, a rules file too can call other files. So, make sure the files called by a rules file are in place.

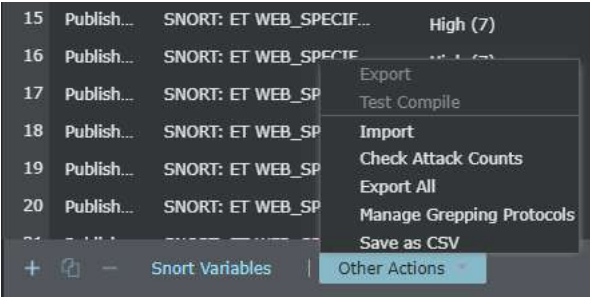
You can import rules directly from a rules file without a conf file. This section provides the steps for importing Snort rules into the Manager using a rules file.

Task

1 Select Policy | <Admin Domain Name> | Intrusion Prevention | Policy Types | IPS. Click Custom Attacks.



2 In the Custom Attack Editor of the Snort Format tab, click Other Actions | Import .



3 Navigate to the .rules file to be imported.

**4 Click Open.**

All the rules are imported into the Manager and the valid ones are converted to Import snort rules through a conf file's format. There could be some rules that were successfully converted to Import snort rules through a conf file's format, some converted with warnings, and some that failed to convert.

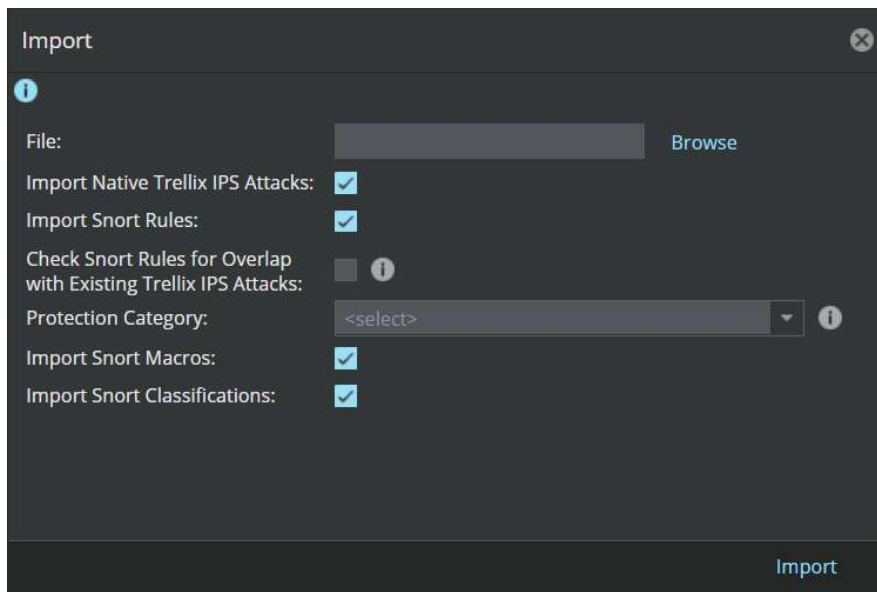
**5 Select a Protection Category value.**

Figure 4-17 New Signature window

**6 Click Import.****7 You can refer to the section [Managing Snort rules](#) on page 1539 to:**

- View the details of the imported Snort rules
- Know which rules converted successfully, which converted with warnings, and which failed to convert

See also

[Variables](#) on page 1595

[Managing Snort rules](#) on page 1539

Managing Snort rules

All Snort rules are managed on a dedicated tab in the Custom Attack Editor. This section briefly describes the management of Snort rules in the Custom Attack Editor.

Intrusion Prevention > Policy Types > IPS

Custom Attacks

Native Trellix IPS Format

Snort Format

Quick Search

Clear All Filters

|    | State     | Name                          | Severity   | BTP        | Attack Category | Trellix IPS ID | SID       | Trellix IPS Snort Engine |              | Suricata   |
|----|-----------|-------------------------------|------------|------------|-----------------|----------------|-----------|--------------------------|--------------|------------|
|    |           |                               |            |            |                 |                |           | Validation               | Test Compile | Validation |
| 1  | Published | SNORT: BLACKLIST URI req...   | High (7)   | Medium (4) | Exploit         | 0xe00b7b00     | 5019882   | Success                  | Success      | Success    |
| 2  | Staged    | SNORT: SPECIFIC-THREATS ...   | High (7)   | Medium (4) | Exploit         | 0xe0051700     | 5018078   | Warning                  | ---          | Success    |
| 3  | Staged    | SNORT: SPYWARE-PUT Ad...      | Low (1)    | Medium (4) | Exploit         | 0xe009e400     | 505753    | Warning                  | ---          | Success    |
| 4  | Staged    | SNORT: FTP LIST buffer ove... | Medium (4) | Medium (4) | Exploit         | 0xe0021300     | 502338    | Warning                  | ---          | Failed     |
| 5  | Staged    | SNORT: EXPLOIT IBM DB2 ...    | High (7)   | Medium (4) | Exploit         | 0xe0086700     | 5019206   | Warning                  | ---          | Failed     |
| 6  | Staged    | SNORT: RPC portmap CA B...    | Medium (4) | Medium (4) | Exploit         | 0xe0069400     | 5010484   | Warning                  | ---          | Success    |
| 7  | Published | SNORT: BOTNET-CNC kno...      | High (7)   | Medium (4) | Exploit         | 0xe0003000     | 511016820 | Success                  | Success      | Success    |
| 8  | Staged    | SNORT: WEB-MISC Ipswitc...    | High (7)   | Medium (4) | Exploit         | 0xe0038000     | 5017279   | Warning                  | ---          | Failed     |
| 9  | Published | NSCM-SNORT: ET TROJAN ...     | High (7)   | Medium (4) | Exploit         | 0xee028c00     | 2008005   | Success                  | Success      | Success    |
| 10 | Staged    | SNORT: SPYWARE-PUT Hija...    | Low (1)    | Medium (4) | Exploit         | 0xe0110c00     | 507534    | Warning                  | ---          | Success    |

+ -

Snort Variables

Other Actions

5562 (of 5562) attacks showing

Custom Attacks

Save Cancel

Figure 4-18 Managing Snort rules in the Custom Attack Editor

For description of the columns see Tab regions on page 1514.

See also  
[Import snort rules through a conf file](#) on page 1535  
[Import snort rules through a rules file](#) on page 1537

Correct the failed conversions

This sections explains how to fix Snort rules that fail to convert. Rules that fail to convert after import are set to **Failed** in the **Validation Result** column.



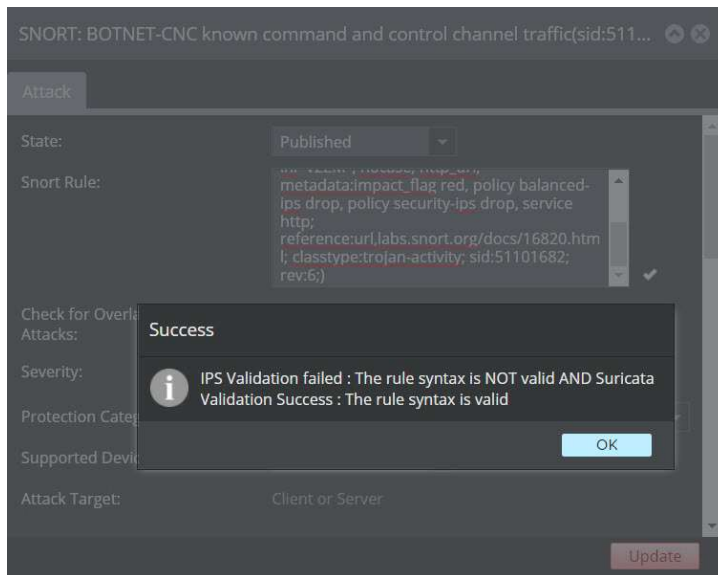
Perform the following steps to troubleshoot a failed rule:

### Task

1 Double click on the failed rule in Custom Attack Editor. The Edit Snort Attack window opens.

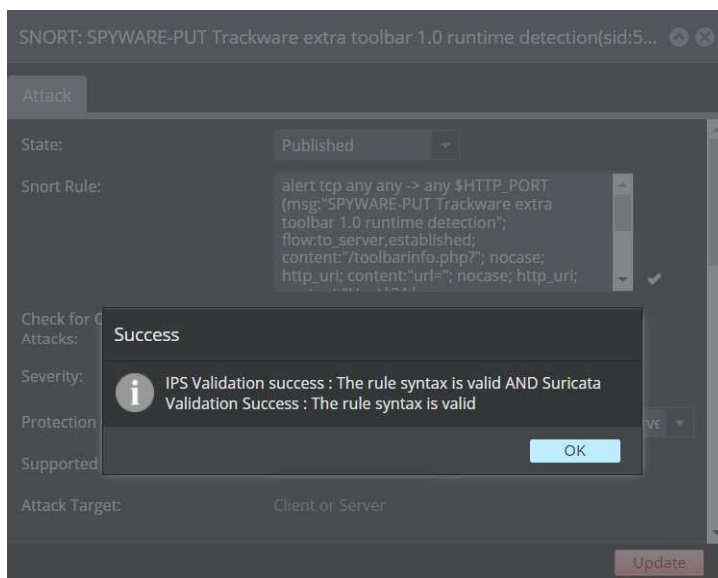
2 Click  to validate the rule.

For example, in the image, the validation error is related to an incorrect input type in the rule.



3 Update the rule to resolve the error.

4 Click  to validate the rule.



5 If the rule is validated successfully, a **Success** pop-up appears. Click **OK**.

6 Click **Save** to save the changes you made.

The rule is now converted to Trellix IPS's format and stored in the Manager database.



### Note

After the rules are corrected, you can use the **Test Compile** feature to check the validity and compatibility. To validate the attacks, select the required attacks, right-click and select **Test Compile**.

### Considerations for rules converted with warnings

Some valid rules may convert with conversion result as **Warning**. The conversion result is displayed in the **Validation Results** column. By default, rules with **Warning** are set to **Staged**. There are two reasons why rules convert with a warning.

- Rules for which there is an equivalent Trellix IPS signature in the Manager are converted with warnings. The Manager identifies such rules based on the CVE ID mentioned in the rule and the signature. You can click on an excluded attack and change its State to **Published**. You can specify your preference in the Custom Attack Editor so that, going forward, the duplicate Snort custom attacks are published in the corresponding policies by default. For information on how to set this preference, see the section [Check for overlaps with Trellix IPS attacks](#) on page 1534.
- Rules that do not have certain rule options are converted with a warning. However, based on your requirement, you can set such rules to **Published** state. It is recommended that you investigate such rules to check if they impact Sensor performance. If so, fix the rules before you publish them in the rule sets.

### Test compile custom attacks

The Snort rules or User-Defined Signatures (UDS) that you create or import, must be compatible with the Trellix IPS signature set. When you save the custom attacks in the Custom Attack Editor, the Manager validates the rules to check their compatibility. If there are any incompatible custom attacks, a critical system fault message **Incompatible custom attack** is raised as a system fault. You can view the system faults in the **Manager** | **<Admin Domain Name>** | **Troubleshooting** | **Logs** page. After you correct the incompatible custom attacks, you can use **Test Compile** to verify the compatibility of the rules before you save them. For subsequent signature set updates, only attacks that pass compilation are published to the Sensors.

Consider the following when running **Test Compile**:

- Running **Test Compile** does not change the compilation status of attacks. The compilation status of attacks is updated only after you save the attacks. After you save, attacks that fail compilation are marked as **Failed** in the **Test Compile** column and will be in **Staged** state.

For information on rules that determine the state of the attack, see [Tab regions](#) on page 1514.

- You need to select the attacks that should be compiled.
- Attacks only in the **Published** state can be compiled. As an exception, if all the selected attacks are in **Staged** state, the attacks are compiled.

- Attacks that fail the test compilation are not highlighted by the Manager. To identify the specific attacks that fail, you can run **Test Compile** on a subset of attacks and verify if they fail. You can continue this process of elimination until you identify the specific attacks that fail.

For example, select 10 attacks for test compilation and compilation reports a failure. To identify the specific attacks that failed, select a subset of attacks from the initial 10 attacks and run test compilation. If test compilation still reports failure, repeat the process of elimination until you find the specific attacks that fail.

- Attacks from the Central Manager are not considered for test compile in the Manager.

### Note

The Manager compiles all the active custom attacks during reboot or startup. The **IPS** page is not available till the Manager finishes compilation. It takes about 7 minutes for the page to load.

### Task

**1** In the Custom Attack Editor, select the custom attacks you want to run a test compilation on.

**2** Select **Other Actions | Test Compile**.

This option is available as part of the right-click menu.

## Mechanics of a custom attack

This section explains the elements that make up a Trellix IPS Custom Attack. A good understanding of these elements and how they can impact your IPS setup is required to create effective attack definitions.

See also

[Create custom attacks on page 1547](#)

## Structure of a custom attack

A Trellix IPS Custom Attack is like a top-level container in which you define one or more signatures. Some of the critical values that you need to specify when defining an attack are the attack severity, the impact package, and protocol. Based on these values, the Manager categorizes the attack and includes it in the policies.

### Note

You must specify at least one impact package or protocol to save a custom attack in the Manager server.

## Structure of a signature

Trellix IPS's signature support is very comprehensive. In addition to the normal pattern matching and numeric tests supported by many network IPS systems, Trellix IPS allows signatures to have a complex structure that can order and group the tests necessary for accurate detection.

### How tests are logically structured

Trellix IPS provides a simple but powerful set of logical constructs that will allow you to create an attack definition to match the fingerprint of nearly any network activity. The logical constructs provide methods for grouping and ordering the more granular pieces of the signature - numeric tests and pattern matches - in any order you require. The core constructs are:

- **AND** - Tests are grouped within AND blocks. The tests must all succeed, but may occur in any order inside the context of a given flow. AND blocks cannot be contained within an OR condition.
- **OR** - The first test to succeed fulfills the condition, and the next AND or OR block is engaged. OR constructs are contained within AND blocks, though at times they may be the only contents.
- **ANDTHEN** - ANDTHEN is the ordering construct. By placing ANDTHEN statements between AND blocks, you can create a structured order for your tests, if required.

The actual tests of which signatures are composed are placed inside either AND or OR constructs. These are called conditions and are like containers. ANDTHEN is used to link the AND containers (conditions) together to form a larger structure.

### Supported test methods

A Trellix IPS Custom Attack supports 2 basic classes of test:

- **Pattern Match** - Pattern matches can be used to match ASCII or binary strings. Support for common regular expression constructs is available.
- **Numeric Comparisons** - Numeric comparisons can be used to test for a match with a given value or a range of values, depending on the numeric comparison test chosen.

### Other signature components

In addition to the mechanics of the signature itself, Trellix IPS requires a few other pieces of information, as described below, to do its job properly:

- **Detection Window** - The detection window is the portion of the traffic flow that the Sensor should check against an attack definition for matches. The detection window may be one of the following:
  - **Packet** - All of the signature tests (and thus all protocol fields) must occur in a single packet on the network.
  - **Request** - All of the signature tests must occur in the request direction of the flow. (Useful if the protocol is bidirectional and might include the same data or commands in either direction.)
  - **Response** - Identical to request, but used to examine only response-direction traffic
  - **Flow** - When the detection window is set to flow, all conditions can be fulfilled by traffic flowing in either direction. Tests are still subject to any ordering imposed by the structure of the signature or flow direction implied by the field being tested.

#### Note

It is possible to create a correct signature that will never trigger if the detection window is set incorrectly. This should be one of the first items you check when troubleshooting a custom attack definition.

- **Benign Trigger Probability** - This is a measure of the confidence with which you believe the signature can identify a network event. A high benign trigger probability (BTP) implies that a signature may be prone to false positives. When creating the signature, you should set the BTP to "Low" to ensure that your signatures are selected by the Default Policy.

### Create more than one signature per attack

The capability of creating more than one signature in an attack definition is part of what allows Trellix IPS to keep its false positive rate so low. You should create multiple signatures for an attack, when possible, so as to be both more inclusive and more specific. You may want to create a generic signature that will catch all attack variants (including those which are unknown

to you). You should also create more specific signatures to detect particular pieces of exploit code, if possible. This can be valuable in providing pointers on what to investigate if a machine has been compromised, in addition to keeping your rate of false positives far below that of the average signature-based IPS.

Signature test reference

This section describes each test type available for use in signatures and the required and optional parameters for each.

See also  
*Create a signature on page 1559*

Numeric comparison tests

- **!=** : Test of numeric inequality
- **==** : Test of numeric equality
- **unsigned <** : True if the unsigned **Value** specified is less than the value obtained from the network traffic
- **signed <** : True if the signed **Value** specified is less than the value obtained from the network traffic
- **unsigned >** : True if the unsigned **Value** specified is greater than the value obtained from the network traffic
- **signed >** : True if the signed **Value** specified is greater than the value obtained from the network traffic
- **unsigned <=** : True if the unsigned **Value** specified is less than the value obtained from the network traffic
- **signed <=** : True if the signed **Value** specified is less than or equal to the value obtained from the network traffic
- **unsigned >=** : True if the unsigned **Value** specified is greater than or equal to the value obtained from the network traffic
- **signed >=** : True if the signed **Value** specified is greater than or equal to the value obtained from the network traffic

Numeric test parameters

| Parameter      | Description                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| Protocol Field | The field that will be compared with the supplied value                                                                                    |
| Value to Match | The value that should be compared against the value retrieved from the network traffic. This field does not support floating point values. |

Numeric range tests

- **signed-in-range** — True if the value obtained from the network traffic falls within the specified signed range
- **unsigned-in-range** — True if the value obtained from the network traffic falls within the specified unsigned range



Tip

Trellix recommends using an in-range test as it is more efficient than -gt and -lt tests.

### Numeric range test parameters

| Parameter      | Description                                                                                                                                     |
|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| Protocol Field | The field that will be tested for equality with the supplied value                                                                              |
| Minimum Value  | Minimum value of the comparison range (Inclusive)                                                                                               |
| Maximum Value  | Maximum value of the comparison range (Inclusive)<br>Note that the Minimum Value and Maximum Value fields do not support floating point values. |

### Numeric enumeration tests

- **unsigned ==** : True if one of the enumerated values matches the value obtained from the network traffic

### Numeric enumeration test parameters

| Parameter      | Description                                                                                                                                 |
|----------------|---------------------------------------------------------------------------------------------------------------------------------------------|
| Protocol Field | The field that will be tested for equality with the supplied value                                                                          |
| Integer values | Integer values to be compared against the values derived from network traffic<br>Note that the values cannot be floating point or negative. |

### Pattern matching

Pattern matching within signatures is always implemented with string-match. String-match accepts a number of optional and required parameters. In the **Text to Match** text box, construct the string that contains the pattern to match. By default, pattern matches are case-sensitive. String-match accepts regular expressions in addition to simple strings.

### Pattern match test parameters

| Parameter              | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Protocol Field         | The field that will be tested for a match with the supplied pattern                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Text to Match          | Pattern to be matched                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Ignore Case            | Select this option to make the expression case insensitive. By default it is case sensitive.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Ignore String Position | Selecting this option enables you to create specific conditions.<br>Offset: Use this to indicate where the Sensor should begin its search for the string. For example, if you set an offset of 5, then the Sensor checks for first byte of the string from the 5th byte of the payload.<br>Depth: Use depth to indicate how far into the traffic should the Sensor look for the string. For example, if you set a depth of 10, the Sensor stops looking for the first byte of the string at the 10th byte from the beginning of the payload.<br>Origin: Select <b>Packet</b> if you want the Sensor should apply the offset or depth to a packet. If you want the Sensor to apply the condition to the entire flow, select <b>Flow</b> . |

### Create a fixed field signature

*Fixed fields* are those protocol fields that are provided by the basic IP protocols ("IPv4"), TCP, UDP and ICMP. They are "fixed" by contrast with the calculated nature of many provided by other protocol specifications. Due to the fact that these fixed fields occur in most packets the Sensor inspects, it is impractical for performance reasons to inspect them in the same fashion as standard protocols. For the same reason, the number of fixed field comparisons is limited to 50.

Two types of tests for fixed fields are available. Both are numeric because the fixed fields are all numeric.

- **fields-eq** — Fields-eq is a special test that is only available for fixed-field testing. It allows you to compare the values in two protocol fields. This is useful if, for example, you want to compare whether the source and destination IP addresses specified by packet headers are the same.
- **All of the numeric tests described for normal signature tests** — This includes both signed and unsigned tests.



#### Note

The **Single Fixed Field Match** in the Custom Attack Editor also provides an "bitmask" function. This is not a normal "bitmask". Instead, starting from the offset you provide (in the selected fixed field), it selects the number of bits you specify, and then uses those bits to create an integer value that is compared against your provided comparison value.

### Packet search protocols

*Packet search protocols* are a form of generic packet inspection capability used by IPS experts when dealing with protocols that are not specifically supported by Trellix IPS. They are generally used for implementing detection on specific ports in an efficient fashion. In these cases, it can be valuable to separate out traffic running on particular ports for processing. This can make detection more efficient because pattern matches and other tests will only examine a limited subset of network traffic. In addition, it is possible to create a packet search protocol to exclude certain traffic from being considered for detection.

### Performance issues

- **False positives** — Large numbers of false positives could (in severe cases) overwhelm the Manager. More commonly, they will simply overwhelm the user due to volume and speed of appearance.
- **Too many pattern matches** — If the pattern your signature searches for occurs too often in the network traffic being analyzed, checking each match to determine whether it fulfills a signature condition may degrade Sensor performance.



#### Note

The most common cause of overwhelming numbers of pattern matches is making them too short. A pattern to be matched should never be shorter than 3 characters because the likelihood of a chance occurrence of such a pattern in normal traffic is very high.



- **Too many fixed-field tests** — If you configure too many fixed-field tests in your signatures, Sensor performance may be degraded. This is due to the fact that the Sensor will have to inspect this field in every packet of the traffic type being tested. As with pattern matches, checking each test result to see if it fulfills a signature condition may affect Sensor performance.

### Note

The Fixed Field test will have a performance impact as every packet needs to get inspected for all the conditions configured by the admin. Therefore, the Sensor performance degrade due to the Fixed Field Tests is directly proportional to the number of fixed field test instructions configured. **Trellix recommends** to factor in the performance degrade while configuring the Fixed Field Tests in your signatures.

- **Testing overly common fields** — If your signature contains a test of a field that occurs very often in the traffic type being analyzed, it might degrade Sensor performance. This can be due to the number of tests that will be performed, or due to the number of tests that must be checked for fulfilling a signature test condition.

## Considerations

Consider the following when you create Trellix IPS Custom Attacks. These should not present an obstacle to most signature writers.

- You can save up to **4500** Trellix IPS Custom Attack definitions in your Manager.
- Each Trellix IPS Custom Attack may not have more than **16** different signatures.
- Each signature may not contain more than **32** different tests, either numeric or pattern-matching.

## Creating custom attacks

This section provides the information required to create and maintain Trellix IPS Custom Attacks with one or more signatures.

### Create custom attacks

The following is the high-level recommended approach for creating Trellix IPS custom attacks.

#### Task

- 1 Be clear on what you are trying to achieve through these attacks. Identify the related protocols, applications, hardware and software platforms, and so on for the attack.
- 2 Understand the mechanics of a Trellix IPS custom attack. You also need to understand the probable impact of your custom attack on the network.
- 3 Take a quick tour of the Custom Attack Editor and get familiar with it. You can use the Custom Attack Editor to manage the custom attacks.
- 4 Create a Trellix IPS custom attack with one or more signatures.
- 5 Verify if the attack is published in the policies.
- 6 Push the attack with its signatures to the Sensors for attack detection to happen.

See also

[Getting started with custom attacks on page 1506](#)

[Mechanics of a custom attack on page 1542](#)

[Trellix IPS custom attack creation interfaces on page 1518](#)

[Viewing a policy to verify inclusion of the attack on page 1604](#)

### Templates for Trellix IPS custom attacks

You can use the predefined templates to create some of the commonly used Trellix IPS custom attacks. By using these templates, you can create effective Trellix IPS custom attacks even if you do not possess detailed knowledge of the related protocol, its header, or the syntax of Trellix IPS custom attacks.

Predefined templates are available to create Trellix IPS custom attacks that:

- Detect a URL
- Detect an email attachment file name
- Detect a DNS query or response
- Detect a string in an application running over a custom port
- Detect a TCP connection attempt from a specific IP address

When you use the templates, a Trellix IPS custom attack with the relevant protocol is automatically created. Also, the corresponding signature is created for the attack. You can add more conditions to the signature or add more signatures to the Trellix IPS custom attack. For example, when you use the template to detect TCP connection attempts from specific IP addresses, the signature for the IP addresses that you specify is automatically created. To specify more IP addresses for this attack, create the corresponding signatures for those IP addresses.

#### Important

By default, the signatures that are created when you use the templates have high value of Benign Trigger Probability (BTP). You can edit this value post-creation. Note that the attacks of BTP value high are not published in the Default Detection and Default Prevention policies.

#### Note

When you save the Trellix IPS custom attacks in the Manager database, an informational fault is displayed in the **Status** page to indicate whether the custom attacks were successfully saved.

### Create a Trellix IPS custom attack to detect a URL

You can use the pre-defined template to create a Trellix IPS custom attack to detect a URL.

#### Task

- 1 Select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click **Custom Attacks**.  
The Custom Attack Editor opens with the existing Custom Attacks listed on the **Native Trellix IPS Format** tab.
- 2 Click .  
The **New Custom Attack** interface opens.
- 3 In the **Name** field, type a new name for your attack. **UDS** (User-Defined Signature) is appended at the front automatically when you save the attack. For example, if you name the new attack "HTTP Attack XYZ", it appears as "UDS-HTTP Attack XYZ" in the Custom Attack Editor as well as in the attack database when you save the attack.

#### Note

The Trellix IPS ID is provided by the Manager when you save it in the Manager server.

**4** Type a **Description** for your attack.

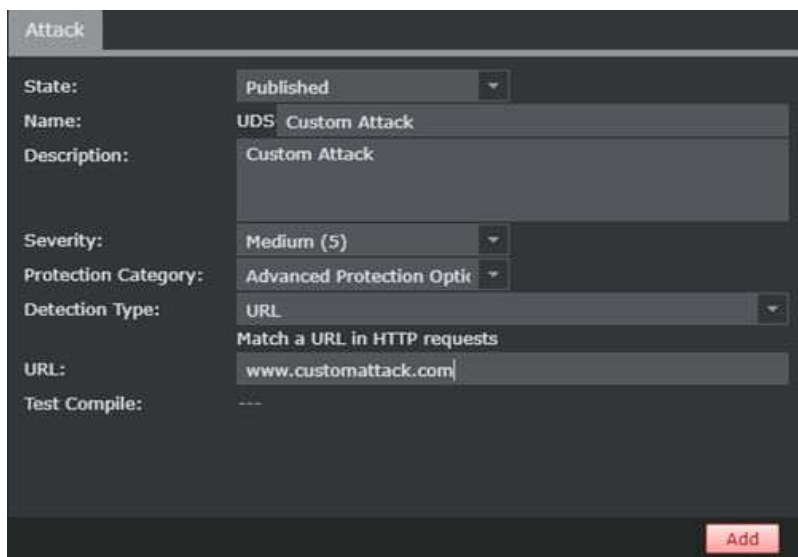
This area can be used for your notes or other specific information pertinent to your new attack.

**5** Select a severity for your attack by toggling the drop-down list. Choices are High (9, 8, 7), Medium (6, 5, 4), Low (3, 2, 1), and Informational (0).

Severity is set to **Medium (5)** by default.

**6** Select the most appropriate **Protection Category** for the attack.**7** Select **URL** from the **Detection Type** drop-down menu for the attack.**8** Enter the **URL** that is to be detected, then click **Add**.

If you are specifying the protocol, you can specify only HTTP or HTTPS in the URL. The attack is listed on the **Native Trellix IPS Format** tab.



The screenshot shows the 'Attack' configuration form in the Trellix IPS interface. The form has a dark theme and includes the following fields and controls:

- State:** A dropdown menu set to 'Published'.
- Name:** A text input field containing 'UDS Custom Attack'.
- Description:** A large text area containing 'Custom Attack'.
- Severity:** A dropdown menu set to 'Medium (5)'.
- Protection Category:** A dropdown menu set to 'Advanced Protection Opti'.
- Detection Type:** A dropdown menu set to 'URL'.
- URL:** A text input field containing 'Match a URL in HTTP requests'.
- Test Compile:** A text input field containing '---'.

An 'Add' button is located at the bottom right of the form.

Figure 4-19 Create a Trellix IPS custom attack to detect a URL

- 9 Double-click on the attack that you created on the **Native Trellix IPS Format** tab.
- In the **Matching Criteria** section, the protocol is automatically selected as HTTP.
- You can also add more conditions to the signature. From the **Add Custom Attack** window, you can also add more signatures to the attack.

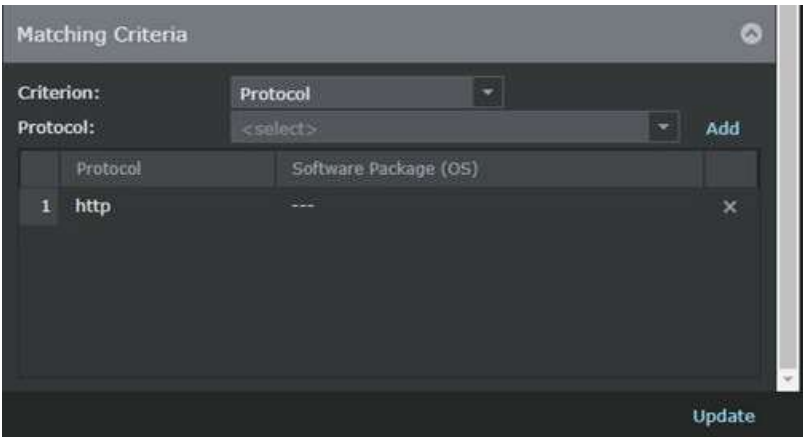


Figure 4-20 Details of the Trellix IPS custom attack

- 10 Click **Save** in the **Custom Attack** interface.
- Until you save the attack in the database, value for the **Trellix IPS ID** column is not generated. The Sensor detects this attack after you deploy pending changes to the corresponding Sensor (**Devices** | **<Admin Domain Name>** | **Global** | **Device Manager**. Select the **Sensors** tab. Then, select the required Sensor from the list and click **Sync**).

Create a Trellix IPS custom attack to detect an email attachment by file name

You can use the predefined template to create a Trellix IPS custom attack to detect an email attachment by the file name.

Task

- 1 Select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click **Custom Attacks**.
- The Custom Attack Editor opens with the existing Custom Attacks listed on the **Native Trellix IPS Format** tab.
- 2 Click **+**.
- The **New Custom Attack** interface opens.
- 3 In the **Name** field, type a new name for your attack. **UDS** (User-Defined Signature) is appended at the front automatically when you save the attack. For example, if you name the new attack "HTTP Attack XYZ", it appears as "UDS-HTTP Attack XYZ" in the Custom Attack Editor as well as in the attack database when you save the attack.

 **Note**

The Trellix IPS ID is provided by the Manager when you save it in the Manager server.

- 4 Type a **Description** for your attack.
- This area can be used for your notes or other specific information pertinent to your new attack.
- 5 Select a severity for your attack by toggling the drop-down list. Choices are High (9, 8, 7), Medium (6, 5, 4), Low (3, 2, 1), and Informational (0).
- Severity is set to **Medium (5)** by default.
- 6 Select the most appropriate **Protection Category** for the attack.

- 7 Select **E-mail Attachment** from the **Detection Type** drop-down menu for the attack.
- 8 Select the required parameter for **Attachment File Name** and enter the corresponding value in the text box.

Attack

State: Published

Name: UDS - Custom Attack

Description: - Custom Attack

Severity: Medium (5)

Protection Category: Advanced Protection Optic

Detection Type: E-mail Attachment

Attachment File Name: Equals Custom\_Attack.zip

Test Compile: ---

Add

Figure 4-21 Custom attack to detect an email attachment by its file name

- 9 Click **Add**.
- 10 Double-click on the attack that you created on the **Native Trellix IPS Format** tab.  
In the **Matching Criteria** section, the protocol is automatically selected as SMTP.  
To edit these values, select the signature on the **Signature-<signature name>** tab. You can modify the default values. You can also add more conditions to the signature.

Matching Criteria

Criterion: Protocol

Protocol: <select> Add

|   | Protocol | Software Package (OS) |
|---|----------|-----------------------|
| 1 | smtp     | ---                   |

Update

Figure 4-22 Details of the Trellix IPS custom attack

- 11 Click **Save** in the **Custom Attack** interface.  
Until you save the attack in the database, value for the **Trellix IPS ID** column is not generated. The Sensor detects this attack after you deploy pending changes to the corresponding Sensor (**Devices | <Admin Domain Name> | Global | Device Manager**). Select the **Sensors** tab. Then, select the required Sensor from list and click **Sync**).

### Create a Trellix IPS custom attack to detect a DNS query or response

You can use the predefined template to create a Trellix IPS custom attack to detect a DNS query or response related to a domain name.

#### Task

- 1 Select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click **Custom Attacks**.

The Custom Attack Editor opens with the existing Custom Attacks listed on the **Native Trellix IPS Format** tab.

- 2 Click .

The **New Custom Attack** interface opens.

- 3 In the **Name** field, type a new name for your attack. **UDS** (User-Defined Signature) is appended at the front automatically when you save the attack. For example, if you name the new attack "HTTP Attack XYZ", it appears as "UDS-HTTP Attack XYZ" in the Custom Attack Editor as well as in the attack database when you save the attack.



#### Note

The Trellix IPS ID is provided by the Manager when you save it in the Manager server.

- 4 Type a **Description** for your attack.  
This area can be used for your notes or other specific information pertinent to your new attack.
- 5 Select a severity for your attack by toggling the drop-down list. Choices are High (9, 8, 7), Medium (6, 5, 4), Low (3, 2, 1), and Informational (0).  
Severity is set to **Medium (5)** by default.
- 6 Select the most appropriate **Protection Category** for the attack.
- 7 Select **DNS Query or Response** from the **Detection Type** drop-down menu for the attack.

- 8 Enter the full or the partial **Domain Name** that the Sensor should detect in a DNS query or response, and then click **Add**.

The screenshot shows a dark-themed 'Attack' configuration window. It contains several labeled input fields: 'State' with a dropdown set to 'Published'; 'Name' with the text 'UDS Custom Attack DNS'; 'Description' with the text 'Custom Attack DNS'; 'Severity' with a dropdown set to 'Medium (5)'; 'Protection Category' with a dropdown set to 'Advanced Protection Optic'; 'Detection Type' with a dropdown set to 'DNS Query or Response'; and 'Domain Name' with the text 'www.custom'. Below these fields is a 'Test Compile' section with a '---' placeholder. An 'Add' button is located at the bottom right of the form.

Figure 4-23 Custom attack to detect an DNS query or response

- 9 Double-click on the attack that you created on the **Native Trellix IPS Format** tab.
- In the **Matching Criteria** section, the protocol is automatically selected as DNS.
- The signature for the URL that you specified is created automatically with the default values. You can modify the default values. You can also add more conditions to the signature.

The screenshot shows a 'Matching Criteria' dialog box. It has a 'Criterion:' dropdown set to 'Protocol'. Below it is a 'Protocol:' dropdown set to '<select>' with an 'Add' button to its right. A table is displayed with two columns: 'Protocol' and 'Software Package (OS)'. The first row of the table has the value '1' in the first column, 'dns' in the second, and '---' in the third. There is an 'x' icon to the right of the first row. An 'Update' button is located at the bottom right of the dialog.

Figure 4-24 Details of the Trellix IPS custom attack

- 10 Click **Save** in the **Custom Attack** interface.
- Until you save the attack in the database, value for the **Trellix IPS ID** column is not generated. The Sensor detects this attack after you deploy pending changes to the corresponding Sensor (**Devices** | **<Admin Domain Name>** | **Global** | **Device Manager**). Select the **Sensors** tab. Then, select the required Sensor from list and click **Sync**).

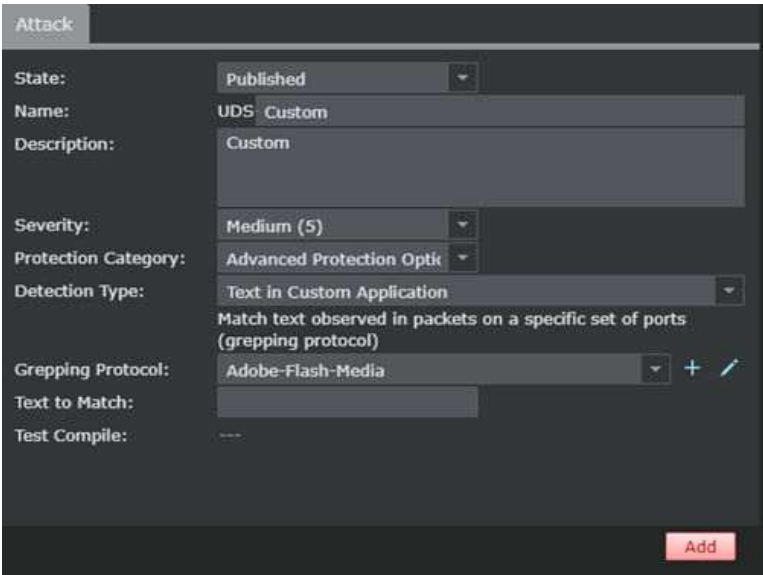
Create a Trellix IPS custom attack to detect a string in an application running over a custom port

You can use the predefined template to create a Trellix IPS custom attack to detect a string in an application that is running over a custom port.

- Task
- 1 Select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click **Custom Attacks**.  
The Custom Attack Editor opens with the existing Custom Attacks listed on the **Native Trellix IPS Format** tab.
  - 2 Click **+**.  
The **New Custom Attack** interface opens.
  - 3 In the **Name** field, type a new name for your attack. **UDS** (User-Defined Signature) is appended at the front automatically when you save the attack. For example, if you name the new attack "HTTP Attack XYZ", it appears as "UDS-HTTP Attack XYZ" in the Custom Attack Editor as well as in the attack database when you save the attack.

 **Note**  
The Trellix IPS ID is provided by the Manager when you save it in the Manager server.

- 4 Type a **Description** for your attack.  
This area can be used for your notes or other specific information pertinent to your new attack.
- 5 Select a severity for your attack by toggling the drop-down list. Choices are High (9, 8, 7), Medium (6, 5, 4), Low (3, 2, 1), and Informational (0).  
Severity is set to **Medium (5)** by default.
- 6 Select the most appropriate **Protection Category** for the attack.
- 7 Select **Text in Custom Application** from the **Detection Type** drop-down menu for the attack.
- 8 Select the appropriate **Grepping Protocol**.
- 9 Enter the string that you want the Sensor to detect in the **Text to Match** field and click **Add**.  
The **Text to Match** field is case-sensitive. That is, the Sensor matches the case when it detects the string.



The screenshot shows the 'Attack' editor interface. It includes a 'State' dropdown set to 'Published'. The 'Name' field contains 'UDS Custom' and the 'Description' field contains 'Custom'. The 'Severity' dropdown is set to 'Medium (5)'. The 'Protection Category' dropdown is set to 'Advanced Protection Optik'. The 'Detection Type' dropdown is set to 'Text in Custom Application', with a sub-label 'Match text observed in packets on a specific set of ports (grepping protocol)'. The 'Grepping Protocol' dropdown is set to 'Adobe-Flash-Media'. The 'Text to Match' field is empty. The 'Test Compile' field shows '---'. An 'Add' button is at the bottom right.

Figure 4-25 Custom attack to detect a string in an application on a custom port



- 10 Double-click on the attack that you created on the **Native Trellix IPS Format** tab.

You can modify the values of the signature on the **Signature-<signature name>** tab. You can also add more conditions to the signature.

- 11 Click **Save** in the **Custom Attack** interface.

Until you save the attack in the database, value for the **Trellix IPS ID** is not generated. The Sensor detects this attack after you deploy pending changes to the corresponding Sensor (**Devices | <Admin Domain Name> | Global | Device Manager**). Select the **Sensors** tab. Then, select the required Sensor from list and click **Sync**.

### Create a Trellix IPS custom attack to detect TCP connection attempts

You can use the predefined template to create a Trellix IPS custom attack to detect TCP connection attempts from specific IP addresses.

#### Task

- 1 Select **Policy | <Admin Domain Name> | Intrusion Prevention | Policy Types | IPS**. Click **Custom Attacks**.

The Custom Attack Editor opens with the existing Custom Attacks listed on the **Native Trellix IPS Format** tab.

- 2 Click .

The **New Custom Attack** interface opens.

- 3 In the **Name** field, type a new name for your attack. **UDS** (User-Defined Signature) is appended at the front automatically when you save the attack. For example, if you name the new attack "HTTP Attack XYZ", it appears as "UDS-HTTP Attack XYZ" in the Custom Attack Editor as well as in the attack database when you save the attack.



#### Note

The Trellix IPS ID is provided by the Manager when you save it in the Manager server.

- 4 Type a **Description** for your attack.

This area can be used for your notes or other specific information pertinent to your new attack.

- 5 Select a severity for your attack by toggling the drop-down list. Choices are High (9, 8, 7), Medium (6, 5, 4), Low (3, 2, 1), and Informational (0).

Severity is set to **Medium (5)** by default.

- 6 Select the most appropriate **Protection Category** for the attack.

- 7 Select **Connection from Specific IP** from the **Detection Type** drop-down menu for the attack.

- 8 Enter the IPv4 or IPv6 **Source IP Address** from which TCP connection attempts are to be detected and click **Add**. The attack is listed on the **Native Trellix IPS Format** tab.

Attack

State: Published

Name: UDS Custom

Description: Custom

Severity: Medium (5)

Protection Category: Advanced Protection Optic

Detection Type: Connection from Specific IP

Match a TCP connection attempt from a specific IPv4 and IPv6 address.

**Warning:** Only 10 fixed field tests (signatures matching an IP address) are supported per device

Source IP Address: 192.168.1.5

Test Compile: ---

Add

- 9 Double-click on the attack that you created on the **Native Trellix IPS Format** tab. In the **Matching Criteria** section, the **Software Package (OS)** is selected as **tcipip-machine**.

Matching Criteria

Criterion: Protocol

Protocol: <select> Add

|   | Protocol | Software Package (OS) |   |
|---|----------|-----------------------|---|
| 1 | ---      | tcipip-machine        | X |

Update

You can modify the values of the signature on the **Signature-<signature name>** tab. You can also add more conditions to the signature. For example, do the following to add conditions based on TCP flags:

- 1 Add a signature or edit a signature.
- 2 In the corresponding signature, click **+**.
- 3 Add a condition by clicking **AND** or **OR**.
- 4 From **Comparison Type** list, select **Single Fixed Field Match**.
- 5 Select **tcp** from the **Protocol** drop-down menu.

- 6 Select the required flag from the **Protocol Field**.
  - 7 Select the **Operator**.
  - 8 Enter the value to match in the **Integer or IP to Match** field.
  - 9 Select **Ignore Bitmask** if you want to ignore the bitmask.
  - 10 Click **Save**.
  - 10 Click **Save** in the **Custom Attack** interface.
- Until you save the attack in the database, value for the **Trellix IPS ID** column is not generated. The Sensor detects this attack after you deploy pending changes to the corresponding Sensor (**Devices | <Admin Domain Name> | Global | Device Manager**. Select the **Sensors** tab. Then, select the required Sensor from list and click **Sync**).

## Create an exploit attack without template

This section explains how to create Trellix IPS Exploit Attacks and the constituent signatures.

To create a Trellix IPS Exploit Attack instance, you start by first adding a new attack. Attacks are configured first because signatures logically relate to attacks as one particular means of detection.

Attack creation includes impact categorization for proper policy integration and enforcement. Each policy — those provided with Trellix IPS and those you create — consists of rule sets, which contain multiple categories of attacks. Attacks are categorized by the protocols, operating systems, and applications they impact. Within the Custom Attack Editor, you define the impact categories wherein your attack definition best fits, so that when you save it in the Manager, it will be published in one or more rule sets.

### Task

- 1 Select **Policy | <Admin Domain Name> | Intrusion Prevention | Policy Types | IPS**. Click **Custom Attacks**.  
The Custom Attack Editor opens with the existing Custom Attacks listed on the **Native Trellix IPS Format** tab.
- 2 Click **+**.

The New Custom Attack interface opens.

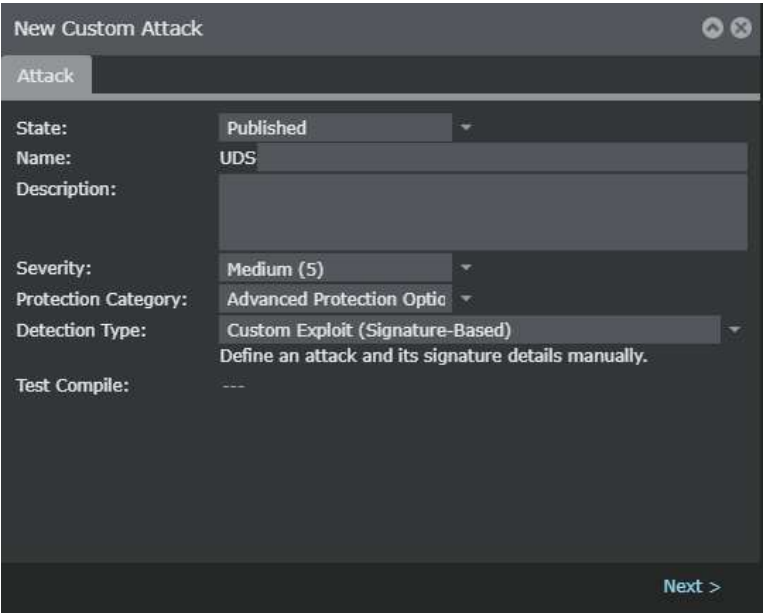


Figure 4-26 Add Exploit Attack window

- 3** In the **Name** field, type a new name for your attack. **UDS** (User-Defined Signature) is appended at the front automatically when you save the attack. For example, if you name the new attack "HTTP Attack XYZ", it appears as "UDS-HTTP Attack XYZ" in the Custom Attack Editor as well as in the attack database when you save the attack.



### Note

The Trellix IPS ID is provided by the Manager when you save it in the Manager server.

- 4** Type a **Description** for your attack.  
This area can be used for your notes or other specific information pertinent to your new attack.
- 5** Select a severity for your attack by toggling the drop-down list. Choices are High (9, 8, 7), Medium (6, 5, 4), Low (3, 2, 1), and Informational (0).
- 6** Select the most appropriate **Protection Category** for the attack.
- 7** Select **Custom Exploit (Signature-Based)** from the **Detection Type** drop-down menu for the attack.
- 8** Click **Next**.
- 9** Select an appropriate **Attack Target**.
- 10** Select an appropriate **Blocking** type.
- 11** On the **Matching Criteria** tab:
- a** Complete at least one of the following:
    - 1** Categorize the attack definition by protocol. For example: HTTP, FTP, DNS. In the **Criterion** dialog box, select **Protocol** and then select a protocol from the **Protocol** list.
    - 2** Categorize by software package (that is, application and operating system). For example: Internet Explorer on Windows 2003. In the **Criterion** dialog, select **Software Package (OS)** and then select one package from the list. Optionally, select an **Operating System**.
  - b** Click **Add**.
  - c** Repeat steps to specify more protocols or packages.



### Note

It is mandatory that you specify at least one impact package or protocol. If you are not sure about the impact package or protocol, select tcpip-machine as the impact package.



### Tip

Check the rule sets (Inbound and Outbound) of the policy you plan to enforce to determine whether your attack will be selected. You also have the option of creating a new rule set in order to enforce your Custom Attack.

- 12** Once your attack configuration is complete, continue to Creating a Signature.

## Tasks

- [Create a signature on page 1559](#)

## Create a signature

Create one or more signatures to detect an attack. This section assumes that you are adding signatures to the attack that you just created in the previous section. That is, the Add Trellix IPS Attack interface is open and the attack you created is yet to be saved in the Manager server. If you are adding signatures to a previously saved attack, double-click on the attack in a tabbed region to open the Edit Trellix IPS Attack interface.

### Note

You can create up to 15 signatures per attack definition.

### Task

- 1 In the **New Custom Attack** interface, click on the **Signature-<signature name>** tab.  
The signature tab appears.
- 2 Optionally clear the **Signature Name** and type a new one.
- 3 Select a **Benign Trigger Probability** value.

UDS-new\_custom\_attack

Attack Signature-1663309908396 +

Name: Signature-1663309908396

Benign Trigger Probability (BTP): Medium (4)

Target Host Architecture: Any

Detection Window: Request Packets

Supported Device Types: Any

Signature Details

Use the buttons below to add signature details. Each signature requires:

1. At least one condition.
2. At least one AND or OR comparison within each condition.

Condition 1

http-req-header == "www.test.com" (casesensitive=false)

Update

Figure 4-27 New Signature window

- 4 Select a **Target Host Architecture**. The default is **Any**.
- 5 Select a **Detection Window**. The choices are as follows:
  - **Single Packet**
  - **Request Packets**
  - **Response Packets**
  - **Entire Flow**

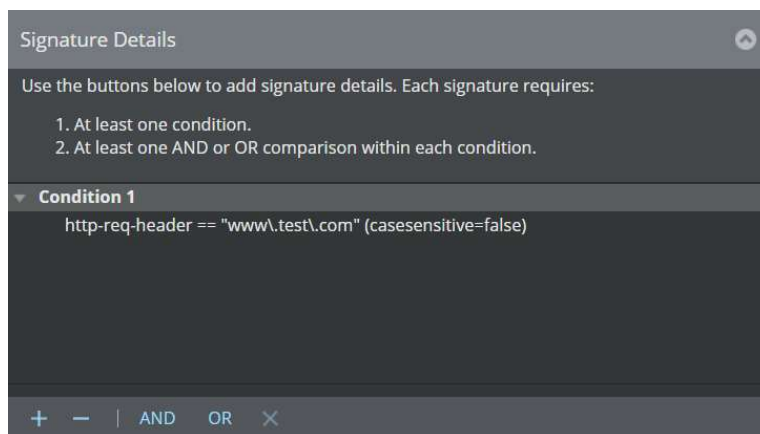
6 Based on the Sensor model that you plan to use you can select any of the following options:

- NS-series Only
- VM-series Only
- Any

7 Under **Signature Details** panel, **Condition 1** is added and selected by default.

To add another condition, do the following:

- a Click **+** under **Signature Details**. **[AND THEN] Condition 2** appears in the blank white field.
- b Click **Condition 1** or **[AND THEN] Condition 2** so that it is highlighted.



c Do one of the following:

- 1 Click **AND** under Comparisons to add an AND comparison.
- 2 Click **OR** under Comparison to add OR comparisons.

d Select a comparison type from the **Comparison Type** list. The choices are as follows:

- **String Pattern Match:** The pattern contains a sequence of characters which will be used for detections. It can contain alphanumeric and hexadecimal sequence.
- **Numeric Value Match:** The pattern contains only numeric values.
- **Numeric Range Match:** The pattern contains only numeric values. You can configure a minimum and maximum value. These are used to detect numeric values which falls between minimum and maximum value.
- **Numeric Enumeration Match:** The pattern contains only numeric values. The values may not be in any order but a sequence of numbers against which detection occurs.
- **Single Fixed Field Match:** This contains mixed values. The input type varies based on the protocol and protocol field selected.
- **Packet Grep Protocol Match:** This contains alphanumeric or hexadecimal as input and does not contain any regular expression. This will detect patterns in the selected protocol for either request or response. Select this option, only if the protocols are not listed in the **String Pattern Match** or if the requested **Protocol** field is missing from the **String Pattern Match** list.

Your selection affects what appears in the subsequent dialogs. The **String Pattern Match** option requires knowledge of the Regular Expression Language.

- e From the **Protocol Details** box, select a protocol from the **Protocol** list.  
The subsequent options are based on the selected protocol.

 **Note**

If you selected **Packet Grep Protocol** from the **Comparison Type** drop-down menu, the available protocols are those packet grep instances created/provided within the **Protocol** drop-down menu.

- f Configure the fields for the comparison(s) you have chosen.

 **Note**

If selection fields (drop-down menus) are blank, then the protocol and comparison you have chosen cannot be used to create a signature condition; select another protocol and/or comparison combination.

For example, if you want to configure a string match in the URI path of an HTTP GET request:

- 1 Select **http** for the **Protocol**, **req-uri-path** for the **Protocol Field** and select **get** as the **http-req-method**.

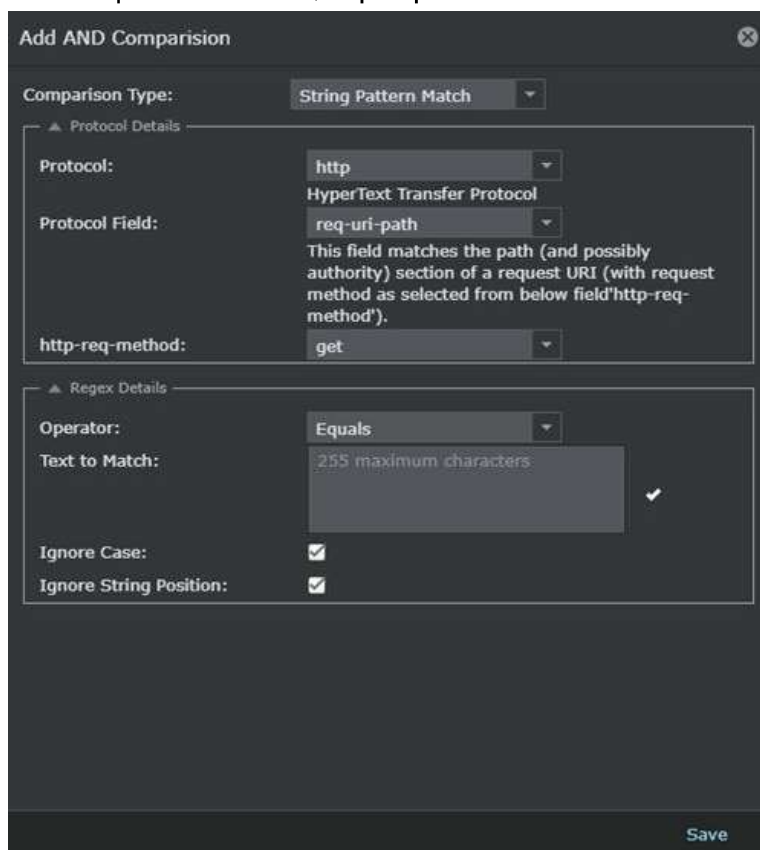


Figure 4-28 Configure Comparison window

- 2 From the **Operator** drop-down list in the **Regex Details** section, select the matching criteria as **Equals**.
- 3 If you are configuring a string pattern match, type your pattern in **Text to Match**.
- 4 Optionally, click the **Ignore Case** check box if you want the pattern to be matched regardless of [letter] case.

- 5 Optionally, click the **Ignore String Position** check box if you want the pattern to be matched regardless of string position.
  - 6 Click  to verify that your pattern is valid.
  - g Click **Save** when done with the Configure Comparison fields. Your comparison appears under **Condition 1**.
  - h Do *one* of the following:
    - Click  to add another condition.
    - Select **Condition 1** and click **AND** under Comparisons to add another AND comparison.
    - Select **Condition 1** and click **OR** under Comparisons to add multiple OR comparisons to your condition. Note that the first of the OR comparison appears under **[AND](One Of)**.
    - (Optional) Do *one* of the following if you have created multiple conditions:
      - Select a condition and click  under to delete a condition, or select a comparison and click  to delete a comparison.
  - 8 Optionally, you can add more signatures to your attack by clicking  next to the signature tab.
  - 9 Click **Save** in the Custom Attack interface.
- The created Attack and its signatures are saved in the Manager client that you are logged on to.

### **Caution**

If you close the Custom Attack Editor without saving the attack, the attack that you created (along with any unsaved changes to other attacks) is lost.

See also

*Signature creation interface* on page 1519

*Signature test reference* on page 1544

*Regular expression language* on page 1566

*Create a packet search protocol* on page 1527

*Viewing a policy to verify inclusion of the attack* on page 1604

## Configure custom reconnaissance attack definition

In the **Custom Attack Editor** (formerly UDS Editor), you create custom recon attacks using both user-defined exploit attacks and Trellix IPS-defined exploit attacks (in sigset) as component attack. You can also define correlated attacks using these individual attack definitions. For example, UDS attacks that check for URI can be further correlated to test for multiple occurrences in a defined time interval to raise a correlated alert.

To configure Custom Reconnaissance Attack Definition:

### Task

- 1 Select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click **Custom Attacks**.  
The Custom Attack Editor opens with the existing Custom Attacks listed on the **Native Trellix IPS Format** tab.
  - 2 Click .
- The New Custom Attack interface opens.



- 3 In the **Name** field, type the new custom reconnaissance attack.

**Note**

The Trellix IPS ID is provided by the Manager when you save it in the Manager server.

- 4 Optionally, type a **Description** for your attack.  
This area can be used for your notes or other specific information pertinent to your new attack.
- 5 Select a severity for your attack by toggling the drop-down list. Choices are High (9, 8, 7), Medium (6, 5, 4), Low (3, 2, 1), and Informational (0).
- 6 Select the most appropriate **Protection Category** for the attack.
- 7 Select **Custom Reconnaissance Attack (Correlation-Based)** from the **Detection Type** drop-down menu for the attack.

Figure 4-29 New Custom Attack window

- 8 Click Next.
- 9 Set the **Benign Trigger Probability (BTP)** for the reconnaissance attack.
- 10 Set the **Attack Subcategory** from **Correlation Logic** for the attack.

The following subcategories are available:

- brute-force
- fingerprinting
- host-sweep
- port-scan
- service-sweep

You can select the above options using a single component attack for correlation. Select **fingerprinting** option if you wish to use multiple component attacks for correlation.

- 11 In the **Attack** field, type the attack name and select the relevant attack from the list.  
You can select either a Trellix IPS-defined exploit attack or a user-defined custom attack as a component attack.
- 12 Set the **Threshold** value between 1 and 255.

- 13 Set the **Interval** in seconds between 1 and 65535.
- 14 Select the **Generate Individual Component Alerts** check box.
- 15 Select either **Protocol** or **Software Package (OS)**, and click **Add**.  
You can add more than one matching criteria to your custom reconnaissance attack.

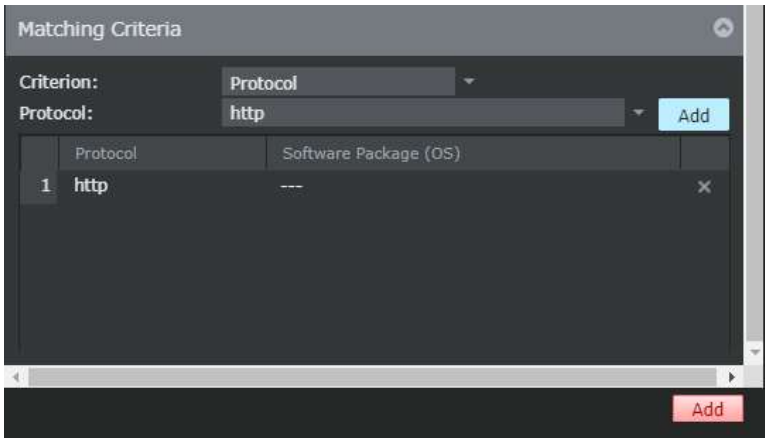


Figure 4-30 Add Protocols Packages page

- 16 Click **Add**.
- 17 Review all the settings, and click **Save**.  
When traffic passing through the Sensor exceeds the threshold count set for the custom reconnaissance attack within a configured **Interval**, the Sensor raises an alert to the Manager. You can view the alert in the **Attack Log** page. The alerts **Summary** shows the type of attack as either a Trellix IPS Exploit or Trellix IPS Reconnaissance.



**Note**

The custom reconnaissance attacks are staged by the IPS engine when:

- One or more of its component attacks is deleted.
- One or more of its component attacks is marked as **Staged**.

**Tasks**

- *Configure custom attack definition for multiple attacks correlation* on page 1564

**Configure custom attack definition for multiple attacks correlation**

To create custom attack definition for multiple attack correlations perform the following steps:

**Task**

- 1 Select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click **Custom Attacks**.  
The Custom Attack Editor opens with the existing Custom Attacks listed on the **Native Trellix IPS Format** tab.
- 2 Click **+**.  
The New Custom Attack interface opens.

- 3 In the **Name** field, type the new custom reconnaissance attack.

**Note**

The Trellix IPS ID is provided by the Manager when you save it in the Manager server.

- 4 Optionally type a **Description** for your attack.  
This area can be used for your notes or other specific information pertinent to your new attack.
- 5 Select a severity for your attack by toggling the drop-down list. Choices are High (9, 8, 7), Medium (6, 5, 4), Low (3, 2, 1), and Informational (0).
- 6 Select the most appropriate **Protection Category** for the attack.
- 7 Select **Custom Reconnaissance Attack (Correlation-Based)** from the **Detection Type** drop-down menu for the attack.
- 8 Click **Next**.
- 9 If you wish to create a custom reconnaissance attack using multiple attacks correlation, select the **Correlation Method** as **fingerprinting**.
- 10 Use the **Add** button to add multiple attacks as the component attacks for the custom attack that you are creating.  
You must select a minimum of 2 component attacks for multiple attacks correlation.

**Note**

You can include a component attack in 10 custom reconnaissance attacks.

Figure 4-31 Custom Attack Definition for Multiple Attacks Correlation

- 11 Set the **Threshold** value between 1 and 255.
- 12 Set the **Interval** in seconds between 1 and 65535.
- 13 Select the **Generate Individual Component Alerts** check box.
- 14 Click **Add**.
- 15 Review all the settings, and click **Save** in the **Custom Attacks** window.

## Regular expression language

This chapter describes the proper method of writing pattern-matching signatures using the Regular Expression Language supported for Trellix IPS Custom Attacks.

 **Note**

Signature creation requires a working knowledge of protocols and the data they carry across your network, as well as an ability to write a proper signature using a regular expression language. This information is intended for advanced network security personnel who have experience with Trellix IPS Custom Attack creation.

Signature creation is based on a set of expressions in a defined syntax, also known as regular expressions.  
See also  
[Create a signature on page 1559](#)

### Regular expression syntax

All expressions are generated from the following set of rules for pattern representation, and only from these rules:

#### Printable characters

You can use any alphanumeric characters for basic string matching purposes.

- lower case 'a' through 'z'
- upper case 'A' through 'Z'
- numerals '0' though '9'

Thus, typing "confidential" and marking the test as Case-Insensitive denotes that either "Confidential" or "confidential" will match. Likewise, the string "007forever" marked as Case-Sensitive denotes a search for the exact phrase "007forever".

#### Special characters

Special characters are those that are non-alphanumeric/punctuation. They require a special representation so that they are visible and not translated as formatting. A "\r" is a carriage return, part of what makes your cursor go to the next line.

| Action or Symbol           | ASCII    | Type as: |
|----------------------------|----------|----------|
| carriage return            | CR       | \r       |
| newline                    | NL or LF | \n       |
| horizontal tab             | HT       | \t       |
| vertical tab               | VT       | \v       |
| formfeed                   | FF       | \f       |
| backspace                  | BS       | \b       |
| bell char                  | BS ??    | \a       |
| match any single character |          | .        |

| Pattern                 | Read as:               |
|-------------------------|------------------------|
| first line\nsecond line | first line second line |

## Hexadecimal values

Hexadecimal values can be represented as two alphanumeric characters (a to f, and A to F, and 0 to 9) preceded by "x"; thus "\x00" is a proper hex string. The range is from \x00 to \xFF, and matching must be case-insensitive.

| Pattern      | Description                                                               |
|--------------|---------------------------------------------------------------------------|
| \x90\x90\x90 | Example of hex used to represent binary; often used in shellcode attacks. |

## Character class

To search for a range of characters, define the range using a hyphen between characters [A - Z].

| Pattern | Description                                       |
|---------|---------------------------------------------------|
| [1-9]   | Any digit from 1 to 9 (1, 2, 3, 4, 5, 6, 7, 8, 9) |
| [a-j]   | Any lowercase letter from a to j.                 |
| [A-H]   | Any capitalized letter from A to H.               |

 **Note:** A search such as [a-F] is not valid.

## Optional

To make a character or pattern optional, use parentheses with a trailing question mark: (*string*)?.

| Pattern           | Description                                                                                                             |
|-------------------|-------------------------------------------------------------------------------------------------------------------------|
| code red (virus)? | The search for the word "virus" is optional. This pattern matches if either "code red" or "code red virus" is detected. |

## Alternation

To search using alternation, use the pipe (|) character to separate strings (a|b).

| Pattern     | Description                            |
|-------------|----------------------------------------|
| (hi hello)  | Either "hi" or "hello" must match.     |
| (su soo)per | Either "super" or "sooper" must match. |

## Repeating

To repeat, append a number in brackets to the pattern: for example, (*string*){*n*} means the string must be repeated *n* times for matching purposes.

### Note

You must enclose the string in parentheses, and the repeating value must be enclosed in brackets.

### Note

The number in braces must be greater than 0 (*n*>0).

| Pattern                | Description                                                                          |
|------------------------|--------------------------------------------------------------------------------------|
| (1){5}=11111           | A string of five consecutive 1s without spaces must match.                           |
| (text){3}=texttexttext | A string consisting of the word text repeated three times without spaces must match. |

To set a minimum and maximum, represent the equation as (string){3,5}

| Pattern                       | Description                                                                          |
|-------------------------------|--------------------------------------------------------------------------------------|
| (1){3,5}=111 or 1111 or 11111 | A string of three, four, <b>OR</b> five consecutive 1s without spaces must be match. |

Anchors

To specify a string that starts or ends a line, use the ^ (carat) or \$ (dollar) characters. Using both ^ and \$, you can also specify a string that makes up an entire line—start to end.

- Start of line anchor: ^(string)
- End of line anchor: (string)\$

| Pattern          | Description                            |
|------------------|----------------------------------------|
| ^Hi              | The first word of a line must be "Hi". |
| bye\$            | The last word of a line must be "bye". |
| ^Hacked by J03\$ | "Hacked by J03" forms an entire line.  |

Reserved characters

The following characters are reserved for regular expression control. They *must* be escaped; that is, prefixed with backslash (\) when they appear as part of a normal matching pattern (that is, not part of an expression), except where noted otherwise.

| Name          | Symbol | Proper Syntax |
|---------------|--------|---------------|
| backslash     | \      | \\            |
| dot (period)  | .      | \.            |
| pipe          |        | \             |
| dash (hyphen) | -      | \-            |
| carat         | ^      | \^            |
| dollar symbol | \$     | \\$           |
| open paren    | (      | \(            |
| close paren   | )      | \)            |
| open bracket  | [      | \[            |
| close bracket | ]      | \]            |

| Pattern             | Read as:                                      |
|---------------------|-----------------------------------------------|
| (text)              | text                                          |
| \\(text)            | \\(text\\)                                    |
| www.trellix.com     | www*trellix*com, where * is a single wildcard |
| www\\.trellix\\.com | www.trellix.com                               |

## Limitations of a custom reconnaissance attack

The limitations of Custom Reconnaissance attack are listed below:

- The maximum threshold count for Custom Reconnaissance attack is 255 for single-AID (type: port scan, host sweep, service sweep, and brute force).
- A component attack can only be included in 10 Custom Reconnaissance attacks.
- A minimum of 2 component attacks need to be added for a multiple-AID (type: fingerprinting) Custom Reconnaissance attack.
- Maximum 5 component attacks can be added in a multiple-AID (type: fingerprinting) Custom Reconnaissance attack.
- Maximum 300 Reconnaissance attacks (Trellix IPS and Custom) can be pushed to the Sensor.
- Maximum 300 component attacks can be pushed to the Sensor.
- Maximum 150 Custom Reconnaissance attacks can be added to the Custom Attack Editor.

## Mechanics of a Snort custom attack

This section explains the structure of a Snort Custom Attack. It also provides some information on how to construct valid Snort rules.

### Structure of a Snort custom attack

A Snort Custom Attack is made up of one Snort rule. To define a Snort Custom Attack, you either write the Snort rule directly in the Custom Attack Editor or import it into the Editor from a file. The Sensor automatically creates the Snort Custom Attack record for each rule that you write or import.

When the Sensor creates the attack for a Snort rule, it automatically defines the values for fields such as attack name and attack severity. The following section explains how the Sensor defines the attack-level values for a Snort Custom Attack:

For the Attack Name, the format that the Sensor uses for the name is SNORT:<the text specified for the msg rule option in the rule>(<SID>). So, this name is modified accordingly if you modify the msg text or the SID of the rule.

#### Note

msg rule option and a unique SID are mandatory for a Snort Custom Attack.

Blocking is set to attack packet for all Snort Custom Attacks. You cannot modify this.

On the **Impact** tab, the Severity is based on the classtype keyword or the priority tag. Each classtype has a default priority defined. Typically, the classtypes with the default priority values are defined in the classification.config file. If a rule has both classtype keyword and priority tag, the severity is based on the priority tag.

The Sensor assigns the severity for a Snort custom attack based on the following:

- A priority 1 Snort attack definition is assigned a severity of High.
- A priority 2 Snort attack definition is assigned a severity of Medium.
- A priority of 3 or higher is assigned a severity of Low.

When you modify the classtype or the priority tag value of a rule, the Sensor will also modify the severity accordingly. This means when you save the attack to the Sensor server, the rule sets (and policies) in which the attack was published may change.

For the Exploit category, tcpip-machine is selected as the package with any as the OS. You cannot add, delete, or modify any impact package or protocol for Snort Custom Attacks.

### Structure of a snort rule

This section discusses the basic structure of a Snort rule and the Snort rule elements that are supported in Trellix IPS. This section contains information on the syntax that you should use in Trellix IPS for each element. You can also find an example rule for each element, wherever applicable.

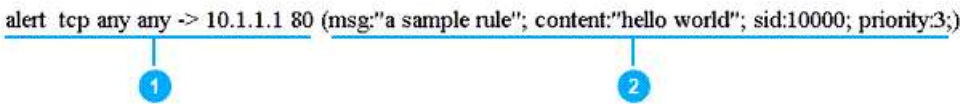
How you combine various Snort rule elements to form a rule depends on your security requirements. This information is out of scope of this document because the requirements can vary from network to network.

 **Note**

The Snort rules that you plan to use in Trellix IPS should conform to Snort rules language.

A Snort rule in Trellix IPS can run into multiple lines. A typical Snort rule has two logical sections — rule header and rule options.

The following is an example of a simple but valid Snort rule.



| Item | Description          |
|------|----------------------|
| 1    | Rule header section  |
| 2    | Rule options section |

As shown above, the rule header section starts at the beginning of the rule and ends at the opening parenthesis. This section contains the following:

- Action — In the sample above, it is alert.
- Protocol — In the sample above, it is TCP.
- Source and destination IP addresses — In this case, the source is any and the destination is 10.1.1.1
- Source and destination ports — In this case, the source port is "any" and the destination port is 80.



The rule options section is enclosed within the parentheses. This section contains alert messages and information on which parts of the packet should be inspected to determine if the rule action should be taken. In the rules options section, the words ending in a colon are option keywords.

### Note

In Trellix IPS, the rule options section should have at least the msg and the sid keywords with values. Rule options enable you accurately define the attack traffic that the Sensor should look for.

All the elements that make up a rule must be true for the Sensor to raise an alert and take the defined response action. You specify the response action for the Snort Custom Attack in the IPS Policy Editor.

### Rule header section

In the rule header section, you define the following:

- Action the Sensor needs to take when it sees the attack traffic
- Transport layer protocol of the attack traffic
- Source and destination IP for the attack traffic
- Communication port at both the source and destination of the attack
- Direction of the traffic that the Sensor should inspect

The items listed above are discussed in detail in the subsequent sections.

### Rule action

The action specifies how the Sensor should respond when it detects traffic as defined in the rule. In Trellix IPS, the only action that you can specify in a Snort rule is to send an alert to the Manager. The procedure to configure other response actions such as packet log and packet drop is the same as that of the regular Trellix IPS attacks. That is, use the IPS Policy Editor to configure these actions.

### Protocols

In the protocols section you specify the protocol to which the Sensor should apply the rule.

The options are:

- |        |       |
|--------|-------|
| • TCP  | • FTP |
| • UDP  | • TLS |
| • ICMP | • SMB |
| • IP   | • DNS |
| • HTTP |       |

For more information, see <http://suricata.readthedocs.io/en/suricata-3.2.3/rules/intro.html#protocol>

### IP addresses

You need to specify the IP address of the source and target for the attack traffic. The available options are listed below:

- You can use the keyword "any" to specify a wildcard IP address.
- You can use CIDR blocks to specify a range of IP addresses. Example: 172.16.230.0/24

- You can specify a list of IP addresses by specifying them inside square brackets with comma as the separator. Example: [172.16.230.0/24, 172.16.231.0/24]
- You can use variables to specify particular IPs or a range of IPs.

### Port numbers

You need to specify the port numbers used at the source and the target for the attack traffic. The available options are listed below:

- You can use the keyword "any" to specify a wildcard port number.
- You can specify a port range using the range operator, colon.  
Example: 1:1024. This indicates that the port must be between 1 and 1024, both inclusive.  
Example: 100: This indicates the port must be equal to 100 or more.  
Example: :500. This indicates that the port must be less than or equal to 500.
- You can use variables to indicate ports.

### Direction operator

The direction operator -> indicates the direction of the traffic that the Sensor should apply the rule to. The IP address and port numbers on the left side of the direction operator are that of the source and the ones on the right side are that of the destination of the attack traffic.

### Rule options section

The rule options section is the more critical section of a Snort rule. The rule options are to be separated by a semicolon. In a rule option, the keyword and argument are separated by a colon.

Rule options fall into one of the following categories:

- **General:** These options contain the metadata for the rule but have no effect on attack detection.
- **Payload:** These options look for data inside the packet payload and can be interrelated.
- **Non-payload:** These options look for non-payload data.
- **Post-detection:** These options indicate what happens when an attack is detected. These options are not relevant in Trellix IPS. Instead, you can configure the response actions in the IPS Policy Editor.

### General rule options

#### msg

In the msg rule option, you indicate the alert message. It is a simple text string that utilizes the \ as an escape character to indicate a discrete character such as a semi-colon.

Syntax:

```
msg: "<message text>";
```

When you create a Snort Custom Attack in the Custom Attack Editor, the Manager assigns the attack and rule name, which you cannot edit directly. The format of the name is as below:

Snort: <text entered for msg option> (sid: <sid value>)

The argument of the msg option features in the attack and rule name that the Manager assigns. Changing the msg argument or sid value and saving the rule will change the attack and rule name accordingly.

### reference

In a rule, you can use the reference keyword to include references related to the corresponding attack traffic. For example, it could be the bugtraq or CVE id of known vulnerabilities. It could also be the address of a website that provides the details of the attack.

The following are the supported systems that you can specify in the reference option:

- bugtraq
- CVE
- Trellix IPS ID
- Arachnids
- URL

Syntax:

```
reference:<id system>, id;
```

Examples:

```
reference: bugtraq, 514;
```

```
reference: url, www.microsoft.com/technet/security/bulletin/MS99-034.msp;
```

You can view the references that are currently available in the Manager.

### gid

gid stands for generator id and is not relevant in Trellix IPS. So, do not use them in your rules.

### sid

sid stands for Snort rule ID. You must specify a unique sid for each Snort rule in the Manager. For Snort Custom Attacks that failed to import, the Manager assigns -1 as the sid.

Syntax:

```
sid:<sid value>;
```

### rev

This enables you to identify any revisions of an existing rule. It is not a mandatory option, but Snort Custom Attacks without the rev option are converted with a warning. When you edit a Snort Custom Attack in the Manager, you can modify the rev value for your reference.

If you want to re-import a Snort Custom Attack with the same sid, then you can change the rev value and import it. The Manager overwrites the existing Snort Custom Attack with the one that you last imported.

Syntax:

```
rev:<revision integer>;
```

### classtype

This is one of the options that you can use to categorize a Snort rule. In the Snort rules provided by Snort.org, the classtypes with the default priority are defined in the `classification.config` file. You can use these classtypes by importing the `classification.config` file into the Manager. Alternatively, you can define your own in a `.config` file and import it into the Manager. If you are defining your own classtype, then make sure you also assign a priority to the classtype.



#### Note

A rule must have a classtype or a priority.



#### Note

You can view the classification types that are currently available in the Manager.

Syntax for using a classtype in a rule is as follows. This assumes that the class name is available in the Manager database:

```
classtype: <class name>;
```

Syntax (for defining a classtype in a `.conf` file):

```
config classification: <class name>,<class description>,<default priority>
```

Example:

```
config classification: attempted-user,Attempted User Privilege Gain,1
```

### priority

Using the priority option you can assign a severity to the rule. This overrides the default severity defined in the classtype.

The Manager assigns the severity for a Snort custom attack based on the following:

- A priority 1 Snort attack definition is assigned a severity of High.
- A priority 2 Snort attack definition is assigned a severity of Medium.
- A priority of 3 or higher is assigned a severity of Low.

Syntax:

```
priority: <integer value>;
```

### Payload rule options

This section describes the options and conditions that you can apply to the payload in TCP, UDP, ICMP, IP, HTTP, FTP, TLS, SMB, or DNS traffic. Note that some of the options work differently in Trellix IPS when compared to Snort. Also, some options are not supported.

The payload rule options discussed in this section include content modifier and preprocessor rule options. You use content modifiers to change the way content option works. These are specifications that the Sensor should apply on the content data before it looks for a match.

### Content

This enables you to specify the content that a Sensor should look for in the packet payload. If the Sensor finds traffic matching what you specify in the content, then it raises an alert and also takes the defined response action such as packet log or packet drop.

The option data for the content keyword can contain text, binary (hex values), or mixed text and binary data. The binary data must be enclosed within the pipe (|) character and represented as bytecode.

### Note

- The following characters must be escaped inside a content rule:
  - : (colon)
  - ; (semi colon)
  - \ (backward slash)
  - " (double quotes)
- Though you can specify even a single-byte token, the longer the content value, the accurate the detection will be. The maximum length of a content or uricontent for NS-series Sensors is 256 bytes.
- Make sure the content option does not contain any generic values identified by Trellix IPS (some examples are listed below). Such rules can severely impact Sensor performance.
  - GET
  - POST
  - Host:
  - User-Agent

Syntax: `content:"<content string>"`;

Example rules:

- `alert tcp any any -> 10.1.1.1 80 (msg:"command dot exe attempt"; content:"cmd.exe"; flow:to_server; sid:2001; priority: 1;)`
- `alert tcp any any -> 10.1.1.1 80 (msg:"command dot exe attempt"; content:"|63 6D 64 2E 65 78 65|"; flow:to_server; sid:2001; priority: 1;)`
- `alert tcp any any -> 10.1.1.1 80 (msg:"command dot exe attempt"; content:"|63| m |64 2E 65 78 65|"; flow:to_server; sid:2001; priority: 1;)`

This section explains the content modifiers, which you can use to change the way content option works. The content modifiers are specifications that the Sensor should apply on the content data before it looks for a match.

### nocase

Use this if you want the Sensor to look for the content data regardless of the case.

Syntax: `nocase;`

Example rule:

```
alert tcp any any -> 10.1.1.1 80 (msg: "command dot exe attempt"; content: "cMd.eXe"; nocase;
flow:to_server; sid:2002; priority: 1;)
```

### rawbytes

Use this if you want the Sensor to consider the content data as raw data. Then Sensor matches the content data against the non-normalized traffic.

Syntax: `rawbytes;`

For example, `http://www.example.com/big%20cars/blue%20colored%20cars/pictures.htm`

will normalize to `http://www.example.com/big cars/blue colored cars/pictures.htm`.

If you want the Sensor to find "blue%20color" in the traffic before normalization (that is, as seen in the "wire"), then your rule could be as shown below:

```
alert tcp any any -> 10.1.1.1 80 (msg: "example rule for rawbytes"; content:"blue%20color";  
rawbytes; flow:to_server; sid:2003;priority:1;)
```

If you do not specify `rawbytes` for the same rule, the Sensor will look for "blue%20color" in the normalized traffic, which it would never find.

### http\_client\_body

Use this modifier if you want the Sensor to look into only the normalized http client body.

Note the following:

- Content should precede `http_client_body`, and it applies only to the immediately preceding Content.
- If you use `http_client_body` with *offset* or *depth*, then these options are calculated from the beginning of the http payload. Consider the example rule below:

```
alert tcp any any -> any any (msg: "example rule for http_client_body"; content:"red";  
http_client_body; depth:30; sid:2007;priority:1;)
```

The Sensor looks for "red" within 30 bytes from the beginning of the http payload and not from the beginning of the http client body.

- Because this modifier works only for the normalized data, you cannot use it with `rawbytes` for the same content.

Example rule:

```
alert tcp any any -> any any (msg: "example rule for http_client_body"; content:"red";  
content:"blue"; http_client_body;sid:2007;priority:1;)
```

For this rule, the Sensor looks for "red" in the entire payload and for "blue" only in the normalized HTTP request.

### http\_cookie

The `http_cookie` modifier is supported in Trellix IPS.

### http\_header

In this case, the Sensor checks for the content only in the normalized HTTP request.

Note the following:

- Content should precede `http_header`, and it applies only to the immediately preceding Content.
- If you use `http_header` with *offset* or *depth*, then these options are calculated from the beginning of the payload. Consider that you created a rule as shown below:

```
alert tcp any any -> 10.1.1.1 80 (msg: "example rule for http_header"; content:"user";  
http_header; depth:100; sid:2008;priority:1;)
```

The Sensor looks for "user" within 100 bytes from the beginning of the http payload.

- You cannot apply the `http_header` and `rawbytes` on the same content.

Syntax: `http_header;`

Example rule:

```
alert tcp any any -> any any (msg:"example for http_header"; content:"red";
content:"user";http_header;sid:2008;priority:1;)
```

In this rule, the Sensor looks for "red" in the entire payload and for "blue" just in the normalized HTTP header part of the payload.

### http\_method

In this case, the Sensor checks for the content only in the normalized HTTP method section within a HTTP request.

Note the following:

- Content should precede `http_method`, and it applies only to the immediately preceding content.
- If you use `http_method` with *offset* or *depth*, then these options are calculated from the beginning of the http payload and not within a specific http field. Consider that you created a rule as shown below:

```
alert tcp any any -> any any (msg: "example rule for http_method"; content:"post"; http_method;
depth:5; sid:2009;priority:1;)
```

The Sensor looks for "post" within 5 bytes from the beginning of the http payload.

- You cannot apply the `http_method` and `rawbytes` to the same content.

Syntax: `http_method;`

Example rule:

```
alert tcp any any -> any any (msg:"example for http_method"; content:"red";
content:"PUT";http_method;sid:2009;priority:1;)
```

For this rule, the Sensor looks for "red" in the entire payload and whether the HTTP method is PUT.

### http\_uri

In this case, the Sensor checks for the content only in the normalized request URI section.

Note the following:

- Content should precede `http_uri`, and it applies only to the immediately preceding Content.
- Functionally, using `http_uri` is same as using the `uricontent` option explained later in this section.
- If you use `http_uri` with *offset* or *depth*, these options are calculated from the beginning of the http payload. Consider that you created a rule as shown below:

```
alert tcp any any -> any any (msg: "example rule for http_uri"; content:"red"; http_uri;
depth:50; sid:2009;priority:1;)
```

The Sensor looks for "red" within 50 bytes from the beginning of the http payload.

- You cannot apply `http_uri` and `rawbytes` on the same content.

Syntax: `http_uri;`

Example rule:

```
alert tcp any any -> any any (msg:"example for http_uri"; content:"red"; content:"blue"; http_uri;
sid:2009; priority:1;)
```

In this rule, the Sensor looks for "red" in the entire payload; "blue" just in the normalized request URI.

### http\_raw\_cookie

This searches the extracted unnormalized cookie header field of a HTTP request or a HTTP response. Since this is a content modifier to the previous content, there must be a content in the rule preceding the http\_raw\_cookie rule option.

#### Note

In a Snort custom attack, you cannot use http\_raw\_cookie modifier with rawbytes or http\_cookie modifiers for the same content.

Syntax: http\_raw\_cookie;

Example rule:

```
alert tcp any any -> any 80 (msg: "example rule for http_raw_cookie"; content:"red";  
content:"blue"; http_raw_cookie;sid:2099;priority:1;)
```

This rule searches only for blue in the extracted unnormalized cookie header field of a HTTP request.

### http\_raw\_header

This searches the extracted unnormalized header fields of a HTTP request or a HTTP response. Since this is a content modifier to the previous content, there must be a content in the rule preceding the http\_raw\_header rule option.

#### Note

In a Snort custom attack, you cannot use http\_raw\_header modifier with rawbytes or http\_cookie modifiers for the same content.

Syntax: http\_raw\_header;

Example rule:

```
alert tcp any any -> any 80 (msg: "example rule for http_raw_header"; content:"red";  
content:"blue"; http_raw_header;sid:2199;priority:1;)
```

This rule searches only for blue in the extracted unnormalized header fields of a HTTP request or HTTP response.

### http\_raw\_uri

This searches the unnormalized request URI field. Since this is a content modifier to the previous content, there must be a content in the rule preceding the http\_raw\_uri rule option.

#### Note

In a Snort custom attack, you cannot use http\_raw\_uri modifier with rawbytes or http\_cookie modifiers for the same content.

Syntax: http\_raw\_uri;

Example rule:

```
alert tcp any any -> any 80 (msg: "example rule for http_raw_uri"; content:"red"; content:"blue";  
http_raw_uri;sid:2499;priority:1;)
```

This rule searches only for blue in the unnormalized URI.



### http\_stat\_code

This searches the extracted status code field from a HTTP response. Since this is a content modifier to the previous content, there must be a content in the rule preceding the http\_stat\_code rule option.

#### Note

In a Snort custom attack, you cannot use http\_stat\_code modifier with rawbytes modifier for the same content.

Syntax: http\_stat\_code;

Example rule:

```
alert tcp any any -> any 80 (msg: "example rule for http_stat_code"; content:"red"; content:"200";  
http_stat_code;sid:7199;priority:1;)
```

This rule searches for status 200 in the extracted status code field of a HTTP response.

### uricontent

Use this keyword to search in the normalized request URI field. Note that if the uricontent value has anything that is normalized, then the rule will return negative. For example, if the value contains %20, then the Sensor does not raise an alert because it will check for %20 in the normalized uricontent, which it will never find. To check for non-normalized content, consider using the content option with rawbytes.

Note the following:

- This is same as using content with http\_uri.
- If you use uricontent with *offset* or *depth*, these options are calculated from the beginning of the http payload. Consider that you created a rule as shown below:

```
alert tcp any any -> any any (msg: "example rule for uricontent"; uricontent:red; depth:50;  
sid:2009;priority:1;)
```

The Sensor looks for "red" within 50 bytes from the beginning of the http payload.

- You cannot use rawbytes with uricontent.

Syntax: uricontent:[!]<content string>;

Example rule:

```
alert tcp any any -> any any (msg:"example for uricontent"; content:"red"; uricontent:"hello  
+world"; sid:2033; priority:1;)
```

In this rule, the Sensor looks for "red" in the entire payload; "hello+world" just in the normalized request URI.

### urilen

This is a condition that enables you to specify the exact, minimum, or maximum lengths, or range of URI lengths to match.

Syntax:

```
urilen: int<>int;
```

```
urilen: [<,>] <int>;
```

Examples:

```
urilen:5
```

Matches URIs that are 5 bytes in length.

Example rule:

```
alert tcp any any -> any any (msg:"example for urilen"; content:"red"; content:"blue"; urilen:5;
sid:2009; priority:1;)
```

For this rule, the Sensor checks for the strings "red" and "blue" in the entire payload and also sees if the request URI is exactly 5 bytes in length.

```
urilen: < 5
```

Matches URIs that are less shorter than 5 bytes.

```
urilen: 5<>10
```

Matches URIs that are >= 5 bytes and <= 10 bytes.

PCRE

You can use Perl Compatible Regular Expressions (PCRE) with the related modifiers in the Snort rules. If you use PCRE in a rule, then it should be preceded by a Content. The Content is the triggering token for PCRE options following it. That is, the Sensor checks for the PCRE part of the rule only when the traffic is positive for the preceding Content. Do not use negation operator on the triggering token.

 Important

PCRE options are resource-intensive on the Sensor. Trellix recommends that you check if there are other options to achieve the same result.

```
Syntax: pcre: [!] "(/(<regex>/) [ismxAEGUPHMCOIDKYS])";
```

- / is the only supported delimiter.
- For the PCRE options, by default, the Sensor considers only the first 256 bytes from the beginning of the triggering token.
- When you import or validate the Snort rules in the Manager, you can know the unsupported PCRE options due to which the rules failed to convert successfully.

The following table explains the PCRE-related options supported in Trellix IPS.

| Modifier | Description                                                                                                                                                                                                                                                                                                         |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| i        | The Sensor ignores the case of the corresponding string.                                                                                                                                                                                                                                                            |
| s        | Use this to consider new lines in the dot meta-character.                                                                                                                                                                                                                                                           |
| m        | This is used with the anchors (^ and \$).<br>Use m with ^ if you want the Sensor to check for the string immediately after a new line as well as at the beginning of the buffer.<br>Use m with \$ if you want the Sensor to check for the string immediately before a new line as well as at the end of the buffer. |
| x        | Use this if you want the Sensor to ignore any empty space characters in the buffer unless it is escaped or inside a character class.                                                                                                                                                                                |
| A        | The Sensor checks if the string is at the beginning of the buffer. This is the same as ^.                                                                                                                                                                                                                           |
| E        | This is used with \$. Without E, \$ also matches immediately before the final character if it is a new line but not before any other new lines.                                                                                                                                                                     |
| G        | This inverts the "greediness" of the quantifiers so that they become greedy only when followed by a question mark.                                                                                                                                                                                                  |

| Modifier | Description                                                                                                                                                                                                                                                                                                                                  |
|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| I        | Matches the unnormalized HTTP request URI buffer. This is similar to <code>http_raw_uri</code> in function. Snort does not allow using this modifier along with the HTTP request uri buffer modifier for the same content.                                                                                                                   |
| C        | Matches normalized HTTP request or HTTP response cookie. This is similar to <code>http_cookie</code> in function. This is not allowed with the unnormalized HTTP request or HTTP response cookie modifier for the same content.                                                                                                              |
| H        | Matches normalized HTTP request or HTTP response header. This is similar to <code>http_header</code> . This modifier is not allowed with the unnormalized HTTP request or HTTP response header modifier for the same content. For SIP message, matches SIP header for request or response, similar to how <code>sip_header</code> functions. |
| P        | Matches unnormalized HTTP request body. This is similar to how <code>http_client_body</code> functions. For SIP message, matches SIP body for request or response, similar to how <code>sip_body</code> works.                                                                                                                               |
| D        | Matches unnormalized HTTP request or HTTP response header, similar to <code>http_raw_header</code> . This modifier is not allowed with the normalized HTTP request or HTTP response header modifier for the same content.                                                                                                                    |
| M        | Matches normalized HTTP request method, similar to <code>http_method</code>                                                                                                                                                                                                                                                                  |
| K        | Matches unnormalized HTTP request or HTTP response cookie, similar to <code>http_raw_cookie</code> . This modifier is not allowed with the normalized HTTP request or HTTP response cookie modifier for the same content.                                                                                                                    |
| S        | Matches HTTP response status code, similar to <code>http_stat_code</code>                                                                                                                                                                                                                                                                    |
| Y        | Matches HTTP response status message, similar to <code>http_stat_msg</code>                                                                                                                                                                                                                                                                  |

#### Note

\X, \P, \K, \U, \R, and \C escape sequences and back references are not supported.

Example rule:

```
alert tcp any any -> any any (msg: "command dot exe attempt"; content: "user"; pcre: "/cMd.eXe/i";
sid:2030; priority: 1;)
```

This rule first checks for the string, "user". Here, "user" is the triggering token. If "user" is found, the Sensor then checks for the PCRE "cmd.exe" in the first 256 bytes from the beginning of the buffer.

### sip\_method

This rule option enables you to check for Session Initiation Protocol (SIP) request methods. In the same option, you can specify multiple SIP request methods separated by commas. In this case, it is considered as an OR condition. That is, the rule triggers if there is a match for any of the mentioned request methods.

Syntax:

```
sip_method:<method>|<method-list>
```

The following request methods are supported:

- invite
- cancel
- ack
- bye
- register
- options
- refer
- subscribe
- update
- join
- info
- message
- notify
- prack

The `!` operator is supported. However, if you use `!`, you can specify only one method in one `sip_method` option.

Examples:

- `sip_method:invite` — This checks for the invite request method.
- `sip_method:!invite` — The condition is true if the method is anything other than invite.
- `sip_method:invite, cancel, bye` — The condition is true if the method is invite, cancel, or bye.
- `sip_method:!invite; sip_method:!cancel` — The condition is true if the method is anything other than invite and cancel.

Example rule:

```
alert udp any any -> 10.1.1.1 5060 (msg:"Example rule"; flow:to_server; sip_method:invite;
content:"SIP/2.0"; nocase; priority:1; sid:20189; rev:1;)
```

### sip\_stat\_code

This rule option enables you to check for SIP response status codes. The condition matches if any of the specified status code is present in the SIP response.

Syntax:

```
sip_stat_code:<code>|<code>,<code>
```

The following request methods are supported:

Examples:

- `sip_stat_code:400` — This checks for the status code 400.
- `sip_stat_code:180,182` — The condition matches if the status code is 180 or 182.
- `sip_stat_code:4` — This condition looks for 4xx. That is, it is true if the method is anything from 400 to 599. The numbers 1 to 6 are expressed as 1xx, 2xx, 3xx, and so on where, for example, 1xx corresponds to the range 100 - 199.

Example rule:

```
alert udp any any -> 10.1.1.1 5060 (msg:"Example rule"; flow:to_server; sip_stat_code:401;
content:"SIP/2.0"; nocase; priority:1; sid:20289; rev:1;)
```

### sip\_header

This rule option searches only the extracted header fields of a SIP request or response.

Syntax: `sip_header;`

Example rule:

```
alert udp any any -> 10.1.1.1 5060 (msg:"Example rule"; flow:to_server; sip_header; content:"SIP/
2.0"; nocase; priority:1; sid:20290; rev:1;)
```

### sip\_body

This rule option points the Sensor to the beginning of the body fields of a SIP message.

Syntax: `sip_body;`

Example rule:

```
alert udp any any -> 10.1.1.1 5060 (msg:"Example rule"; flow:to_server; sip_body; content:"SIP/
2.0"; nocase; priority:1; sid:20491; rev:1;)
```

### ssl\_version

This rule option can check for the SSL version numbers exchanged between the server and the client during the handshake. In the same option, you can check for multiple SSL versions separated by commas. In this case, it is considered as an OR condition. That is, the rule triggers if there is a match for any of the versions. To check for an AND condition involving two or more versions, use separate `ssl_version` rule options.

Syntax:

`ssl_version:<version-list>`

The `!` operator is supported. However, if you use `!`, you can specify only one method in one `sip_method` option.

Examples:

- `ssl_version:sslsv2` — This checks for SSL version 2.
- `ssl_version:!sslsv3` — The condition is true for any version other than SSL version 3.
- `ssl_version:tls1.0,tls1.1,tls1.2` — The condition is true if the SSL version is TLS 1.0, TLS 1.1, or TLS 1.2.
- `ssl_version:!tls1.0; ssl_version:!tls1.1` — The condition is true if the SSL version is anything other than TLS 1.0 and TLS 1.1.

Example rule:

```
alert tcp any any -> 10.1.1.2 443 (msg:"Example rule"; flow:to_server; ssl_version:sslsv2;  
content:"|0B|"; priority:1; sid:20389;rev:1;)
```

### ssl\_state

This option is not relevant for a Snort rule in Trellix IPS, and is not supported. If a rule that you imported contained this option, it is displayed in the Conversion Notes section of the rule. The Sensor ignores this option when it checks the traffic against this rule.

### modbus\_func

This rule option checks for the specified function code in the in a Modbus header. You can specify either the code number in the decimal format or the equivalent string.

Syntax: `modbus_func:<code>`

The following are the supported values for code:

- |                                       |                                                 |
|---------------------------------------|-------------------------------------------------|
| • A number ranging from 0 to 255      | • <code>get_comm_event_log</code>               |
| • <code>read_coils</code>             | • <code>write_multiple_coils</code>             |
| • <code>read_discrete_inputs</code>   | • <code>write_multiple_registers</code>         |
| • <code>read_holding_registers</code> | • <code>report_slave_id</code>                  |
| • <code>read_input_registers</code>   | • <code>read_file_record</code>                 |
| • <code>write_single_coil</code>      | • <code>write_file_record</code>                |
| • <code>write_single_register</code>  | • <code>mask_write_register</code>              |
| • <code>read_exception_status</code>  | • <code>read_write_multiple_registers</code>    |
| • <code>diagnostics</code>            | • <code>read_fifo_queue</code>                  |
| • <code>get_comm_event_counter</code> | • <code>encapsulated_interface_transport</code> |

### Examples:

- `modbus_func:1`
- `modbus_func:read_discrete_inputs`

### Example rule:

```
alert tcp any any -> 10.1.1.5 502 (msg:"Example rule"; flow:to_server;  
modbus_func:write_multiple_coils; byte_test:2,>,1968,10; reference:url,www.modbus.org/docs/  
Modbus_Application_Protocol_V1_1b.pdf; classtype:protocol-command-decode; priority:1;  
sid:20589;rev:1;)
```

### modbus\_data

This rule option points the Sensor to the beginning of the data field in a Modbus request or response.

Syntax: `modbus_data;`

### Example rule:

```
alert tcp any any -> 10.1.1.7 502 (msg:"Example rule"; flow:to_server; modbus_data;  
content:"example content"; nocase; priority:1; sid:20481;rev:1;)
```

### dnp3\_func

This rule option checks against the function code of a DNP3 request or response header. You can specify either the code number in the decimal format or the equivalent string.

### Syntax:

`dnp3_func:<code>`

The following are the supported values for code:

- |                                  |                       |
|----------------------------------|-----------------------|
| • A number ranging from 0 to 255 | • stop_appl           |
| • confirm                        | • save_config         |
| • read                           | • enable_unsolicited  |
| • write                          | • disable_unsolicited |
| • select                         | • assign_class        |
| • operate                        | • delay_measure       |
| • direct_operate                 | • record_current_time |
| • direct_operate_nr              | • open_file           |
| • immed_freeze                   | • close_file          |
| • immed_freeze_nr                | • delete_file         |
| • freeze_clear                   | • get_file_info       |
| • freeze_clear_nr                | • authenticate_file   |
| • freeze_at_time                 | • abort_file          |
| • freeze_at_time_nr              | • activate_config     |
| • cold_restart                   | • authenticate_req    |
| • warm_restart                   | • authenticate_err    |

- initialize\_data
- initialize\_appl
- start\_appl
- response
- unsolicited\_response
- authenticate\_resp

Examples:

- dnp3\_func:2;
- dnp3\_func:get\_file\_info;

### dnp3\_data

This rule option points the Sensor to the beginning of the application-layer fragment so that the Sensor can execute the other rule options.

Syntax:

dnp3\_data;

Example:

```
dnp3_data; content:"example content";
```

### Non-payload rule options

This section explains Short non-payload rule options.

#### flags

Use this if you want the Sensor to check for TCP flags. You can check for the following flags:

- F - FIN (LSB in TCP Flags byte)
- S - SYN
- R - RST
- P - PSH
- A - ACK
- U - URG
- 1 - Reserved bit 1 (MSB in TCP Flags byte)
- 2 - Reserved bit 2
- 0 - No TCP Flags Set

You can apply the following operators on the flags option:

- + — Checks if all the specified flags are set
- \* — Checks if at least one of the specified flags is set
- ! — Checks if none of the flags are set

Syntax: flags:[!|\*|+]<FSRPAU120>[,<FSRPAU120>];

Example rule:

```
alert tcp 10.1.1.1 any -> any any (msg:"example for flags"; flags:R; sid:2019; priority:3;)
```

This rule raises an alert when the host 10.1.1.1 resets a TCP connection.

#### flow

This option enables you to write rules specific to a direction of the traffic flow. For example, you can write a rule that the Sensor applies only on the traffic from the clients.

The following table describes the flow options that you can use in Trellix IPS:

| Option      | Description                                                                       |
|-------------|-----------------------------------------------------------------------------------|
| to_client   | Rule is applicable only to the response traffic from the server in a TCP session. |
| to_server   | Rule is applicable only to the request traffic from the client in a TCP session.  |
| from_client | Same as to_server                                                                 |
| from_server | Same as to_client                                                                 |

Syntax: flow: [, (to\_client|to\_server|from\_client|from\_server)];

Example rule:

```
alert tcp !$HOME_NET any -> 10.1.1.1 80 (msg:"example for flow"; flow:to_server;  
content:"cmd.exe";sid:2015; priority:1;)
```

For this rule, the Sensor checks for cmd.exe in the request traffic from outside network to 10.1.1.1.

#### Notes:

- Make sure you specify the flow option in the Snort custom attacks to avoid false-positives.
- If you do not specify flow for a TCP rule, then it is saved in the Manager with Conversion Result, "warning" and State **Published**.

### itype

Use this if the Sensor is to check for a specific ICMP type value.

Syntax: itype: [<|>]<number>[<><number>];

### icode

Use this if the Sensor is to check for a specific ICMP code value.

Syntax: icode: [<|>]<number>[<><number>];

Example rule:

```
alert icmp !$HOME_NET any -> $HOME_NET any (msg:"example for itype and icode"; itype 8; icode:0;  
sid:2022; priority:3;)
```

This rule is triggered for inbound traffic where the type is 8 and code is 0.

### icmp\_seq

This is to check for a specific ICMP sequence value.

Syntax: icmp\_seq:<number>;

Example rule:

```
alert icmp !$HOME_NET any -> $HOME_NET any (msg:"example for icmp_seq"; icmp_seq:0; sid:2041;  
priority:3;)
```

## Post-detection rule options

Post-detection rule options are irrelevant for Snort Custom Attacks. After an attack is saved in the Manager server, you can configure the response actions for the attack just like you would for any other attack definition in Trellix IPS.



## Managing Snort custom attacks

The Trellix IPS supports the Trellix IPS Snort and the Suricata Snort. This section provides information on how to create and maintain Trellix IPS Snort and Suricata custom attacks. In Trellix IPS, you can create the custom attacks in one of two ways:

- Construct the Snort rules, one at a time, directly in the Custom Attack Editor.
- Construct multiple Snort rules in a file and import the file into the Manager. You may want to consider this method when you have a large number of attack definitions to create or if you want to use the Snort rules from a source such as the Snort user community.

### Important

Before you create custom attacks, make sure you have reviewed and understood the considerations and best practices to be followed when using Snort rules in Trellix IPS.

See also

*Management of custom attacks from the Central Manager on page 1670*

## Snort Engines

Trellix IPS NS-series Sensors support either of the following Snort engines:

- Trellix IPS Snort
- Suricata Snort

### Note

A Sensor can run only one of the above engines at a time.

The Trellix IPS Snort engine is the existing Snort engine available on Trellix IPS, whereas the Suricata Snort engine is new. If you need support for a wide range of Snort rules, use the Suricata Snort engine.

You can select a Snort engine at the global level and override the selection per admin domain and/or Sensor, as necessary. The **Advanced Device Settings** page at the global and device levels allows you to select the engine you prefer to use.

The screenshot shows the 'Advanced Device Settings' page in the Trellix IPS management console. The breadcrumb trail at the top is '/ > IPS Device Settings > Advanced Device Settings'. The page has a dark theme. Under the 'Advanced Device Settings' header, there is an information icon (i) and three settings: 'Pre-Attack Bytes to Capture' set to '128 bytes', 'Inspect Tunneled Traffic' as an unchecked checkbox, and 'Snort Rule Engine' set to 'Trellix IPS Snort'. Below these is a section header 'CLI' with an upward arrow icon. Under the 'CLI' section, there are four settings: 'CLI Activity Logging' set to 'Disabled', 'Show CPU Usage in the CLI' as an unchecked checkbox, 'Log SSH Access to the CLI' as an unchecked checkbox, and 'Restrict SSH Access to the CLI' as an unchecked checkbox. Each setting has an information icon (i) to its right. A 'Save' button is located at the bottom right of the settings area.

 **Note**

The default selection is **Trellix IPS Snort**.

## Trellix IPS Snort

The Trellix IPS Snort engine is the existing Snort engine that is available on the Trellix IPS.

Consider the following factors when selecting the Trellix IPS Snort engine:

- The NS-series Sensors can support up to 5000 Snort rules. If you require support for more rules, contact Trellix Support.
- \X, \P, \K, \U, \R, and \C escape sequences and backreferences are not supported.
- Isdataat with rawbytes modifier is not supported.
- Norm and raw modifiers are not supported in urilen.
- Snort rules cannot be created on ASN1 decoded content.
- If the length of the content or uricontent (token length) is more than 96 bytes and less than or equal to 256 bytes, the target device type is automatically set to NS-series.

- The following PCRE constructs are not supported:
  - Lookahead and lookbehind assertions.
  - Backreferences and capturing subexpressions.

```
alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"WEB-CLIENT Adobe Acrobat getCosObj file
overwrite
attempt"; flow:established,to_client; flowbits:isset,http.pdf; file_data; content:".write|28|";
nocase;
content:".getCosObj|28|"; distance:0; nocase; pcre:"/([A-Z\d_]+)\.write\x28.*?\1\.getCosObj\x28/
smi"; reference:
cve,2011-2442; reference:url,www.adobe.com/support/security/bulletins/apsb11-24.html;
classtype:attempted-user;
sid:20156; rev:1;)
```

In this example rule, \1 is the backreference.

- Subroutine references and recursive patterns.
- Conditional patterns.

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"WEB-IIS ASP.NET 2.0 cross-site
scripting
attempt"; flow:to_server,established; content:"__LASTFOCUS="; fast_pattern:only; pcre:"/
__LASTFOCUS=(?!([_a-z]\w*))
([\x26\x3B]|$))/i"; reference:bugtraq,20337; reference:cve,2006-3436;
reference:url,www.microsoft.com/technet/security
/bulletin/MS06-056.msp; classtype:attempted-user; sid:8700; rev:5;)
```

In this example rule, !([\_a-z]\w\*)([\x26\x3B]|\$) is the conditional pattern.

- Unicode character properties \p{xx} and \P{xx}.
- Possessive quantifiers.

The following are additional limitations regarding PCRE in Trellix IPS Snort:

- In a Snort custom attack, the maximum value for repetition counter in PCRE is 256. If it exceeds this number, the error message *Pattern too large* is displayed.
- Atomic grouping and possessive quantifiers are not supported.



#### Note

PCRE rules validation is included in the Custom Attack Editor. So, any rule containing invalid or unsupported PCRE options fail to import. You can find the reason in the error message. After you correct these rules, use the **Test Compile** feature to check if they are valid and compatible. You can select the required attacks and select **Other Actions | Test Compile**.

The following constructs are not supported in Trellix IPS Snort:

- |                          |                 |
|--------------------------|-----------------|
| • ack                    | • ftpbounce     |
| • asn1                   | • http_stat_msg |
| • bidirectional operator | • icmp_id       |
| • byte_extract           | • id            |
| • byte_jump              | • ip_proto      |
| • byte_test              | • ipopts        |
| • cvs                    | • isdataat      |
| • dce_iface              | • metadata      |
| • dce_opnum              | • modbus_unit   |

- |                 |                      |
|-----------------|----------------------|
| • dce_stub_data | • offset             |
| • depth         | • port / IP negation |
| • distance      | • rpc                |
| • dnp3_ind      | • sameip             |
| • dnp3_obj      | • seq                |
| • dsize         | • stream_size        |
| • fast_pattern  | • tos                |
| • flowbits      | • ttl                |
| • fragbits      | • window             |
| • fragoffset    | • within             |

The following are not supported in Trellix IPS Snort, but an equivalent option is available in Trellix IPS:

- Rules using ftpbounce keyword are not supported. However, the Trellix IPS signature sets provide adequate protection against FTP bounce attacks.
- Rules to detect CVS attacks are not supported. However, Trellix IPS signature sets can protect your network against most of the CVS attacks.
- Regarding Sensor response actions, you cannot use any post-detection rule options such as Logto, Session, or Tag. Instead, you can configure the required response action after the rule is saved in the Manager database. This is the same as how you would configure the response action for any other attack definition in Trellix IPS.
- Preprocessor plugins are not supported.
- Snort's multi-event logging (event queue) is not supported.
- For event thresholding and alert suppression, you can use only the equivalent features in Trellix IPS and not the options in Snort.

### Classification types

It is mandatory that you define the priority for each Snort Custom Attack. Priority is one of the parameters that the Manager uses when it categorizes a Snort Custom Attack. To specify the priority, you can either use the priority keyword directly in the rule or use a classification type in the rule. However, this classification type should either be available in the Manager or declared in a file up front. That is, the Manager must read the classification type declaration before it reads the rule that uses it.



#### Note

You can view the classification types that are currently available in the Manager. You can also re-submit rules for conversion with the currently available values.

Similar to macros, you can define the classification types in a .conf, .config, or .rules file and import it into the Manager.

In the .conf, .config, or .rules file, define the classification type using the syntax as shown below:

```
config classification:<class name>,<class description>,<default priority>
```

Example: config classification: brute-force,attempted brute force,1

Note the following:

- The priority value 1 is mapped to a severity of high in Trellix IPS, 2 to medium, 3 to low, and 4 and above to informational.
- The default priority value that you specify in a classification type can be overridden by using the priority keyword in the rule.

### Suricata Snort

The Suricata Snort engine provides a dedicated Snort environment and supports most of the open-source Snort constructs that are available in the public domain. This allows you to import most custom and third-party Snort rules without modification.

The usage of some of the constructs differ in Suricata Snort when compared with the open-source Snort. For more information on the differences, see <http://suricata.readthedocs.io/en/latest/rules/differences-from-snort.html>

Some of the open-source Snort constructs that are not supported in the Suricata Snort are as follows:

- activated\_by
- activates
- byte\_jump bitmask
- content http\_header
- content http\_raw\_cookie
- cvs invalid-entry
- flowbits reset
- flowbits setx
- gtp\_info
- gtp\_type
- gtp\_version
- http\_encode\*
- ipopts lsrr
- logto
- modbus\_\*
- pcre modifiers:
  - D
  - K
  - SIP (combination of S, I, and P modifiers)
- protected\_content\*
- react
- resp
- sip\_\*



#### Note

There can be other open-source constructs which are not supported in the Suricata Snort.

The following is the list of considerations when using the Suricata Snort:

- You can import only the Suricata rules having tcp, udp, ip, and icmp protocols without modification from the Emerging Threats.
- You cannot push the Suricata Reference config file to the Sensor.
- You cannot import rule variables from yaml file from the Emerging Threats.
- You cannot exclude a rule for a particular snort engine.

### Configuring Snort engine at a global level

To configure the appropriate snort engine at a global level, perform the following steps.

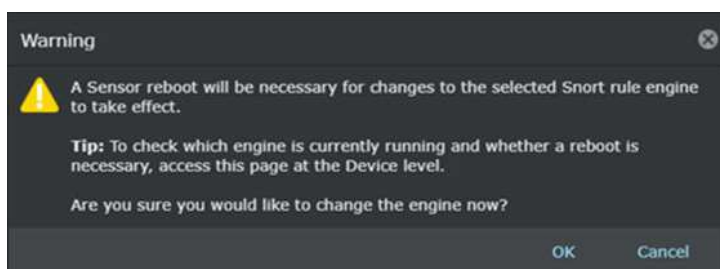
#### Task

- 1 Go to **Devices** | **<Admin Domain Name>** | **Global** | **IPS Device Settings** | **Advanced Device Settings**.

The default engine selected is the Trellix IPS Snort engine.

- 2 To select the snort engine, click the **Snort Rule Engine** dropdown and select the **Suricata Snort**.

A pop-up appears asking you to confirm your changes and informing you that a reboot will be necessary for the change to take effect.



- 3 Click **OK** and then click **Save** at the bottom of the page.

- 4 Reboot the Sensors which require this change.

After the reboot, the **Advanced Device Settings** page of the Sensor displays Suricata Snort.

### Configuring Snort engine at a device level

To configure the appropriate snort engine at a device level, perform the following steps.

#### Task

- 1 Go to **Devices** | **<Admin Domain Name>** | **Global** | **IPS Device Settings** | **Advanced Device Settings**.

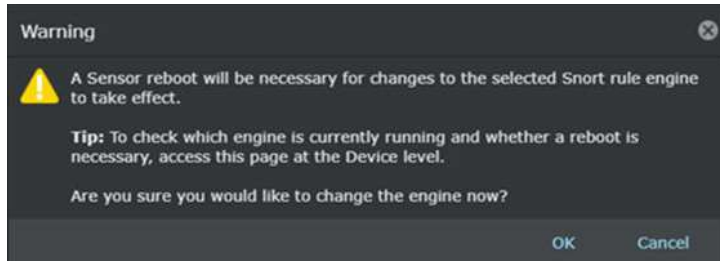
The **Advanced Device Settings** page appears.

- 2 Uncheck the **Inherit Settings?** checkbox to get access to modify settings in this page.

- 3 Select **Inspected Tunneled Traffic** if you want to inspect GRE, GTP, and tunneled IPv4 and IPv6 traffic.

- 4 To select the snort engine, click the **Snort Rule Engine** dropdown and select **Suricata Snort**.

A pop-up appears asking you to confirm your changes and informing you that a reboot will be necessary for the change to take effect.



- 5 Click **OK** and then click **Save** at the bottom of the page.

- 6 Reboot the Sensors which require this change.

After the reboot, the **Advanced Device Settings** page of the Sensor displays Suricata Snort.

### CLI commands to monitor Snort

The Sensor CLI provides the following commands to monitor Snort.

- `show snort config`
- `show suricata enginestats`
- `show suricata sbstats`

For more information about the commands, see the *CLI commands* section.

### Troubleshooting Tips

This section describes the issues that you might face when working with the Snort rules and the corresponding troubleshooting tips.

#### Issue: Suricata Snort rules are imported successfully in the Manager but the Sensor does not generate alerts.

Cause: The Snort rules might be failing in the Sensor.

The Sensor does not generate alerts for the failed rules. The Sensor sends the information about the failed rules to the Manager. Information about the failed rules in the Sensor is available in a log file. To view the log file, go to the **Manager** | **<Admin domain>** | **Troubleshooting** | **Logs** page in the Manager.

You can also verify the number of rules processed by a Sensor by using the `show suricata sbstats` command. For more information, see the [CLI commands to monitor Snort](#) on page 1593 section.

#### Issue: An alert does not contain the packet information.

Cause: It is possible that the alert might be generated for a buffered packet attack.

The Suricata engine integrated into the Sensor performs packet buffering. When a buffered packet attack is detected and an alert is generated, the Sensor may not have the actual packet information if the alert was raised based on the buffered packet data. In such a scenario, the alert will not have any packet information.

### Best practices

This section details the best practices that you **must** follow when you use Snort Custom Attacks:

- Do not use a Snort Custom Attack if there is an equivalent available in the signature set.
- Make sure that the content option value is more than one byte. If you import a rule with a one-byte content, it will fail to import. The longer the content value, the accurate the detection will be. However, the maximum length of a content or uricontent for NS-series Sensors is 256 bytes.
- Make sure the content option does not contain any generic values identified by Trellix IPS (some examples are listed below). Such rules can severely impact Sensor performance.
  - GET
  - POST
  - Host
  - User-Agent
- For better accuracy and performance, Trellix recommends that you use the Custom Attack Editor to create custom attack definitions as opposed to importing Snort rules.
- If you are using `byte_test` or `byte_jump`, use them in relation to a content match.
- Specify the classtype or priority to all rules. This enables the Manager to determine the severity for the rule. Understand how the Manager categorizes a Snort Custom Attack to publish it in the rule sets.
- If you are importing the Snort rules, import them from files that are accordingly named. For example, import HTTP rules from a file named `http.rules` file. In these rules, do not specify the destination port; the Sensor automatically detects protocols running on non-standard ports and applies the rule to the corresponding traffic. If you specify a port number, the Sensor applies the rule only to the traffic destined for that port.
- If you create the Snort rule in the Custom Attack Editor, or if you import it from a generically named file (like `myrules.rules`), it is very important that you specify the destination port number.
- Specify the revision number for all rules.
- For TCP rules, specify the flow.
- In a rule, do not specify the same value for more than one Content option. For example, do not use a Snort Custom Attack such as the following: `alert tcp any any -> 10.1.1.1 80 (msg:"Example rule"; content:"private"; content:"private"; priority:1;sid:20209;rev:1;)`.

### Create snort custom attacks

The following is the high-level approach for creating Snort Custom Attacks.

#### Task

- 1 Have a clear understanding on what you are trying to achieve through these attacks. See if there are any alternatives or better methods to achieve the same. Identify the related protocols, applications, hardware and software platforms for the attack. Make sure you have all the required information in hand.
- 2 Understand the mechanics of a Snort Custom Attack. You need to have a strong understanding of network concepts as well as Snort rules language to create effective Snort Custom Attacks.
- 3 Take a quick tour of the Custom Attack Editor. Get familiar with the Custom Attack Editor - the tool that you will use to manage Custom Attacks. Understand the interfaces related to Snort Custom Attacks.



- 4 Verify the required variables. Check if the variables that you plan to use are available in the Sensor.
- 5 Create Snort Custom Attacks. There are two methods:
  - You can import the rules from a file.
  - You can construct the Snort rules directly in the Custom Attack Editor.
- 6 Check the rules that failed to convert. The Sensor automatically converts all the valid rules, which you imported or wrote, to Trellix IPS's proprietary format. Note that some rules could have failed to convert. Troubleshoot and fix the rules that failed to convert.



### Note

If you are creating or importing Suricata Snort rules, ensure that one of **TCP**, **UDP**, **IP**, and **ICMP** protocols is used in the Snort rules.

- 7 Save the converted rules in the Manager database.
- 8 Once saved in the database, the rules are like any other custom attack definition. For example, you may want to customize the response action for the saved Snort rules, or you may want to delete a Snort rule from the database.
- 9 Update the respective Sensors with the changes.  
The Sensors raise alerts based on the saved Snort rules. You can view these alerts in the Attack Log.

## Variables

There can be some values that you may need to mention in all or most of the rules. For example, the target subnet could be the same for many rules. Instead of repeating it in each rule, you can define a variable for the subnet, and then use the variable in the rules. The advantage with using variables is when you want to modify the value. For example, if the target subnet is now different, then instead of modifying each rule, you can just redefine the variable with the new value.

You can use variables for the following:

- Path to a file: When you import Snort rules from a file, you can include other files from within the file that you are importing. Then rules from all these files are automatically imported. When you include a file within another file, you need to provide the absolute or the relative path to the file being referenced. Alternatively, you can use a variable for the path and mention it instead of the mentioning the path to the file.
- IP addresses: You can use variables for the source and target IP addresses or subnets.
- Port numbers: You can use variables for the source and target port numbers.

See also

[Import snort rules through a conf file on page 1535](#)

[Import snort rules through a rules file on page 1537](#)

## Define the snort variables

You can define the snort variables in Trellix IPS and then use it in the rules.

### Task

- 1 Define the variables with the appropriate values in a file.
- 2 Import the file into the Sensor.

To define the variables in a file:

- a** Create a text file and change its file extension to `.rules` or `.conf`.

Assume that you have named it `variables.conf`

- b** In the text file, define the variables as explained below

- Use the **var** keyword to define a variable for a file path, IP addresses, and ports.

For example, `var RULE_PATH ../rules`.

In this example, `RULE_PATH` is the variable name and its value is the relative path to a folder named "rules".

- Use the **ipvar** keyword to define a variable for IP addresses. Some examples are:

- `ipvar INSIDE_NETWORK [10.1.1.0/24, !10.1.1.22, 11.1.1.1, 12.1.1.0/24]`

- `ipvar EXAMPLE1 [$INSIDE_NETWORK, !10.1.1.23]`

- `ipvar EXAMPLE2 [$EXAMPLE1]`

- `ipvar EXAMPLE3 [1.1.1.1, 2.2.2.0/24, ![2.2.2.2, 2.2.2.3]]`

- Use the **portvar** keyword to define the port numbers

For example, `portvar EXAMPLE_PORTS [100, 102, 150:160, !155]`

In this example, the value of `EXAMPLE_PORTS` is 100, 102, 150 through 160 except 155.

- c** Save the `variables.conf` file.

To import the variables file into Sensor:

- a** In the Custom Attack Editor, from the **Snort Format** tab, click **Snort Variables**. Alternatively, to import select **Other Actions | Import**.

- b** Locate `variables.conf` and click **Open**.

If you do not see the conf file at the location where you saved, check the **Files of Type** field in the Open dialog.

- c** Click **Open**.

The Import Status may show zero for all the fields. Click **OK**.

- d** Select **Snort Format** to make sure the variables are imported with the values you specified in the `variables.conf` file.

- e** If a variable that you imported is already available in the database, it is assigned the value from the current import.

## Viewing the Snort variables

You can use the **Snort Variables** feature to view the names and values of the variables and classification types that are available in the Manager database.

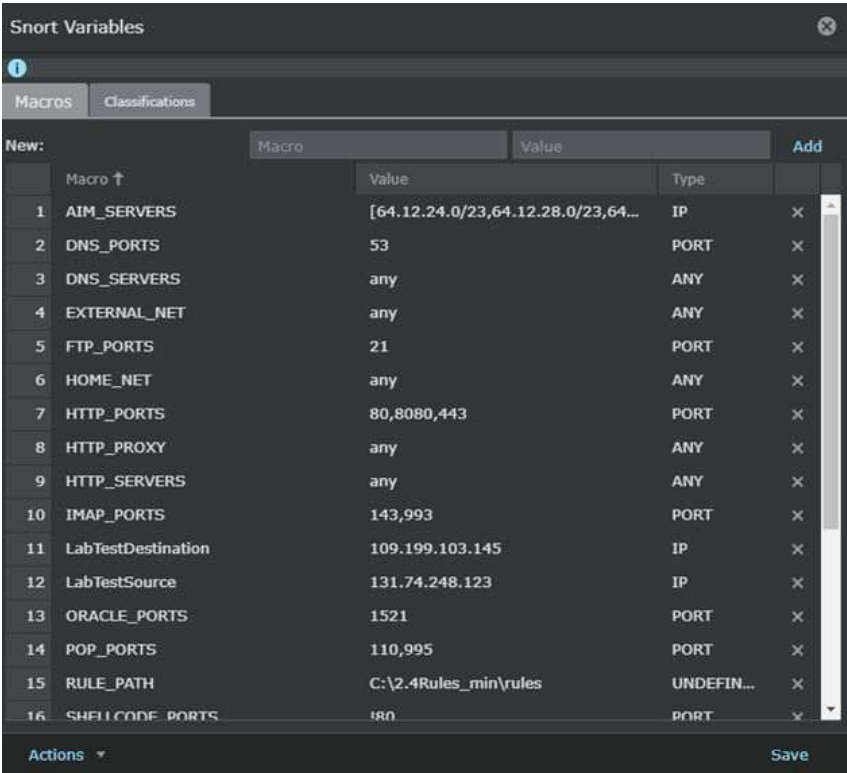
This feature enables you to the following:

- Check if you are using valid macros and classification types when you create or import a Snort Custom Attack. (You cannot create a Snort Custom Attack that contains an undefined macro or classification type. If you import a rule with an undefined macro or classification type, it will fail validation.)
- Verify if a macro or classification type that you want to define is already available.
- Verify if an import of macro or classification types was successful.
  - Add or delete macros of a classification type.

To view the names of the macros and classification types in the Manager:

Task

- In the Custom Attack Editor from the **Snort Format** tab, click **Snort Variables**.  
The **Snort Variables** dialog opens.



Import variables

To import all the listed variables, classification types, and references:

- 1 Click **Actions | Import**.  
This imports the variables to the Manager client.
- 2 Click **Save**.

This imports the variables to the Manager server. The Manager automatically detects the type of the following types of variable at the time of import:

- IP address
- Port
- Any



### Note

- If the Manager cannot determine the type of the variable, the variable is listed as **UNDEFINED**.
- You can specify the value of the variable as **any** in which case the variable can be used as an IP address as well as a port number.
- If any of the values of a variable is **any** or **![any]**, the entire value of the variable is considered as **any**.



### Tip

As a best practice if you change the value of any variable, always re-evaluate the rules before saving the attack changes.

### Modify variables

To modify a variable:

- 1 Click on **Remove** icon for the variable.
- 2 Enter the new value in **Value**.
- 3 Click **Add**.
- 4 Click **Save**.

After you click **Save**, the updated value of the variable is saved. However, the updated value of the variable is not reflected in the rules that use the variable. To update the value of the variable in all the rules, you should re-evaluate the rules. To re-evaluate the rules:

- 1 In the **Snort Variables** dialog, click **Actions | Re-Evaluate Rules**.
- 2 Click **Save** to save the updated Snort rules.



### Note

When you re-evaluate the rules, all the rules listed in the **Snort Format** tab are submitted for re-evaluation and not just the failed ones.

### Re-submitting rules with the current variables and classification types

In the Snort Variables dialog, click **Actions | Re-Evaluate Rules**. if you want the Manager to reevaluate all the rules on the **Snort Format** tab to the currently available variables.

This feature is useful if you had:

- Imported the rules before you had defined the variables. If you resubmit the rules now, the ones that failed to evaluate because of invalid variables get successfully evaluated.
- Modified the values of variables

### Restore default variables and classification types

To restore all the listed variables, classification types, and references:

- 1 Click **Actions | Restore Default Variables**.

This restores the details to the Manager client.

- 2 Click **Save**.

This restores the details from the Manager server.

### **Deleting variables and classification types from the Manager**

To delete all the listed variables, classification types, and references from the Manager:

- 1 Click **Actions | Delete All Variables**.

This deletes the details from the Manager client.

- 2 Click **Save**.

This deletes the details from the Manager server.

## Identification of the protocol of a snort custom attack

When you save a custom attack in the Manager server, the Manager categorizes the attack to include it in the applicable rule sets. One of the criterion that the Manager uses to categorize an attack is the application layer protocol that the attack is intended for. For Native Trellix IPS Format Custom Attacks, you can specify this when creating the attack. For Snort Custom Attacks, the Manager identifies it by itself.

Understand how the Manager identifies the impact protocol for a Snort Custom Attack, to ensure if a Snort Custom Attack is published in the policies it is intended for.

How the Manager identifies the impact protocol is explained below:

- 1 First, it tries to use the name of the rules file from which you imported the Snort Custom Attack. For example, if you imported it from ftp.rules, the impact protocol for that attack is identified is FTP. Depending on other criteria for classification, the Manager includes the attack definition in the Rule Sets that include FTP.

If the rules file name starts with web, for example web-attacks.rules or web-client.rules, then the identified protocol is HTTP.

As a best practice, do not specify the destination port number in the rules if you import rules from files named after the protocol. If you do, the Sensor restricts its search to those port numbers.

- 2 If the Manager is unable to identify the protocol by the first method, it identifies the protocol based on the destination port number in the rule. This assumes that the destination port number is a standard port number. For example, if you import a rule from a generically named rules file but the destination port is 80, the identified protocol is HTTP.

The Manager uses this method when you import a rule from a generically named rules file (for example, myrules.rules or when you create the rule in the Custom Attack Editor.

As a best practice, specify the destination port number (standard or otherwise) if you import the rules from a generically named file. This can improve Sensor performance because even if the protocol is not identified, the Sensor looks for the pattern only for the matching destination protocol.

See also

[Create a snort custom attack — an example on page 1658](#)

## How to use snort rules to detect IP communication between specific hosts

This section explains how to construct Snort rules that can detect TCP, UDP, or ICMP communication between specific hosts or networks based on IP address.

### Note

To detect TCP, UDP and ICMP communication between a set of IP addresses, you must create 3 separate rules.

#### TCP communication

To detect TCP communication between hosts specify `ip_proto:tcp;` and `flags:S` as the attack parameters.

##### Example:

```
alert tcp 10.1.1.10 any -> [192.168.230.0/24, 192.168.231.0/24] any (msg:"Malicious TCP Traffic";ip_proto:tcp;flags:S;sid:3121;)
```

Here 10.1.1.10 is an example source IP and [192.168.230.0/24, 192.168.231.0/24] are the example destination subnets.

If you do not specify `flag:S` the rule is triggered for all the TCP packets between the specified hosts.

#### UDP communication

To detect UDP communication between hosts, specify `ip_proto:udp;` as the attack parameter.

##### Example:

```
alert udp 10.1.1.10 any -> [192.168.230.0/24, 192.168.231.0/24] any (msg:"Malicious UDP Traffic";ip_proto:udp;sid:3122;)
```

Here 10.1.1.10 is an example source IP and [192.168.230.0/24, 192.168.231.0/24] are the example destination subnets. The Sensor sends alerts for any UDP packet detected between the specified hosts.

#### ICMP communication

To detect ICMP communication between hosts, specify `ip_proto:icmp;` as the attack parameter.

##### Example:

```
alert ip 10.1.1.10 any -> [192.168.230.0/24, 192.168.231.0/24] any (msg:"Malicious ICMP Traffic";ip_proto:icmp;sid:3123;)
```

Here 10.1.1.10 is an example source IP and [192.168.230.0/24, 192.168.231.0/24] are the example destination subnets.

In the rule, use the following parameters along with `icmp` for a more specific detection:

**itype** - Use this to check for a specific ICMP type value

Syntax: `itype: [<|>]<number> [<><number>];`

**icode** - Use this to check for a specific ICMP code value.

Syntax: `icode: [<|>]<number> [<><number>];`

##### Example:

```
alert icmp 10.1.1.10 any -> [192.168.230.0/24, 192.168.231.0/24] any (msg:"example for itype and icode"; itype 8; icode:0; sid:2022; priority:3;)
```

Here 10.1.1.10 is an example source IP and [192.168.230.0/24, 192.168.231.0/24] are the example destination subnets. This rule is triggered for inbound traffic where the type is 8 and code is 0.

Write snort custom attacks

Before you begin

Before you begin to construct the Snort rules, review the following:

- In the rule, you can use only the variables, classtypes, and references that are available in the Manager. For information on how to view the available values, see Viewing the variables and classification types.
- You can write only one rule at a time.

You can construct Snort rules directly in the Manager using the Custom Attack Editor. Note that these rules must conform to the Snort rules language syntax. Structure of a Snort rule provides information how to construct Snort rules within Trellix IPS.

To construct Snort rules in the Manager:

Task

- 1 Select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click **Custom Attacks**.  
The Custom Attack Editor opens with the existing Custom Attacks listed on the **Native Trellix IPS Format** tab.
- 2 Click **Snort Format** tab.
- 3 Click **+**.

The **New Snort Rule** interface opens.

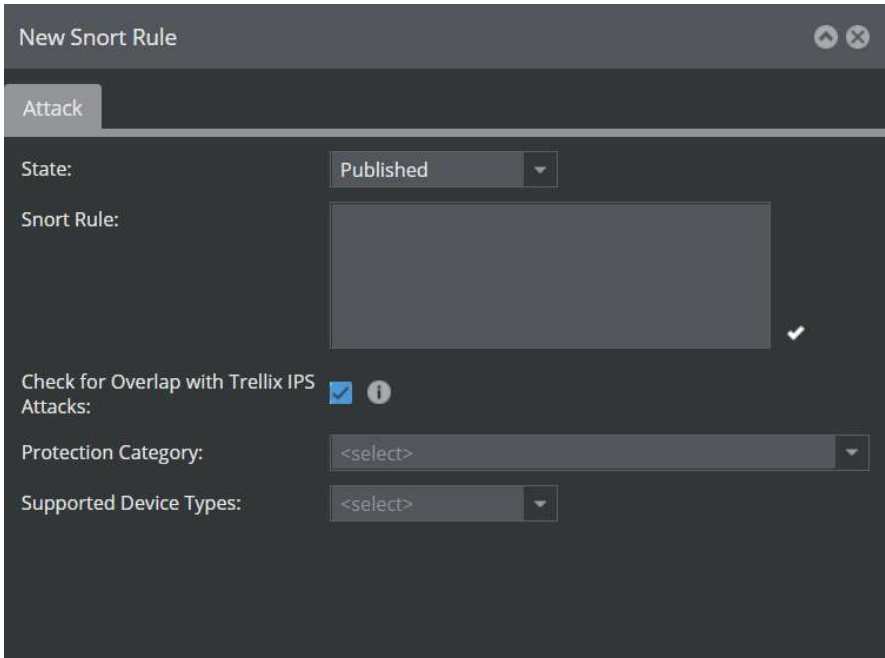


Figure 4-32 New Snort Rule window

- 4 Construct the Snort rule in the **Add Snort Rule** dialog.

Note the following:

- You cannot define variables in the **Add Snort Rule** dialog, but you can use the variables that are available in the database.
  - You can write only one rule at a time.
  - If you are using the keywords such as classtype or reference, make sure the corresponding values are already defined and available in the database. For example, to import classifications, you can import a file that calls the classification config file and then use these classifications in the rules that you construct.
- 5 Select **Check for Overlap with Trellix IPS Attacks** to verify if the rule matches an existing Trellix IPS attack definition. If the rule matches then the Snort Rule will be **Staged**.
  - 6 Select the most appropriate **Protection Category** for the attack.
  - 7 Click **Add**.  
The rule is listed in the **All Custom Attacks** tab.
  - 8 Save the rule to the database so that it gets published in the relevant policies.

See also

[Saving the Snort custom attacks on page 1602](#)

### Saving the Snort custom attacks

After you write or import the rules, you need to save them to the Manager database. Then, the rules for which the State is **Published**, are automatically added to the various exploit policies (both for inbound and outbound). This is similar to how the Trellix IPS custom rules are **Published** in the policies. Review the following points to understand how the rules are published in the policies.

The newly added or modified attacks are automatically compiled at the time of saving. The attacks that fail compilation are set to:

- **Failed** in the **Test Compile** column
- **Staged** in the **State** column

Only rules that comply with the following conditions qualify to be published in the exploit policies of Trellix IPS:

- The Snort rule has been converted to Trellix IPS format successfully or with warnings.
- The State is set to **Published**.

All the qualified rules are published in the **Default Prevention** policy. In addition, these rules are also published in other policies that meet the following criteria:

- Benign Trigger Probability of medium. By default, all the imported Snort rules are assigned a Benign Trigger Probability value of medium which cannot be modified.
- Severity of the attack. This depends on the default classification based on the classtype of a rule or the priority tag in the rule.
- Protocol

See also

[Write snort custom attacks on page 1601](#)



### Save the qualified rules

Before you save the Snort Custom Attacks to the database, you can also specify the Sensor type. For example, if you choose NS-series as the Sensor type for an attack, then this attack definition is relevant only to the NS-series Sensors of the corresponding admin domain. So only the NS-series Sensors inspect the traffic for this particular attack definition. Even when you apply the same policy to a Virtual Sensor, it does not check for this attack.

#### Task

- 1 Go to the corresponding tab in the Custom Attack Editor.
- 2 Verify that the rules that you want to be published in the rulesets are in the **Published** state. If not, you can click on the attack and select **Published**.
- 3 Verify the Target Device to which the Snort Custom Attack should be applied to.  
For example, if you select NS-series, only the NS-series Sensors in the current admin domain inspect traffic for this attack.
- 4 Click **Save**.  
The qualified rules are saved in the database and the corresponding policies are updated with these attacks. You can view the progress in the bottom part of the Custom Attack Editor.
- 5 You need to update the Sensors for the saved attacks to be detected.  
You can also verify if the attacks are published in the policies.
  - 1 From the Resource Tree, select **Policy | <Admin Domain Name> | Intrusion Prevention | Policy Types | IPS**.
  - 2 Double-click on one of the All Inclusive policies; for example, you can open the **Default Testing** policy.
  - 3 Click **Attack Definitions** tab.
  - 4 Sort the attacks based on Name and verify if the Snort rules have been published.

### Customizing the snort rules attack responses

You can use the Policy Editor to customize the Snort attacks like you would customize any other Trellix IPS attack. If you make any configuration changes to an attack, you should update the Sensor of the changes for them to take effect.

#### Note

By default, when you save a Snort attack to the database, Enable Attack and Enable Alert are enabled. All other configurations including blocking and packet logging are disabled.

### Delete the snort rules from the database

Whenever required, you can delete the snort rules from the database and stop the Sensors from sending alerts:

#### Task

- 1 Open the Custom Attack Editor.
- 2 All the custom rules in the database are listed.
- 3 Select the required rules, click and select .

- 4 Click **Save** to save the changes and update the policies.
- 5 Update the Sensors of the configuration changes.

## Common tasks

This section contains information on tasks that are common to both Trellix IPS as well as Snort Custom Attacks.

### How attacks are published in policies

After you create a Custom Attack and the constituent signatures or rules, you need to save the attack in the Manager server for it to be published in every rule set where the attack definition fits one or more categories, rule set categorization that was configured as part of attack creation. When you apply a policy that includes a rule set containing your attack definition, detection of your attack is active.

Consider that when you created a Trellix IPS Custom Attack, you set the Severity to Medium and chose HTTP as the Exploit classification. Once exported, this attack is published in all rule sets that publish Medium-severity, HTTP attacks, such as the Default, Outside Firewall, and Web Server rule sets provided with Trellix IPS. (This also publishes any rule set that you have created which calls for HTTP attacks of Medium severity or higher.) When you apply a policy that publishes one of these rule sets, you are applying your Custom Attack for active searching and alerting upon detection.

When you save the custom attacks to the Manager server, only those attacks with State as **Published** are published in the rule sets. If you want to change the State of a custom attack, right click on it and select **Published** or **Staged**.

When you create a Trellix IPS Custom Attack, the State is set to **Published** by default. In case of Snort Custom Attacks, there are instances where the state is set to **Staged**. For example, if there is a Trellix IPS attack signature with the same CVE ID, the Manager sets the State to staged when you save the attack. The State is also set to staged if the Conversion Result is failed or warning.

#### Note

If after saving the Custom Attack to the Manager server, you import the attack back to the Custom Attack Editor, edit the file and make a name change to either the attack or a signature in the attack, then save it again in the Manager server, the name change will not take affect in any open Attack Log views. You must close all Attack Log windows and restart the Attack Log to see the name change upon attack detection.

### Viewing a policy to verify inclusion of the attack

After you save the custom attacks in the Manager server, verify if the attacks in **Published** State have actually been published in the policies. The attacks are categorized based on the following and then published in the corresponding rule sets.

When you have finished creating a custom attack, you need to save it. Saving the attack in the Manager server, publishes your attack definition in every rule set where your attack fits one or more categories, provided the State of the attack is **Published**. When you apply a policy that publishes a rule set containing your attack, detection of your attack is active.

The Manager categorizes attacks based on:

- Impact application
- Impact operating system
- Impact application layer protocol
- Severity of the attack
- Benign Trigger Probability (BTP)

For example, when you created a Trellix IPS Custom Attack, you set the Severity to Medium and chose HTTP as the Exploit classification. Once saved in the Manager server, the attack is published in all rule sets that include Medium-severity, HTTP attacks, such as the **Default**, **Outside Firewall**, and **Web Server** rule sets provided with Trellix IPS. (This also publishes any rule set that you have created which calls for HTTP attacks of Medium severity or higher.) When you apply a policy that publishes one of these rule sets to a port or VIDS, you are publishing your attack for IPS.

### Note

If you make a name change to either the attack or a signature within the attack, then save it back into the Manager, the name change will not take effect in any open Attack Log views. You must close all Attack Log windows and restart the Attack Log to see the name change upon attack detection.

### See also

[Create custom attacks on page 1547](#)

[Create a signature on page 1559](#)

[Import previously exported custom attacks on page 1531](#)

[Update the Sensor configuration to apply a policy on page 1611](#)

[Create a snort custom attack — an example on page 1658](#)

[Add the condition on page 1656](#)

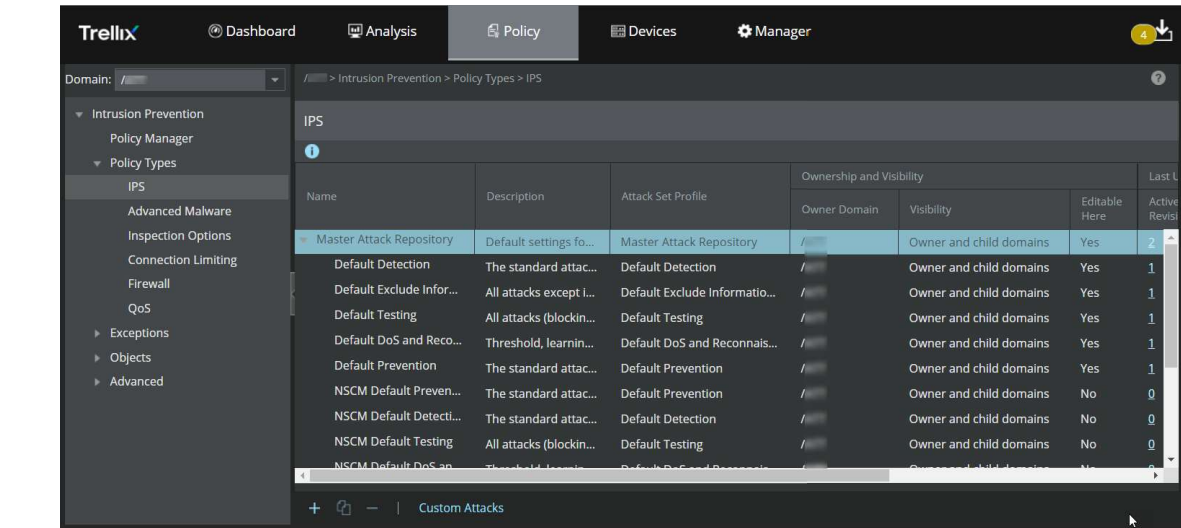
[Add the second condition with OR comparisons on page 1666](#)

## Verify the inclusion of custom attack in IPS policies

To verify if a custom attack has been published in a rule set:

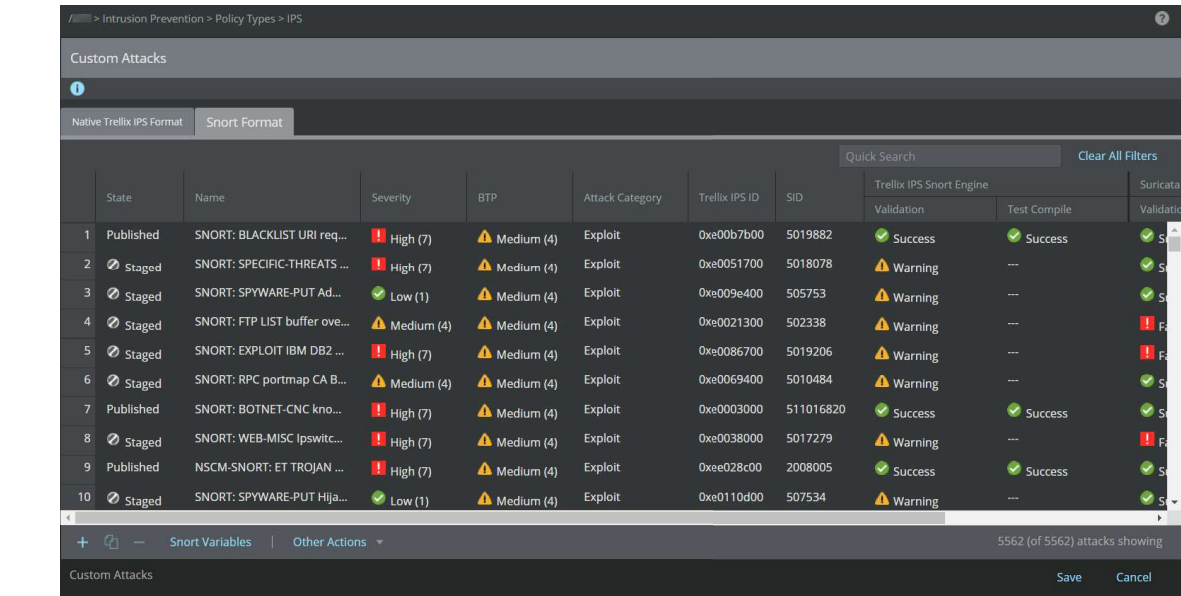
Task

- 1 Select Policy | <Admin Domain Name> | Intrusion Prevention | Policy Types | IPS.
- 2 Double-click the "Default Testing " attack set to view the selected attacks.



- 3 Scroll down the list (sorted alphabetically) to find your attack file.

Trellix IPS Custom Attacks have "UDS-" appended to the beginning of the name, while Snort Custom Attacks have "SNORT-" appended to the beginning.



### Add attack descriptions to the Attack Encyclopedia

You can enter Attack Descriptions and comments in the Attack Encyclopedia for custom attacks. Attack Description can be added at the time of creating a custom attack. Comments can be added after creating the custom attack. When an alert is raised for a custom attack, these details can be viewed from the Attack Log as well

This procedure on adding comments assumes that you have a Trellix IPS or a Snort Custom Attack published.

#### Task

- 1 Select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**.

The **IPS** page is displayed. The **Default Testing** policy listed in the **IPS** page contains all the custom attacks (both Trellix IPS and Snort) that have been published.

- 2 Double-click on the **Default Testing** policy.

All the Attacks Definitions under the **Default Testing** policy are listed under the **Attacks Definitions** tab.

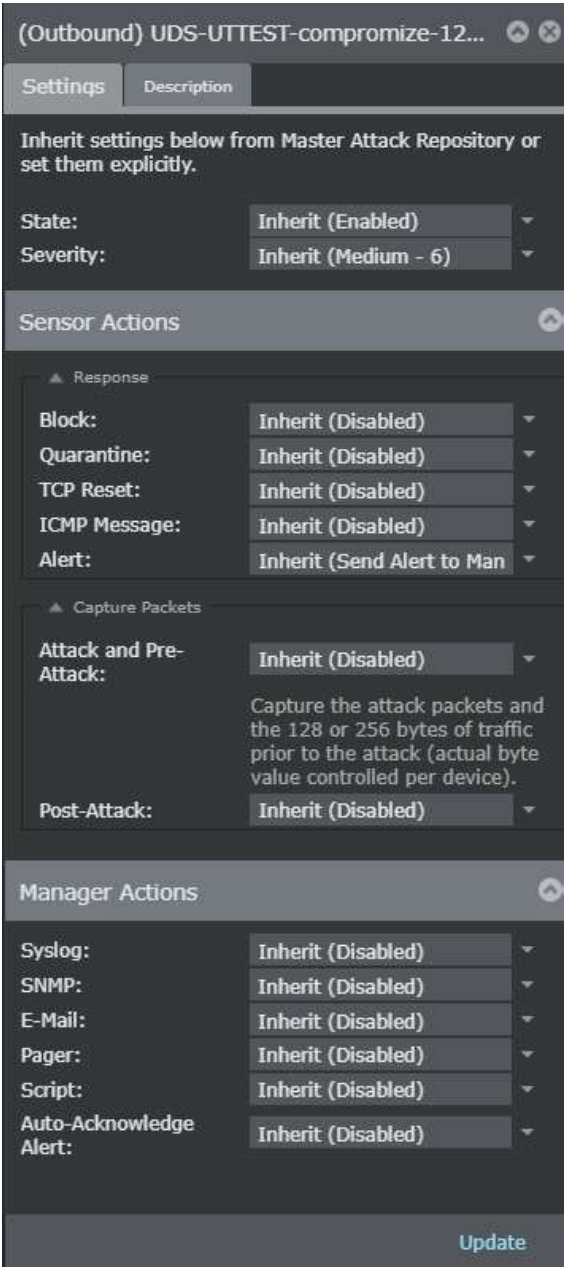
- 3 Search for the custom attack for which you want to add attack descriptions by entering the name of the attack in the **Quick Search** box



#### Note

All Trellix IPS custom attacks are prefixed "UDS" and all Snort attacks are prefixed "Snort".

- 4 Double-click on the custom attack that you have located.  
The attack details pane for the selected attack is displayed.



5 Click **Description** to view the details of the custom attack from the attack encyclopedia.

(Outbound) AbsoluteFTP: AbsoluteFTP LIST Co...

Settings

Description

BTP:

Medium (3)

RfSB:

No

Protection Category:

Server Protection/File Servers

Target:

Client

HTTP Response Attack:

No

Priority:

Low

Protocols:

ipv4

Attack Category:

Exploit

Attack Subcategory:

code-execution

Snort Engine:

---

Version Added:

9.8.4.6

Last Updated:

10.8.26.3

Reference

Trellix IPS ID:

0x45d29200

CVE ID:

[CVE-2011-5164](#)

Microsoft ID:

---

Bugtraq ID:

50614

CERTID:

---

ArachNIDS ID:

---

Additional Information:

<http://secunia.com/advisories/46781>

Signatures

Comments

Show Comments from Parent Admin Domains:

☒

Comment for this Admin Domain:

Maximum 1024 characters

Make Comment Available to Child Admin Domains:

☐

Save Comment

6 Enter the comment for the Attack in the **Comment for this Admin Domain** field.

7 Click **Save Comment**.

### Compile the attack definitions

In the policies, you can use the attack definitions from the following types:

- Trellix IPS-supplied Attack definitions (signature set)
- Trellix IPS Custom Attack
- Snort Custom Attack

There can be instances where you may want to publish only specific types of attack definitions for a specific Sensor. For example, you may want to configure a Sensor to monitor traffic for specific attacks. You can also use this feature to troubleshoot and isolate the attack definitions that may cause an adverse effect on a Sensor's performance.

To specify the attack definition type for a Sensor:

#### Task

**1** Select **Devices** | **<Admin domain>** | **Devices** | **<Device Name>** | **Setup** | **Attack Compilation**.

**2** Select the attack definition type:

- **Signature Set Attacks** - These are the attacks from Trellix IPS signature set.

When the **Signature Set Attacks** option is selected, the Manager allows you to choose **Signature Set Attack Priorities** for the Sensor. This allows the Manager to dynamically compile only critical attacks from the standard signature set for Sensors that do not have enough resources to support all attacks.

The signature set attack priorities available are as follows:

- **All**: Includes all attack definitions in the signature set. This is the default signature set attack priority selected for NS-series and Virtual IPS Sensors and provides complete attack coverage.
- **High and Medium only**: It comprises of high and medium priority attacks in the signature set. This option provides partial attack coverage.
- **High only**: It comprises of high priority signature set attacks. You can use this option to optimize Sensor resources on Sensor models running older Sensor software versions to support the latest signatures against most critical attacks.



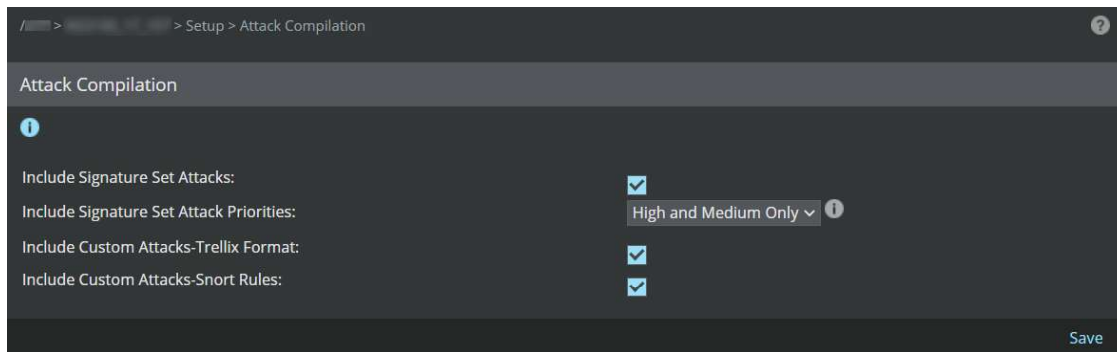
The **High Only** signature set attack priority provides an attack coverage only against the most critical attacks.



- **Custom Attacks - Trellix IPS Format:** Select this if you want to use the definitions that you created in the Trellix IPS format. This also includes the Trellix IPS-supplied custom attacks (emergency UDS).
- **Custom Attacks-Imported Snort Rules:** Select this if you want to use the Snort Custom Attack definitions that you created or imported.

### 3 Click **Save**.

When you update a Sensor with the configuration changes, only the attack definitions from the type specified here are pushed to the Sensor.



## Update the Sensor configuration to apply a policy

Once you have verified that a custom attack is published in a rule set, you need to update the Sensors of the changes for the Sensors to look for traffic matching the attack definition. Since addition of a custom attack affects one or more rule sets, which in turn affects one or more policies, your policies may require updating across all of your Sensors in order for effective detection of your attack definition.

The **Device Configuration Changes** action sends configuration changes, signature updates, and policy changes to all of the devices under the **Devices** node.

### Note

The **Device Configuration Changes** action updates multiple Sensors, but only transmits the update to one device at a time.

To update the configurations of multiple devices:

### Task

- 1 Select **Devices** | <Admin Domain Name> | **Global** | **Device Manager**.

The **Device Manager** page is displayed.

- 2 Select the **Sensors** tab. Then, select the Sensor(s) to be updated from the list.

### Note

You can also do it at the device level by selecting **Devices** | <Admin Domain Name> | **Devices** | <Device Name> | **Deploy Pending Changes**.

- 3 Click **Sync**. The **Bulk Sync** window is displayed. Select the required configurations and click **Sync**.

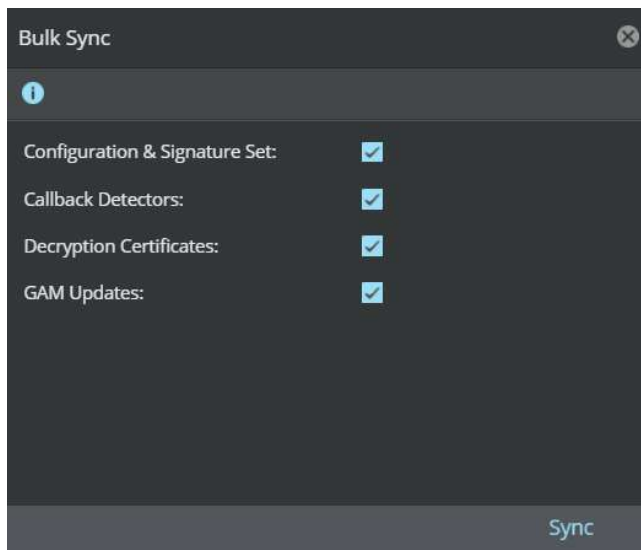


Figure 4-33 Bulk Sync

A Snort Custom Attack definition being applied to a Sensor port depends on the following:

- The attack definition should be published in the corresponding policy.
- The attack definition type that you have specified.
- You should have specified the corresponding Target Device for the attack definition. That is, if it is an NS-series Sensor port, you should have selected NS-series as the Target Device for the attack definition.

See also

*Viewing a policy to verify inclusion of the attack on page 1604*

## Custom attacks export

You can export all the Trellix IPS Custom Attacks, Snort Custom Attacks, and custom-defined grep protocols in the Manager to a ZIP file. The export feature enables you to use the custom attacks on a different Manager without having to recreate them. Trellix strongly recommends that you do not try to modify the exported attack files and then re-import them.

In the exported ZIP file:

- The Trellix IPS Custom Attacks are stored in the attacks.xml file.
- The packet grep protocols are stored in the pktgrepprotocol.xml file.
- The Snort Custom Attacks are exported in the Snort format.
- The Snort Custom Attacks Rules and the related data such as variables that you imported are stored in correspondingly named files. For example, assume that you had imported rules from ftp.rules file. When you export, these rules are stored in ftp.rules file within the ZIP.
- The rules that you directly created in the Editor are contained in unknown.rules file.



### Note

There is no option to export just the Trellix IPS Custom Attacks or just the Snort Custom Attacks. When you export, all custom attacks listed in the All Custom Attacks tab and the custom-defined packet grep protocols are exported.

### Export the custom attacks

To export the custom attacks to your client or other reachable location, do the following:

#### Task

- 1 In the Custom Attack Editor, click **Native Trellix IPS Format** tab and select **Other Actions | Export** to export Native Trellix IPS Format custom attacks.  
Click **Snort Format** tab, and then select **Other Actions | Export** to export Snort Format custom attacks.
- 2 Select the folder where you want to save the ZIP file.  
The ZIP file is downloaded automatically to your set download folder, if your system is configured for automatic download to a specific folder.
- 3 Type a name for the zip file. Do not add an extension.
- 4 Click **Save**.

### Examples

This section contains examples that describe how to create various types of custom attack definitions and import Snort rules for the Suricata Snort Engine.

### Use case scenarios

This section describes the following user scenarios:

- Select the Suricata Snort engine and import the Snort rules.
- Create an attack by using String Pattern Match to detect a specific string in the HTTP GET requests.
- Create an attack by using Numeric Pattern Match to detect a specific numeric value in the FTP protocol.
- Create an attack by using Numeric Range Match to detect if the ports are in the specified range in the signature for HTTP protocol.
- Create an attack by using Numeric Enumeration Match to detect if the response code matches the ones specified in the signature for HTTP protocol.
- Create an attack by using Single Fixed Field Match to detect if the IP address and the protocol matches the ones specified in the signature.
- Create an attack by using Packet Grep Protocol to detect if the packets contain a specific string sequence.

To open the Custom Attack Editor, select **Policy | <Admin Domain Name> | Intrusion Prevention | Policy Types | IPS**. Click **Custom Attacks**.

### Importing Snort rules for the Suricata Snort engine

This sample user scenario describes the steps for the following activities:

- Selecting the Suricata Snort engine
- Rebooting the Sensor
- Importing Snort variables and rules from a file

- Viewing the status of the imported Snort rules
- Viewing the imported Snort variables
- Pushing changes from the Manager to the Sensor

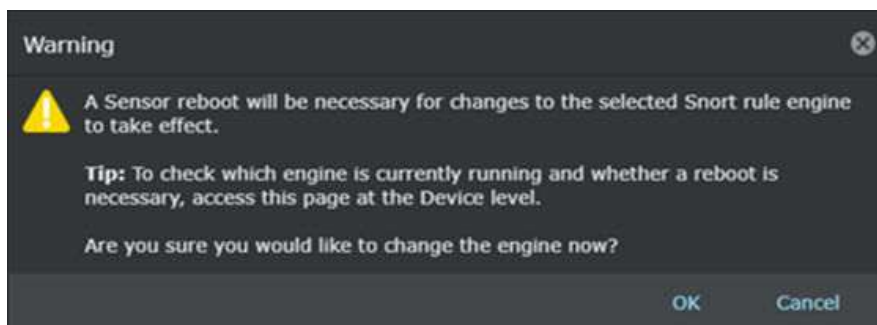
### Note

You can configure the Snort engine at a global level but can override the setting per Sensor and select a different engine at a Sensor level. In this scenario, you will be configuring the Snort engine at a global level.

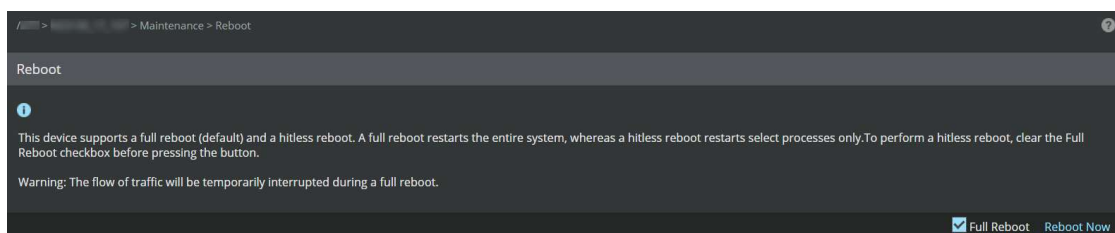
## Importing Snort rules for the Suricata Snort engine

### Task

- 1 Go to **Devices** | **<Admin Domain Name>** | **Global** | **IPS Device Settings** | **Advanced Device Settings**. The default engine selected is **Trellix IPS Snort**.
- 2 To select the snort engine, click the **Snort Rule Engine** drop-down and select the **Suricata Snort**.  
A pop-up appears asking you to confirm your changes and informing you that a reboot will be necessary for the change to take effect.



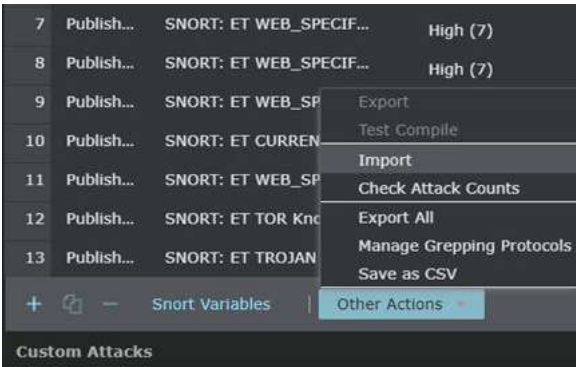
- 3 Click **OK** and then click **Save** at the bottom of the page.
- 4 Reboot the Sensors which require this change. Go to **Devices** | **<Admin Domain Name>** | **Devices**. Select the device to reboot. Go to **Maintenance** | **Reboot**. Click **Reboot Now**



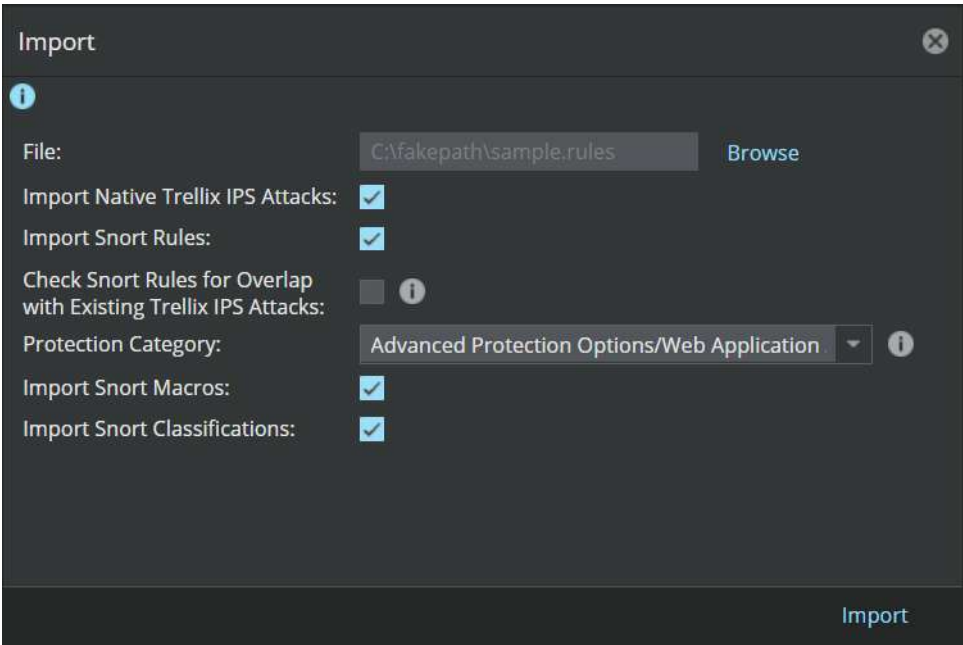
After the reboot, the **Advanced Device Settings** page of the Sensor displays **Suricata Snort**.

- 5 Go to **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click the **Custom Attacks** button (bottom left corner).

6 In the Custom Attack Editor, from the Snort Format tab, click Other Actions | Import



7 Browse to the location of the rules file to be imported and select the rules file.



8 Select a Protection Category value.

The Protection Category indicates the intent of the attack and the intended target. The list of Protection Categories is pre-defined and is provided by Trellix Advance Research Center. In this scenario, the selected value is **Advanced Protection Option/File Reputation**

9 Click **Import**

After the import, the **Import Results** screen is displayed.

Import Results

Trellix

Snort

|                         | Validation Engine |                |
|-------------------------|-------------------|----------------|
|                         | Trellix IPS Snort | Suricata Snort |
| ✔ Successfully Imported | 20                | 41             |
| ⚠ Imported with Warning | 38                | 0              |
| ❌ Failed Import         | 0                 | 17             |

Variables Updated: Yes

Notes:

1. Rules imported with one or more warning are left in a staged state by default (and are therefore excluded from policies) but can be changed to a published state manually.

2. Rules that fail import are staged and cannot be published until their syntax has been corrected.

3. Imported Snort variables can be managed using the Snort Variables button.

Check Attack Counts

10 Close the **Import Results** screen.

11 View the imported Snort rules.

Know which rules converted successfully, which converted with warnings, and which failed to convert. For more information, see [Managing Snort rules](#) on page 1539.

12 View the imported Snort variables.

Know which variables were imported and their classification type. For more information, see [Viewing the Snort variables](#) on page 1596.

**13** Verify if the attacks in **Published** state are actually published in the policies.

- a** From the Resource Tree, select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**.
- b** Double-click one of the All Inclusive policies; for example, you can open the **Default Testing policy**.
- c** Click **Attack Definitions** tab.
- d** Sort the attacks based on Name and verify if the Snort rules are published.

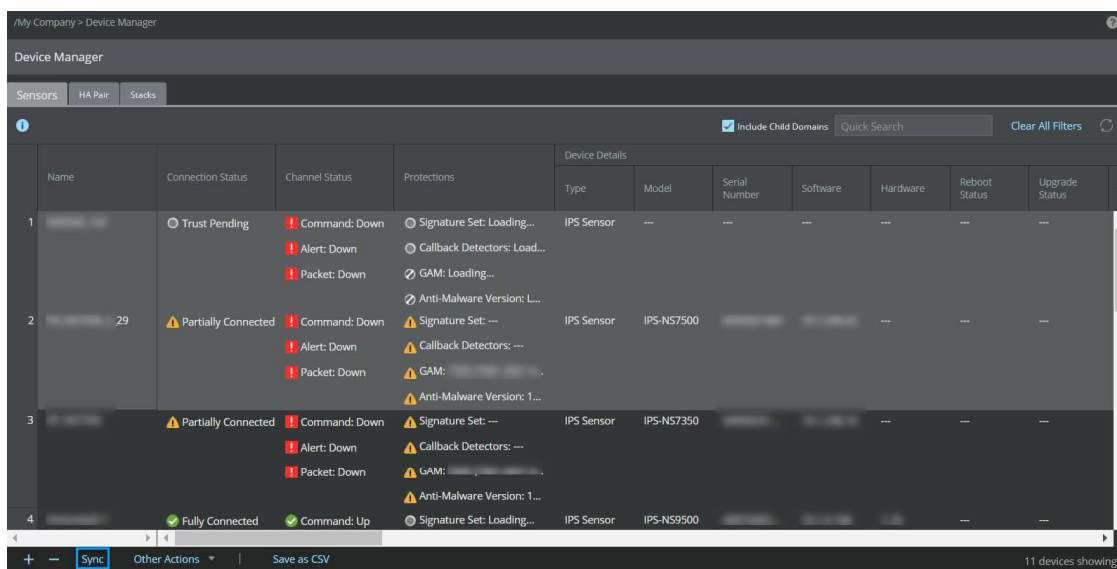


#### Note

You can also create your own attack set profiles to define the exact environment resources you want to protect. For more information, see the *Trellix Intrusion Prevention System Product Guide*.

**14** Go to **Devices** | **<Admin Domain Name>** | **Global** | **Device Manager**.

**15** Select the devices on which you want to deploy the imported rules by selecting the check boxes. Click **Sync**.



The **Bulk Sync** window is displayed.

**16** Select **Configuration & Signature Set** from the window, and click **Sync**.

This deploys the IPS policies that contain the new Snort rules to the selected Sensors.

#### Sample Snort rules file

The following is a sample Snort rules file.

```
var HTTP_PORTS [80,443,8080]
var HOME_NET [10.0.0.0/8,192.168.0.0/16,172.16.0.0/12,169.254.0.0/16,1.0.0.0/8,2.0.0.0/8]
var EXTERNAL_NET !
[ $HOME_NET,198.80.42.0/23,198.217.128.0/17,198.241.128.0/17,64.94.118.0/23,66.185.176.0/20]

alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET TROJAN TeleBots BCS-server CnC Beacon_NIDS779"; flow:established,to_server; urilen:1; content:"POST"; http_method; content:"value="; depth:6; http_client_body; fast_pattern; content:!"Content-Type|3a|"; http_header; content:!"Referer|3a|";
```

```
http_header; reference:url,www.welivesecurity.com/2016/12/13/
rise-telebots-analyzing-disruptive-killdisk-attacks/; classtype:trojan-activity; sid:2023652; rev:1;)
```

### Note

- The variables and rules can be in separate files. If you are using separate files for variables and rules, ensure that you load the variables before loading the rules.

## Create an attack definition for string pattern match

This scenario is about creating a Custom Attack Definition that uses string pattern match to detect any HTTP GET requests in the request URL for a specific string.

In order to write a proper Native Trellix IPS Format Custom Attack for this example, you must identify several key elements:

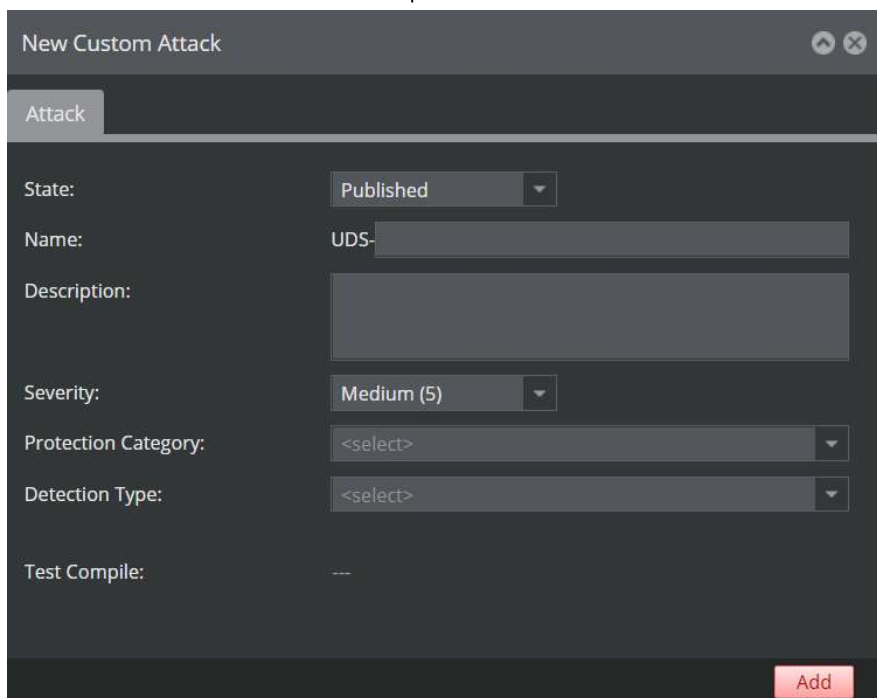
- Application protocol: HTTP
- Request method: get
- Where to look: URL
- String to match: LpEhZWWuLc0AwAAeHB3DAAAABA
- Detection window: request

To create the attack definition for this example:

### Task

- 1 In the Custom Attack Editor, click **+**.

The **New Custom Attack** interface opens.



- 2 Select one of the options from the **State** drop-down menu.



- 3 You can specify a name such as " Custom Exploit: String Pattern Match" as the **Name**.  
The letters "UDS" (user-defined signature) are appended to the front of the name upon completion; thus, this attack appears as "UDS-Custom Exploit: String Pattern Match" in the Custom Attack Editor, as well as the attack database when you save in the Manager server.
- 4 Type a description for your attack. This area can be used for your notes or other specific information pertinent to your new attack.
- 5 You can select **Medium (5)** as the **Severity** because this example scenario does not necessarily involve malicious activity.
- 6 Select **Advanced Protection Options/Web Application Server** as the **Protection Category**.
- 7 Select **Custom Exploit (Signature-Based)** from the **Detection Type** drop-down menu for the attack.
- 8 Click **Next**.

The screenshot shows a 'New Custom Attack' window with the following fields and values:

- State:** Published
- Name:** UDS- Custom Exploit: String Pattern Match
- Description:** Custom Exploit: String Pattern Match
- Severity:** Medium (5)
- Protection Category:** Advanced Protection Options/Web Application Server
- Detection Type:** Custom Exploit (Signature-Based)
- Test Compile:** ---

Below the Detection Type dropdown, the text 'Define an attack and its signature details manually.' is visible. A 'Next >' button is located at the bottom right of the window.

- 9 In the **Matching Criteria** section, select **Protocol** as the **Criterion** and select **http** as the protocol.

When you subsequently save this attack definition in the Manager server, it will be part of every rule set which includes Medium-severity, HTTP attacks (must match both severity and protocol).

**New Custom Attack**

Attack Signature-1663264802421 +

Protection Category: Advanced Protection Options/Web Application Ser

Attack Target: Server

Blocking (As Applicable): Attack Packet Only

Benign Trigger Probability (BTP): Medium (5)

Attack Category: Exploit

Test Compile: ---

Trellix IPS ID: ---

Supported Device Types: Any

Last Updated: ---

**Matching Criteria**

Criterion: Protocol

Protocol: http Add

| Protocol   | Software Package (OS) |
|------------|-----------------------|
| No Matches |                       |

Add

- 10 Attack details configuration is complete. Continue to create signature.

### Create the signature for string pattern match

After you create the attack definition, you create the signature for the attack.

## Task

- 1 In the New Custom Attack window, click **Signature-<signature name>** tab.

New Custom Attack

Attack Signature-1663264802421 +

Name: Signature-1663264802421

Benign Trigger Probability (BTP): Medium (3)

Target Host Architecture: Any

Detection Window: Single Packet

Supported Device Types: Any

Signature Details

Use the buttons below to add signature details. Each signature requires:

1. At least one condition.
2. At least one AND or OR comparison within each condition.

Condition 1

+ - | AND OR X

Add

Figure 4-34 New Signature window

- 2 (Optional) Clear the **Name** and type a new name for your signature.
- 3 For this example, you can leave the **Benign Trigger Probability (BTP)**, **Target Host Architecture**, and **Detection Window** with the default values.
- 4 Based on the Sensor model that you plan to use for this example, select the **Supported Device Types**.
- 5 Add the condition to the signature.  
It is in the conditions that you specify the following details:
  - The string that the Sensor should look for
  - Which section of the HTTP request should it look for the string
- 6 Proceed to add the condition.

Add the condition for string pattern match

To add the condition to the example signature:

Task

- 1 Condition 1 is added automatically according the URL you provided.
  - 2 Click Condition 1 so that it is highlighted.
  - 3 Click AND in the Comparisons section.
- The Add AND Comparison dialog opens.

Add AND Comparison

Comparison Type:

String Pattern Match

Protocol Details

Protocol:

http

HyperText Transfer Protocol

Protocol Field:

req-uri

This field matches the entire URI of a HTTP request( with request method as selected from below field'http-req-method').

http-req-method:

get

Regex Details

Operator:

Equals

Text to Match:

LpEhZWWuLc0AwAAeHB3DAAAABA

Ignore Case:

☒

Ignore String Position:

☒

Save

Figure 4-35 String Pattern Match

- 4 For this example, select String Pattern Match in the Comparison Type drop-down menu.
- 5 Select http from the Protocol drop-down menu.  
Because you selected HTTP, the Custom Attack Editor displays the HTTP-specific protocol fields on the following screen.
- 6 Configure the fields for the comparison you have chosen.  
For this example, specify req-uri for the Protocol Field. This specifies that the Sensor should search in the URI of the request packet.
- 7 Select get as the http request method.
- 8 From the drop-down list in the Regular Expression Operator section, select the matching criteria as Equals which means that the comparison criteria must be equal to the regular expression entered.

- 9 Type the pattern to match in the **Text to Match** text box.  
For this example, the pattern to match is "LpEhZWWuLc0AwAAeHB3DAAAABA".
- 10 Click ☒ to verify that your expression is a valid string, and all required options are represented. Click **OK** to close the validation message window.
- 11 (Optional) Deselect **Ignore Case** and **Ignore String Position** check boxes.
- 12 Click **Save**.  
Your comparison appears under **Condition 1**.
- 13 Click **Add** in the **New Custom Attack** window.
- 14 Verify that the attack definition is listed on the **Native Trellix IPS Format** tab.

Custom Attacks

Native Trellix IPS Format

Snort Format

Quick Search

Clear All Filters

|   | State     | Name                                     | Severity     | BTP          | Attack Category | Test Compile | Trellix IPS ID | Last Update Time  |
|---|-----------|------------------------------------------|--------------|--------------|-----------------|--------------|----------------|-------------------|
| 1 | Published | UDS-Custom Exploit: String Pattern Match | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ✔ Success    | 0xc0000400     | Sep 16, 2022 0:02 |

+ 🔍 -

Other Actions

1 (of 1) attacks showing

- 15 Click **Save** to save the Trellix IPS Custom Attack in the Manager server database.
- 16 Make sure the Trellix IPS Custom Attack is saved in the database and also published in the policies.

Create an attack definition for numeric value match

This scenario is about creating a Custom Attack Definition that uses numeric value match to detect if the command line parameter length in make directory operation is greater than 50.

In order to write a proper Native Trellix IPS Format Custom Attack for this example, you must identify several key elements:

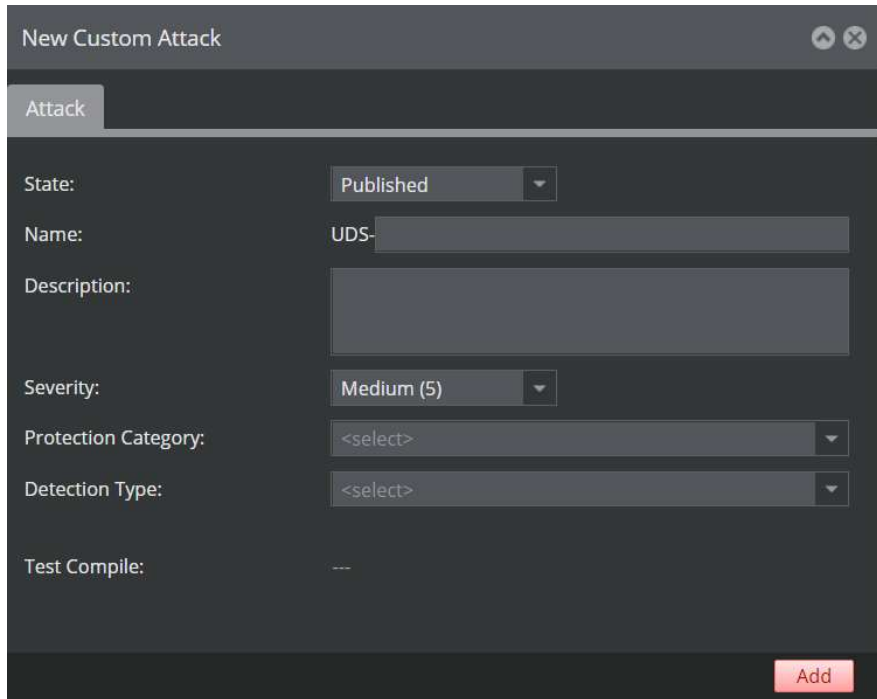
- Application protocol: FTP
- Protocol field: cmd-param-length
- Cmd qualifier: mkd
- Value to match: 50

To create the attack definition for this example:

#### Task

- 1 In the Custom Attack Editor, click **+**.

The **New Custom Attack** interface opens.



- 2 Select one of the options from the **State** drop-down menu.
- 3 You can specify a name such as "Custom Exploit: Numeric Value Match" as the **Name**.  
The letters "UDS" (user-defined signature) are appended to the front of the name upon completion; thus, this attack appears as "UDS-Custom Exploit: Numeric Value Match" in the Custom Attack Editor, as well as the attack database when you save in the Manager server.
- 4 Type a description for your attack. This area can be used for your notes or other specific information pertinent to your new attack.
- 5 You can select **Medium (5)** as the **Severity** because this example scenario does not necessarily involve malicious activity.
- 6 Select **Advanced Protection Options/Web Application Server** as the **Protection Category**.
- 7 Select **Custom Exploit (Signature Based)** from the **Detection Type** drop-down menu for the attack.

8 Click **Next**.

New Custom Attack

Attack

State:

Published

Name:

UDS- Custom Exploit: Numeric Value Match

Description:

Custom Exploit: Numeric Value Match

Severity:

Medium (5)

Protection Category:

Advanced Protection Options/Web Application Server

Detection Type:

Custom Exploit (Signature-Based)

Define an attack and its signature details manually.

Test Compile:

---

Next >

- 9 Select **Client or Server** as Attack Target.
- 10 Select **Attack Packet Only** as **Blocking (As Applicable)**.

11 In the **Matching Criteria** section, select **Protocol** as the **Criterion** and select **HTTP** as protocol.

When you subsequently save this attack definition in the Manager server, it will be part of every rule set which includes Medium-severity, HTTP attacks (must match both severity and protocol).

New Custom Attack

Attack

Signature-1663267348199

+

Severity:

Medium (5)

Protection Category:

Advanced Protection Options/Web Application Ser

Attack Target:

Client or Server

Blocking (As Applicable):

Attack Packet Only

Benign Trigger Probability (BTP):

Medium (5)

Attack Category:

Exploit

Test Compile:

---

Trellix IPS ID:

---

Supported Device Types:

Any

Last Updated:

---

Matching Criteria

Criterion:

Protocol

Protocol:

http

Add

Protocol

Software Package (OS)

12 Attack details configuration is complete. Continue to create signature.

Create the signature for numeric value match

After you create the attack definition, you create the signature for the attack.



## Task

- 1 In the New Custom Attack window, click **Signature-<signature name>** tab.

New Custom Attack

Attack Signature-1663267348199 +

Name: Signature-1663267348199

Benign Trigger Probability (BTP): Medium (3)

Target Host Architecture: Any

Detection Window: Single Packet

Supported Device Types: Any

Signature Details

Use the buttons below to add signature details. Each signature requires:

1. At least one condition.
2. At least one AND or OR comparison within each condition.

Condition 1

+ - | AND OR X

Add

Figure 4-36 New Signature window

- 2 (Optional) Clear the **Name** and type a new name for your signature.
- 3 For this example, you can leave the **Benign Trigger Probability (BTP)**, **Target Host Architecture**, and **Detection Window** with the default values.
- 4 Based on the Sensor model that you plan to use for this example, select the **Supported Device Types**.
- 5 Add the condition to the signature.  
It is in the conditions that you specify the following details:
  - The string that the Sensor should look for
  - Which section of the HTTP request should it look for the string
- 6 Proceed to add the condition.

Add the condition for numeric value match

To add the condition to the example signature:

Task

- 1 Click **Condition 1** so that it is highlighted.

New Custom Attack

Attack

Signature-1663267348199

+

Name:

Signature-1663267348199

Benign Trigger Probability (BTP):

Medium (3)

Target Host Architecture:

Any

Detection Window:

Single Packet

Supported Device Types:

Any

Signature Details

Use the buttons below to add signature details. Each signature requires:

1. At least one condition.

2. At least one AND or OR comparison within each condition.

Condition 1

+ - | AND OR X

Add

- 2 Click **AND** in the Comparisons section.

The **Add AND Comparison** dialog opens.

**Add AND Comparison**

Comparison Type: **Numeric Value Match**

**Protocol Details**

Protocol: **ftp**  
File Transfer Protocol

Protocol Field: **cmd-param-length**  
This field matches the Individual command string length ( with command qualifier as selected from below field"cmd-qualifier1").

cmd-qualifier1: **mkd**

**Numeric Details**

Operator: **Equals**

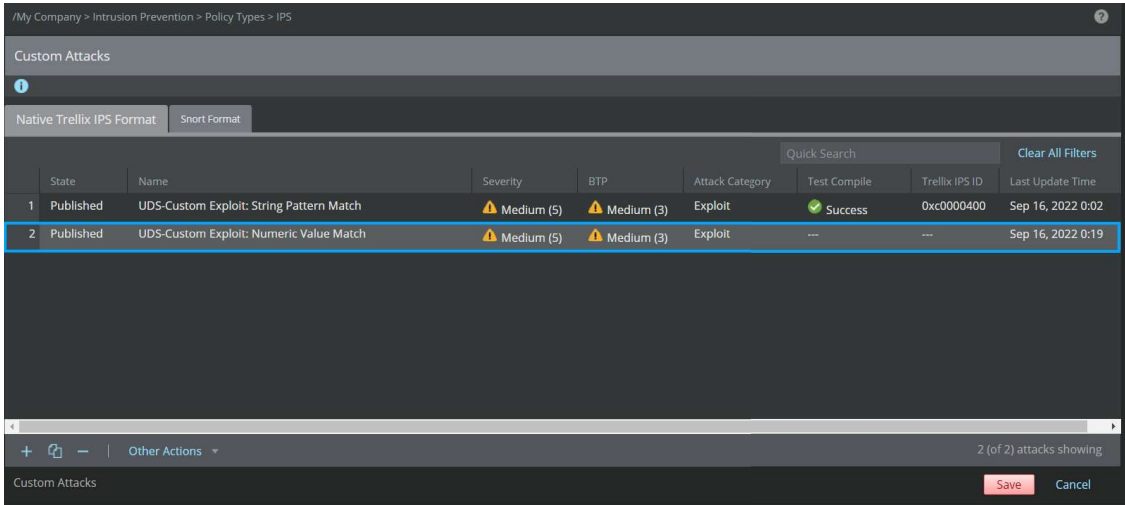
Value to Match: **50**

**Save**

Figure 4-37 Numeric Value Match

- 3 For this example, select **Numeric Value Match** in the **Comparison Type** drop-down menu.
- 4 Select **FTP** from the **Protocol** list.  
Because you selected FTP, the Custom Attack Editor displays the FTP-specific protocol fields on the following screen.
- 5 Configure the fields for the comparison you have chosen.  
For this example, specify **cmd-param-length** for the Protocol Field. This specifies that the Sensor should search in the URI of the request packet.
- 6 Select **mkd** as the qualifier.
- 7 From the **Operator** drop-down list, select the matching criteria as **Equals** which means that the comparison criteria must be equal to the regular expression entered.
- 8 Type the value to match using the **Value to Match**.  
For this example, the value to match is "50".
- 9 Click **Save**.  
Your comparison appears under **Condition 1**.
- 10 Click **Add** in the **New Custom Attack** window.

11 Verify that the attack definition is listed on the **Native Trellix IPS Format** tab.



12 Click **Save** to save the Trellix IPS Custom Attack in the Manager server database.

13 Make sure the Trellix IPS Custom Attack is saved in the database and also published in the policies.

Create an attack definition for numeric range match

This scenario is about creating a Custom Attack Definition that uses numeric range match to detect if the destination port range is between 100 and 2000 for HTTP protocol.

In order to write a proper Native Trellix IPS Format Custom Attack for this example, you must identify several key elements:

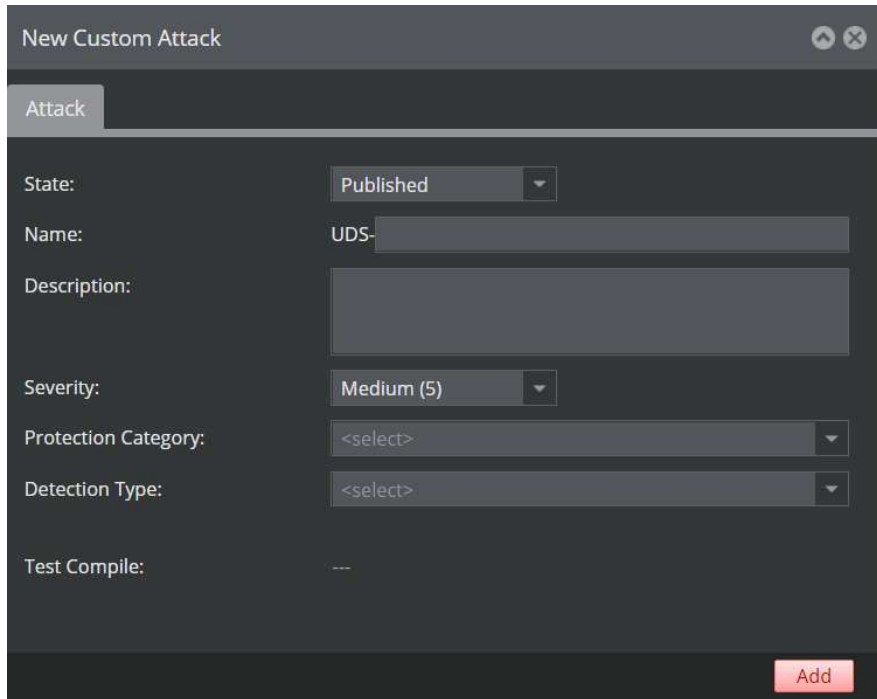
- Application protocol: HTTP
- Protocol field: dst-port
- Minimum value: 100
- Maximum value: 2000

To create the attack definition for this example:

#### Task

- 1 In the Custom Attack Editor, click **+**.

The **New Custom Attack** interface opens.



- 2 Select one of the options from the **State** drop-down menu.
- 3 You can specify a name such as "Custom Exploit: Numeric Range Match" as the **Name**.  
The letters "UDS" (user-defined signature) are appended to the front of the name upon completion; thus, this attack appears as "UDS-Custom Exploit: Numeric Range Match" in the Custom Attack Editor, as well as the attack database when you save in the Manager server.
- 4 Type a description for your attack. This area can be used for your notes or other specific information pertinent to your new attack.
- 5 You can select **Medium (5)** as the **Severity** because this example scenario does not necessarily involve malicious activity.
- 6 Select **Advanced Protection Options/Web Application Server** as the **Protection Category**.
- 7 Select **Custom Exploit (Signature Based)** from the **Detection Type** drop-down menu for the attack.

8 Click **Next**.

New Custom Attack

Attack

State:

Published

Name:

UDS- Custom Exploit: Numeric Range Match

Description:

Custom Exploit: Numeric Range Match

Severity:

Medium (5)

Protection Category:

Advanced Protection Options/Web Application Server

Detection Type:

Custom Exploit (Signature-Based)

Define an attack and its correlation details manually.

Test Compile:

---

Next >

9 Select **Client or Server** as **Attack Target**.

10 Select **Attack Packet Only** as **Blocking (As Applicable)**.

11 In the **Matching Criteria** section, select **Protocol** as the **Criterion** and select **ftp** as the protocol.

When you subsequently save this attack definition in the Manager server, it will be part of every rule set which includes Medium-severity, HTTP attacks (must match both severity and protocol).

New Custom Attack

Attack

Signature-1663268332509

+

Protection Category:

Advanced Protection Options/Web Application Ser

Attack Target:

Client or Server

Blocking (As Applicable):

Attack Packet Only

Benign Trigger Probability (BTP):

Medium (5)

Attack Category:

Exploit

Test Compile:

---

Trellix IPS ID:

---

Supported Device Types:

Any

Last Updated:

---

Matching Criteria

Criterion:

Protocol

Protocol:

ftp

Add

|   | Protocol | Software Package (OS) |   |
|---|----------|-----------------------|---|
| 1 | ftp      | ---                   | x |

12 Attack details configuration is complete. Continue to create signature.

Create the signature for numeric range match

After you create the attack definition, you create the signature for the attack.

## Task

- 1 In the New Custom Attack window, click **Signature-<signature name>** tab.

New Custom Attack

Attack Signature-1663268332509 +

Name: Signature-1663268332509

Benign Trigger Probability (BTP): Medium (3) ▾

Target Host Architecture: Any ▾

Detection Window: Single Packet ▾

Supported Device Types: Any ▾

Signature Details

Use the buttons below to add signature details. Each signature requires:

1. At least one condition.
2. At least one AND or OR comparison within each condition.

Condition 1

+ - | AND OR X

Figure 4-38 New Signature window

- 2 (Optional) Clear the **Name** and type a new name for your signature.
- 3 For this example, you can leave the **Benign Trigger Probability (BTP)**, **Target Host Architecture**, and **Detection Window** with the default values.
- 4 Based on the Sensor model that you plan to use for this example, select the **Supported Device Types**.
- 5 Add the condition to the signature.  
It is in the conditions that you specify the following details:
  - The string that the Sensor should look for.
  - Which section of the http request should it look for the string.
- 6 Proceed to add the condition.



### Add the condition for numeric range match

To add the condition to the example signature:

#### Task

- 1 Click **Condition 1** so that it is highlighted.
- 2 Click **AND** in the Comparisons section.

The **Add AND Comparison** dialog opens.

**Add AND Comparison**

Comparison Type: **Numeric Range Match**

**Protocol Details**

Protocol: **http**  
HyperText Transfer Protocol

Protocol Field: **dst-port**  
This field matches the destination port in a HTTP request.

**Numeric Details**

Operator: **Equals**

Minimum Value: **100**

Maximum Value: **2000**

Save

Figure 4-39 Numeric Range Match

- 3 For this example, select **Numeric Range Match** in the **Comparison Type** drop-down menu.
- 4 Select **http** from the **Protocol** list.  
Because you selected HTTP, the Custom Attack Editor displays the http specific protocol fields on the screen.
- 5 Configure the fields for the comparison you have chosen.  
For this example, specify **dst-port** for the Protocol Field. This specifies that the Sensor should search in the URI of the request packet.
- 6 From the drop-down list in the **Operator**, select the matching criteria as **Equals** which means that the comparison criteria must be between to the minimum and maximum values entered.
- 7 Type the minimum value in the **Mininum Value** field.  
For this example, the minimum value is "100".

- 8 Type the maximum value in the **Maximum Value** field.  
For this example, the value to match is "2000".
- 9 Click **Save**.  
Your comparison appears under **Condition 1**.
- 10 Click **Add** in the **New Custom Attack** window.
- 11 Verify that the attack definition is listed on the **Native Trellix IPS Format** tab.

/My Company > Intrusion Prevention > Policy Types > IPS

Custom Attacks

Native Trellix IPS Format

Snort Format

Quick Search

Clear All Filters

|   | State     | Name                                     | Severity     | BTP          | Attack Category | Test Compile | Trellix IPS ID | Last Update Time  |
|---|-----------|------------------------------------------|--------------|--------------|-----------------|--------------|----------------|-------------------|
| 1 | Published | UDS-Custom Exploit: String Pattern Match | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ✔ Success    | 0xc0000400     | Sep 16, 2022 0:02 |
| 2 | Published | UDS-Custom Exploit: Numeric Value Match  | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 0:19 |
| 3 | Published | UDS-Custom Exploit: Numeric Range Match  | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 0:32 |

+ - | Other Actions

3 (of 3) attacks showing

Custom Attacks

SaveCancel

- 12 Click **Save** to save the Trellix IPS Custom Attack in the Manager server database.
- 13 Make sure the Trellix IPS Custom Attack is saved in the database and also published in the policies.

Create an attack definition for numeric enumeration match

This scenario is about creating a Custom Attack Definition that uses numeric enumeration match to detect if the HTTP response code is either 302 or 404.

In order to write a proper Native Trellix IPS Format Custom Attack for this example, you must identify several key elements:

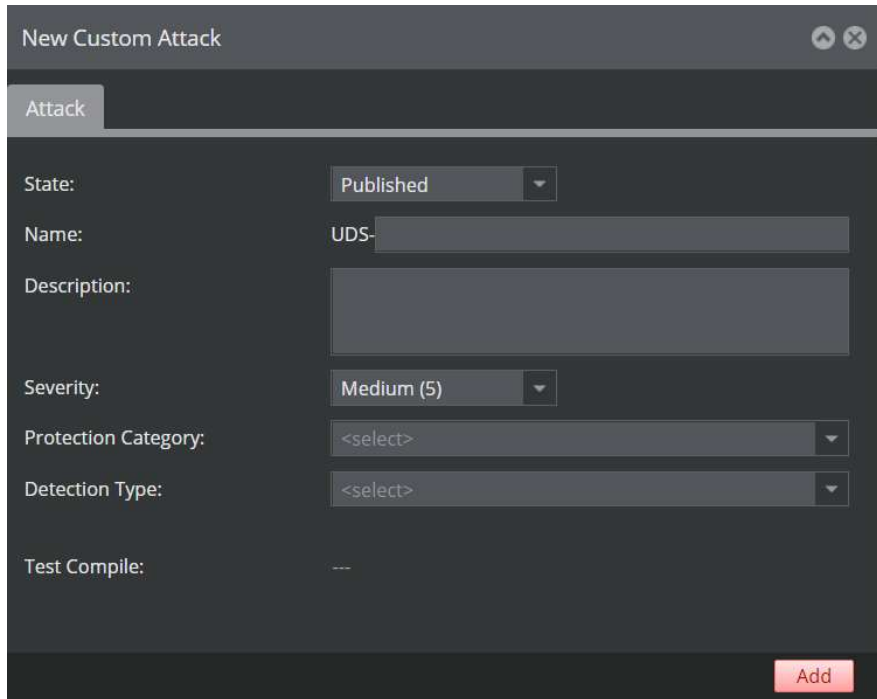
- Application protocol: HTTP
- Protocol field: rsp-code
- Enumeration value (integer): 302 and 404

To create the attack definition for this example:

#### Task

- 1 In the Custom Attack Editor, click **+**.

The **New Custom Attack** interface opens.



- 2 Select one of the options from the **State** drop-down menu.
- 3 You can specify a name such as "Custom Exploit: Numeric Enumeration Match" as the **Name**.  
The letters "UDS" (user-defined signature) are appended to the front of the name upon completion; thus, this attack appears as "UDS-Custom Exploit: Numeric Enumeration Match" in the Custom Attack Editor, as well as the attack database when you save in the Manager server.
- 4 Type a description for your attack. This area can be used for your notes or other specific information pertinent to your new attack.
- 5 You can select **Medium (5)** as the **Severity** because this example scenario does not necessarily involve malicious activity.
- 6 Select **Advanced Protection Options/Web Application Server** as the **Protection Category**.
- 7 Select **Custom Exploit (Signature Based)** from the **Detection Type** drop-down menu for the attack.

8 Click Next.

New Custom Attack

Attack

State:

Published

Name:

UDS- Custom Exploit: Numeric Enumeration Match

Description:

Custom Exploit: Numeric Enumeration Match

Severity:

Medium (5)

Protection Category:

Advanced Protection Options/Web Application Server

Detection Type:

Custom Exploit (Signature-Based)

Define an attack and its signature details manually.

Test Compile:

---

Next >

9 Select Client or Server as Attack Target.

10 Select Attack Packet Only as Blocking (As Applicable).

11 In the **Matching Criteria** section, select **Protocol** as the **Criterion** and select **http** as the protocol.

When you subsequently save this attack definition in the Manager server, it will be part of every rule set which includes Medium-severity, HTTP attacks (must match both severity and protocol).

New Custom Attack

Attack

Signature-1663268773387

+

Protection Category:

Advanced Protection Options/Web Application Ser

Attack Target:

Client or Server

Blocking (As Applicable):

Attack Packet Only

Benign Trigger Probability (BTP):

Medium (5)

Attack Category:

Exploit

Test Compile:

---

Trellix IPS ID:

---

Supported Device Types:

Any

Last Updated:

---

Matching Criteria

Criterion:

Protocol

Protocol:

http

Add

|   | Protocol | Software Package (OS) |   |
|---|----------|-----------------------|---|
| 1 | http     | ---                   | X |

12 Attack details configuration is complete. Continue to create signature.

Create the signature for numeric enumeration match

After you create the attack definition, you create the signature for the attack.

Task

- 1 In the New Custom Attack window, click Signature-<signature name> tab.

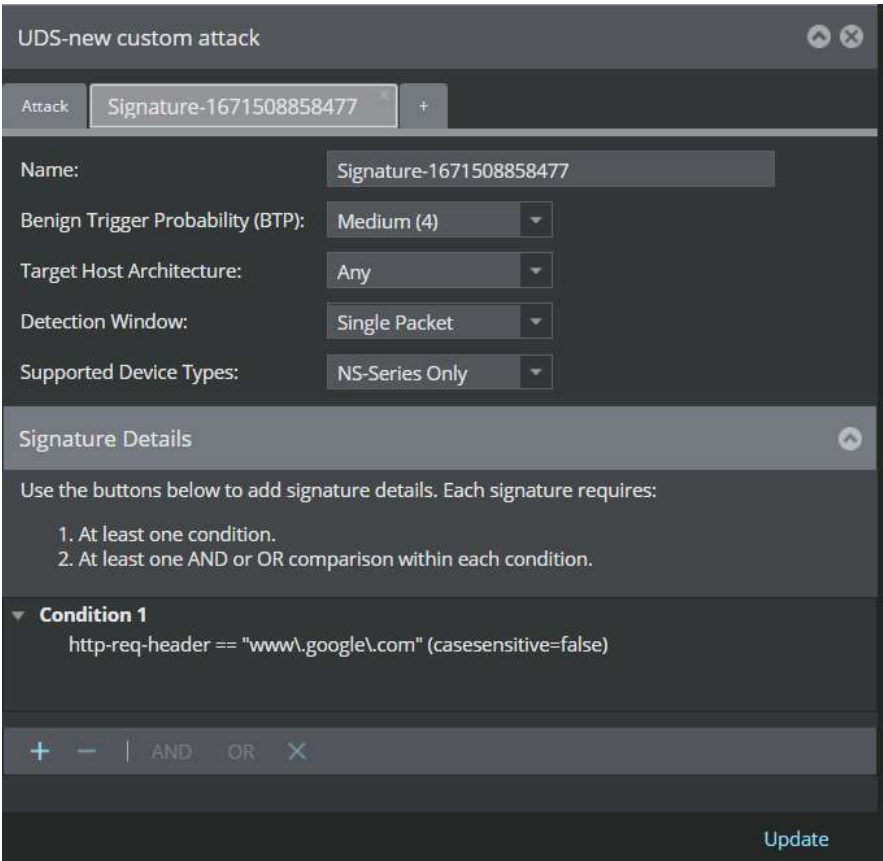


Figure 4-40 New Signature window

- 2 (Optional) Clear the **Name** and type a new name for your signature.
- 3 For this example, you can leave the **Benign Trigger Probability (BTP)**, **Target Host Architecture**, and **Detection Window** with the default values.
- 4 Based on the Sensor model that you plan to use for this example, select the **Supported Device Types**.
- 5 Add the condition to the signature.  
It is in the conditions that you specify the following details:
- The string that the Sensor should look for
  - Which section of the HTTP request should it look for the string
- 6 Proceed to add the condition.

Add the condition for numeric enumeration match

To add the condition to the example signature:

Task

- 1 Click **Condition 1** so that it is highlighted.
- 2 Click **AND**.

The **Add AND Comparison** dialog opens.

Add AND Comparison

Comparison Type: Numeric Enumeration Match

Protocol Details

Protocol: http  
HyperText Transfer Protocol

Protocol Field: rsp-code  
This field matches the response code returned from the web server.

Enumeration Details

New Integer: -2,147,483,648 - 2147483647 Add

|   | Integers |   |
|---|----------|---|
| 1 | 302      | × |
| 2 | 404      | × |

Save

Figure 4-41 Numeric Enumeration Match

- 3 For this example, select **Numeric Enumeration Match** in the **Comparison Type** drop-down menu.
- 4 Select **http** from the **Protocol** list.  
Because you selected HTTP, the Custom Attack Editor displays the HTTP-specific protocol fields on the following screen.
- 5 Configure the fields for the comparison you have chosen.  
For this example, specify **rsp-code** for the Protocol Field. This specifies that the Sensor should search in the URI of the request packet.
- 6 In the **New Integer** field, enter the in the numeric value.
- 7 Click **Add**.  
You can add multiple values in the **New Integer** field. For this example enter 302 and 404.

- 8 Click **Save**.  
Your comparison appears under **Condition 1**.
- 9 Click **Add** in the **New Custom Attack** window.
- 10 Verify that the attack definition is listed on the **All Custom Attacks** tab.

/My Company > Intrusion Prevention > Policy Types > IPS

Custom Attacks

Native Trellix IPS Format

Snort Format

Quick Search

Clear All Filters

|   | State     | Name                                          | Severity     | BTP          | Attack Category | Test Compile | Trellix IPS ID | Last Update Time  |
|---|-----------|-----------------------------------------------|--------------|--------------|-----------------|--------------|----------------|-------------------|
| 1 | Published | UDS-Custom Exploit: String Pattern Match      | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ✔ Success    | 0xc0000400     | Sep 16, 2022 0:02 |
| 2 | Published | UDS-Custom Exploit: Numeric Value Match       | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 0:19 |
| 3 | Published | UDS-Custom Exploit: Numeric Range Match       | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 0:32 |
| 4 | Published | UDS-Custom Exploit: Numeric Enumeration Match | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 0:39 |

4 (of 4) attacks showing

Custom Attacks

Save Cancel

- 11 Click **Save** to save the Trellix IPS Custom Attack in the Manager server database.
- 12 Make sure the Trellix IPS Custom Attack is saved in the database and also published in the policies.

Create an attack definition for single fixed field match

This scenario is about creating a Custom Attack Definition that uses single fixed field match to detect if the IPv4 address matches 103.17.47.20 in HTTP protocol.

In order to write a proper Native Trellix IPS Format Custom Attack for this example, you must identify several key elements:

- Application protocol: IPv4
- Protocol field: ipv4-destination-ip
- IP address to match: 103.17.47.20

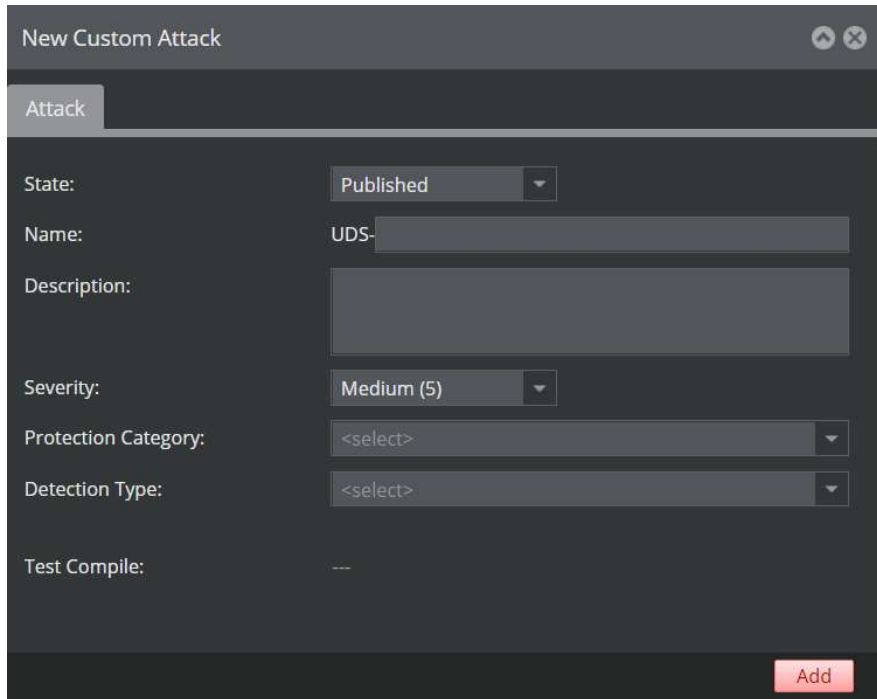


To create the attack definition for this example:

#### Task

- 1 In the Custom Attack Editor, click **+**.

The **New Custom Attack** interface opens.



- 2 Select one of the options from the **State** drop-down menu.
- 3 You can specify a name such as "Custom Exploit: Single Fixed Field Match" as the **Name**.  
The letters "UDS" (user-defined signature) are appended to the front of the name upon completion; thus, this attack appears as "UDS-Custom Exploit: Single Fixed Field Match" in the Custom Attack Editor, as well as the attack database when you save in the Manager server.
- 4 Type a description for your attack. This area can be used for your notes or other specific information pertinent to your new attack.
- 5 You can select **Medium (5)** as the **Severity** because this example scenario does not necessarily involve malicious activity.
- 6 Select **Advanced Protection Options/Web Application Server** as the **Protection Category**.
- 7 Select **Custom Exploit (Signature Based)** from the **Detection Type** drop-down menu for the attack.

8 Click **Next**.

New Custom Attack

Attack

State:

Published

Name:

UDS- Custom Exploit: Single Fixed Field Match

Description:

Custom Exploit: Single Fixed Field Match

Severity:

Medium (5)

Protection Category:

Advanced Protection Options/Web Application Server

Detection Type:

Custom Reconnaissance Attack (Correlation-Based)

Define an attack and its correlation details manually.

Test Compile:

---

Next >

9 Select **Client** or **Server** as Attack Target.

10 Select **Attack Packet Only** as **Blocking (As Applicable)**.

- 11 In the **Matching Criteria** section, select **Protocol** from the **Criterion** and select **http** as the protocol.
- When you subsequently save this attack definition in the Manager server, it will be part of every rule set which includes Medium-severity, HTTP attacks (must match both severity and protocol).

New Custom Attack

Attack

Signature-1663272026111

+

Protection Category:

Advanced Protection Options/Web Application Ser

Attack Target:

Client or Server

Blocking (As Applicable):

Attack Packet Only

Benign Trigger Probability (BTP):

Medium (5)

Attack Category:

Exploit

Test Compile:

---

Trellix IPS ID:

---

Supported Device Types:

Any

Last Updated:

---

Matching Criteria

Criterion:

Protocol

Protocol:

http

Add

|   | Protocol | Software Package (OS) |   |
|---|----------|-----------------------|---|
| 1 | http     | ---                   | X |

- 12 Attack details configuration is complete. Continue to create signature.

Create the signature for single fixed field match

After you create the attack definition, you create the signature for the attack.

Task

- 1 In the New Custom Attack window, click Signature-<signature name> tab.

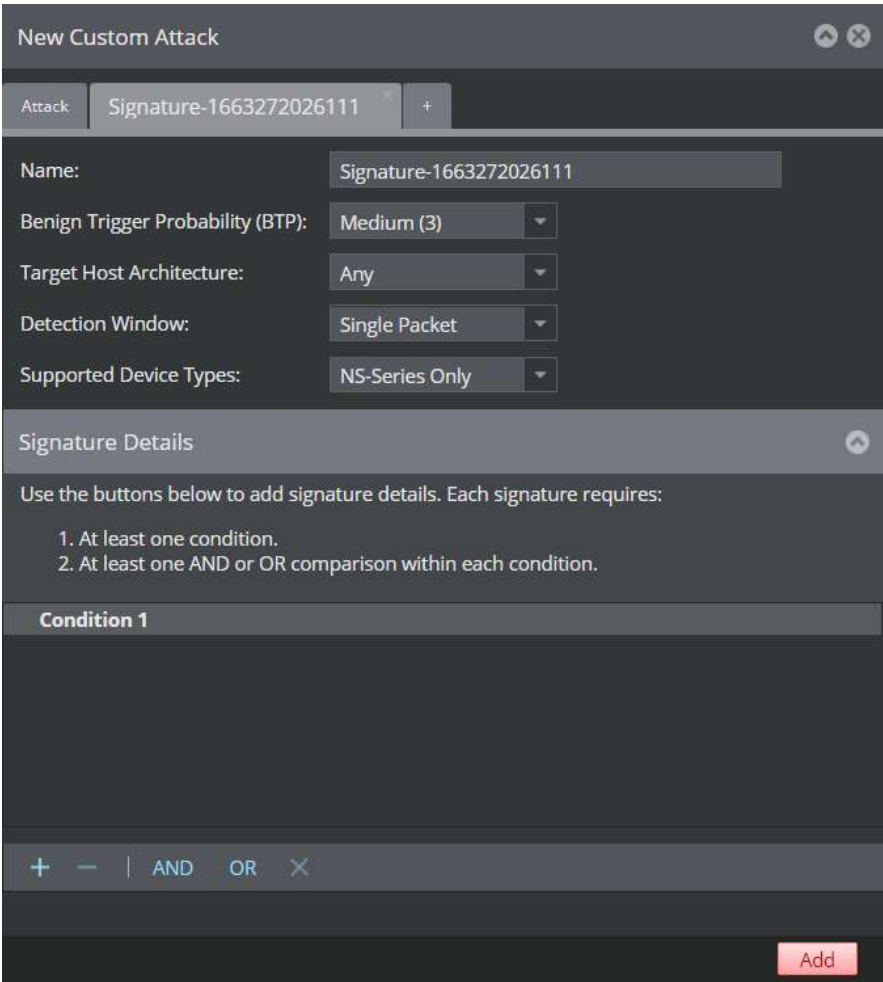


Figure 4-42 New Signature window

- 2 (Optional) Clear the **Name** and type a new name for your signature.
- 3 For this example, you can leave the **Benign Trigger Probability (BTP)**, **Target Host Architecture**, and **Detection Window** with the default values.
- 4 Based on the Sensor model that you plan to use for this example, select the **Supported Device Types**.
- 5 Add the condition to the signature.  
It is in the conditions that you specify the following details:
- The string that the Sensor should look for
  - Which section of the HTTP request should it look for the string
- 6 Proceed to add the condition.

### Add the condition for single fixed field match

To add the condition to the example signature:

#### Task

- 1 Click **Condition 1** so that it is highlighted.
- 2 Click **AND** in the Comparisons section.

The **Add AND Comparison** dialog opens.

The screenshot shows the 'Add AND Comparison' dialog box with the following configuration:

- Comparison Type:** Single Fixed Field Match
- Protocol Details:**
  - Protocol:** ipv4 (Internet Protocol Version 4)
  - Protocol Field:** ipv4-destination-ip (Destination IP address, 4 bytes)
- Field Match Details:**
  - Operator:** Equals
  - Integer or IP to Match:** 192.168.1.1
  - Ignore Bitmask:** ☒

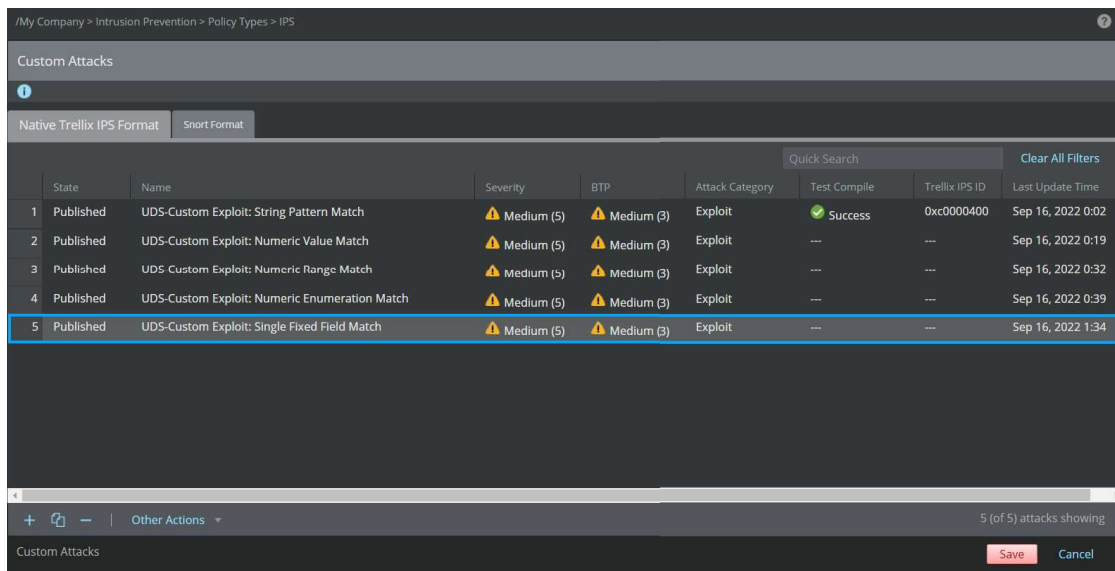
A 'Save' button is located at the bottom right of the dialog.

Figure 4-43 Single Fixed Field Match

- 3 For this example, select **Single Fixed Field Match** in the **Comparison Type** drop-down menu.
- 4 Select **ipv4** from the **Protocol** list.  
Because you selected IPv4, the Custom Attack Editor displays the IPv4-specific protocol fields on the following screen.
- 5 Configure the fields for the comparison you have chosen.  
For this example, specify **ipv4-destination-ip** for the Protocol Field.
- 6 From the **Operator** drop-down list, select the matching criteria as **Equals** which means that the comparison criteria must be equal to the IP address entered.
- 7 Type the IP address in the **Integer or IP to Match**.  
For this example, the value to match is "192.168.1.1".
- 8 (Optional) Uncheck **Ignore Bitmask** if you want to process the bitmask of the IP address.

9 Click **Save**.

Your comparison appears under **Condition 1**.

10 Click **Add** in the **New Custom Attack** window.11 Verify that the attack definition is listed on the **Native Trellix IPS Format** tab.


The screenshot shows the 'Custom Attacks' window in the Trellix IPS interface. The 'Native Trellix IPS Format' tab is selected. A table lists 5 attacks, with the first one highlighted. The table has columns for State, Name, Severity, BTP, Attack Category, Test Compile, Trellix IPS ID, and Last Update Time.

|   | State     | Name                                          | Severity     | BTP          | Attack Category | Test Compile | Trellix IPS ID | Last Update Time  |
|---|-----------|-----------------------------------------------|--------------|--------------|-----------------|--------------|----------------|-------------------|
| 1 | Published | UDS-Custom Exploit: String Pattern Match      | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ✓ Success    | 0xc0000400     | Sep 16, 2022 0:02 |
| 2 | Published | UDS-Custom Exploit: Numeric Value Match       | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 0:19 |
| 3 | Published | UDS-Custom Exploit: Numeric Range Match       | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 0:32 |
| 4 | Published | UDS-Custom Exploit: Numeric Enumeration Match | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 0:39 |
| 5 | Published | UDS-Custom Exploit: Single Fixed Field Match  | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 1:34 |

12 Click **Save** to save the Trellix IPS Custom Attack in the Manager server database.

## 13 Make sure the Trellix IPS Custom Attack is saved in the database and also published in the policies.

## Create an attack definition for packet grep protocol match

This scenario is about creating a Custom Attack Definition that uses packet grep protocol match to detect if the DNP3 protocol contains a specific character sequence.

In order to write a proper Native Trellix IPS Format Custom Attack for this example, you must identify several key elements:

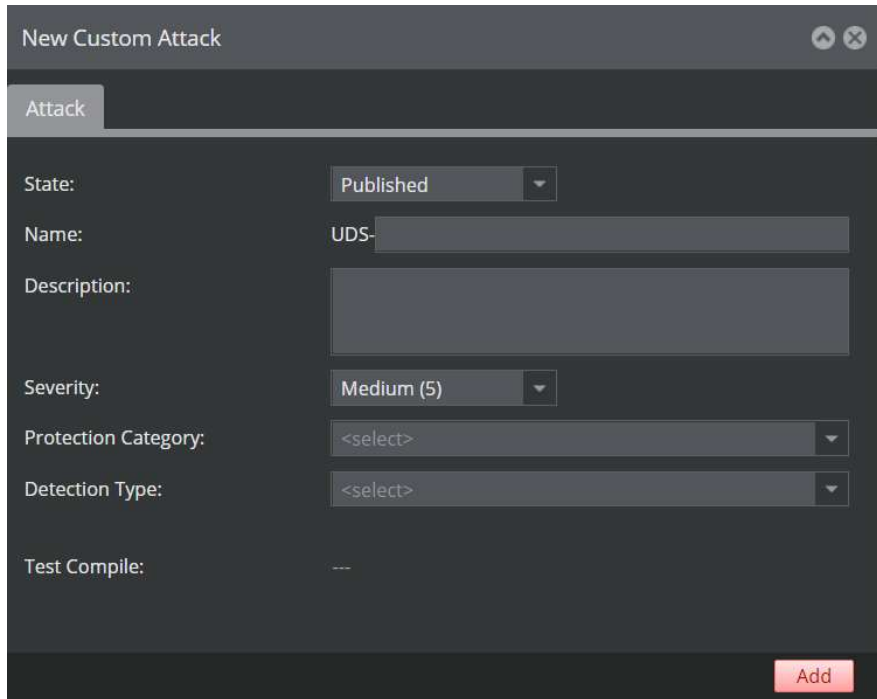
- Application protocol: dnp3
- Packets to parse: Request Packets Only
- Text to match: \x00\x00\x00\x3a\x20\x45\x56\x00\x0a

To create the attack definition for this example:

#### Task

- 1 In the Custom Attack Editor, click **+**.

The **New Custom Attack** interface opens.



- 2 Select one of the options from the **State** drop-down menu.
- 3 You can specify a name such as "Custom Exploit: Packet Grep Protocol Match" as the **Name**.  
The letters "UDS" (user-defined signature) are appended to the front of the name upon completion; thus, this attack appears as "UDS-Custom Exploit: Packet Grep Protocol Match" in the Custom Attack Editor, as well as the attack database when you save in the Manager server.
- 4 Type a description for your attack. This area can be used for your notes or other specific information pertinent to your new attack.
- 5 You can select **Medium (5)** as the **Severity** because this example scenario does not necessarily involve malicious activity.
- 6 Select **Advanced Protection Options/Web Application Server** as the **Protection Category**.
- 7 Select **Custom Exploit (Signature Based)** from the **Detection Type** drop-down menu for the attack.

8 Click **Next**.

New Custom Attack

Attack

State:

Published

Name:

UDS- Custom Exploit: Packet Grep Protocol Match

Description:

Custom Exploit: Packet Grep Protocol Match

Severity:

Medium (5)

Protection Category:

Advanced Protection Options/Web Application Server

Detection Type:

Custom Exploit (Signature-Based)

Define an attack and its signature details manually.

Test Compile:

---

Next >

- 9 Select **Client or Server** as **Attack Target**.
- 10 Select **Attack Packet Only** as **Blocking (As Applicable)**.



11 In the **Matching Criteria** section, select **Protocol** as the **Criterion** and select **pktsearch** as the protocol.

New Custom Attack

AttackSignature-1663273025724+

Protection Category:Advanced Protection Options/Web Application Ser

Attack Target:Client or Server

Blocking (As Applicable):Attack Packet Only

Benign Trigger Probability (BTP):Medium (5)

Attack Category:Exploit

Test Compile:---

Trellix IPS ID:---

Supported Device Types:Any

Last Updated:---

Matching Criteria

Criterion:Protocol

Protocol:pktsearchAdd

|   | Protocol  | Software Package (OS) |
|---|-----------|-----------------------|
| 1 | pktsearch | ---                   |

12 Attack details configuration is complete. Continue to create signature.

Create the signature for packet grep protocol match

After you create the attack definition, you create the signature for the attack.

Task

- 1 In the New Custom Attack window, click Signature-<signature name> tab.

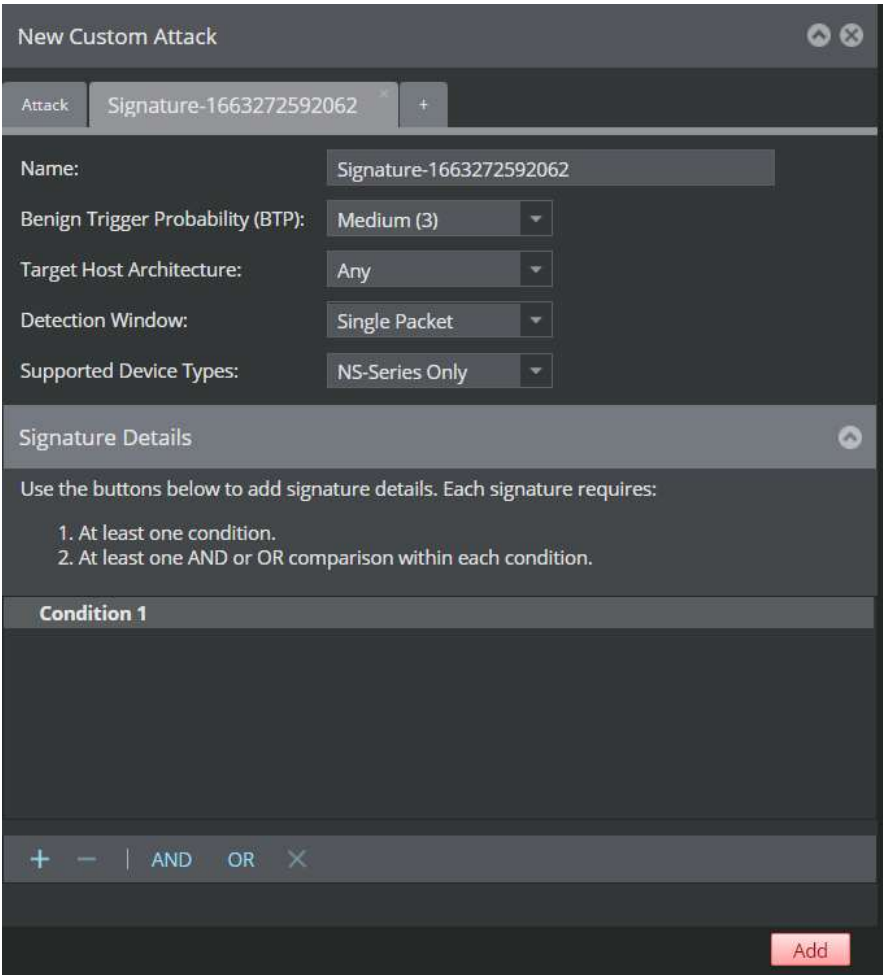


Figure 4-44 New Signature window

- 2 (Optional) Clear the **Name** and type a new name for your signature.
- 3 For this example, you can leave the **Benign Trigger Probability (BTP)**, **Target Host Architecture**, and **Detection Window** with the default values.
- 4 Based on the Sensor model that you plan to use for this example, select the **Supported Device Types**.
- 5 Add the condition to the signature.  
It is in the conditions that you specify the following details:
- The string that the Sensor should look for
  - Which section of the HTTP request should it look for the string
- 6 Proceed to add the condition.

### Add the condition for packet grep protocol match

To add the condition to the example signature:

#### Task

- 1 Click **Condition 1** so that it is highlighted.
- 2 Click **AND** in the Comparisons section.

The **Add AND Comparison** dialog opens.

**Add AND Comparison**

Comparison Type: **Packet Grep Protocol Match**

▲ Grepping Protocol Details

Protocol: **dnp3**  
TCP 20000

Parse: **Request Packets Only**

▲ Regex Details

Operator: **Equals**

Text to Match: **\x00\x00\x00\x3a\x20\x45\x56\x00\x0a** ✓

Ignore Case: ☒

Ignore String Position: ☒

Save

Figure 4-45 Packet Grep Protocol Match

- 3 For this example, select **Packet Grep Protocol Match** in the **Comparison Type** drop-down menu.
- 4 Select **dnp3** from the **Protocol** list.  
Because you selected IPv4, the Custom Attack Editor displays the IPv4-specific protocol fields on the following screen.
- 5 Configure the packets you want to parse.  
For this example, specify **Request Packets Only** for the **Parse** field.
- 6 From the **Operator** drop-down list, select the matching criteria as **Equals**.
- 7 Type the text you are searching for in the packets in the **Text to Match**.  
For this example, the text to match is " \x00\x00\x00\x3a\x20\x45\x56\x00\x0a".
- 8 (Optional) Deselect **Ignore Case** and **Ignore String Position**.

**9** Click **Save**.

Your comparison appears under **Condition 1**.

**10** Click **Add** in the **New Custom Attack** window.**11** Verify that the attack definition is listed on the **Native Trellix IPS Format** tab.

|   | State     | Name                                           | Severity     | BTP          | Attack Category | Test Compile | Trellix IPS ID | Last Update Time  |
|---|-----------|------------------------------------------------|--------------|--------------|-----------------|--------------|----------------|-------------------|
| 1 | Published | UDS-Custom Exploit: String Pattern Match       | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ✓ Success    | 0xc0000400     | Sep 16, 2022 0:02 |
| 2 | Published | UDS-Custom Exploit: Numeric Value Match        | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 0:19 |
| 3 | Published | UDS-Custom Exploit: Numeric Range Match        | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 0:32 |
| 4 | Published | UDS-Custom Exploit: Numeric Enumeration Match  | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 0:39 |
| 5 | Published | UDS-Custom Exploit: Single Fixed Field Match   | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 1:34 |
| 6 | Published | UDS-Custom Exploit: Packet Grep Protocol Match | ⚠ Medium (5) | ⚠ Medium (3) | Exploit         | ---          | ---            | Sep 16, 2022 1:52 |

**12** Click **Save** to save the Trellix IPS Custom Attack in the Manager server database.**13** Make sure the Trellix IPS Custom Attack is saved in the database and also published in the policies.

## Custom attack with a pattern-match signature

### Scenario

This is about creating a Custom Attack Definition that uses pattern matching to detect any HTTP GET requests in the URI path for a specific CGI script. This scenario is useful because it illustrates the simplest method of configuring a custom attack definition with proper syntax, defined options, and case sensitivity.

Creation of such an attack definition is also a good example of activity that may not be a malicious attack; rather, you could create such an instance and use it to track requests for sensitive information.

In order to write a proper Native Trellix IPS Format Custom Attack or a Snort Custom Attack for this example, you must identify several key elements:

- Application protocol: HTTP
- Where to look: URI path
- Detection window: request
- Request method: get
- String to match: cgi.bin/trillion.pl or cgi.bin/trilliant.pl, where only "pl" is case insensitive

When you have all of the required elements to properly identify the activity, attack definition can be successful.

To create a custom attack definition:

- Open the Custom Attack Editor.  
Select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click **Custom Attacks**.

### Create a custom attack

This section explains how to create a Native Trellix IPS Format Custom Attack definition for the scenario explained in the previous section.

#### Create an attack definition

To create the attack definition for this example:

##### Task

- 1 In the Custom Attack Editor, click .  
The **New Custom Attack** interface opens.
- 2 Select one of the options from the **State** drop-down menu.
- 3 You can specify a name such as "CGI: Trillion or Trilliant" as the **Name**.  
The letters "UDS" (user-defined signature) are appended to the front of the name upon completion; thus, this attack appears as "UDS-CGI: Trillion or Trilliant" in the Custom Attack Editor, as well as the attack database when you save in the Manager server.
- 4 Type a description for your attack. This area can be used for your notes or other specific information pertinent to your new attack.
- 5 You can select **Medium** as the **Severity** because this example scenario does not necessarily involve malicious activity.
- 6 Select **Advanced Protection Options/Web Application Server Protection** as the **Protection Category**.
- 7 Select **URL** from the **Detection Type** drop-down menu for the attack.
- 8 Enter the **URL** that is to be detected, then click **Add**.
- 9 In the **Matching Criteria** section, the protocol is automatically selected as **http**.  
When you subsequently save this attack definition in the Manager server, it will be part of every rule set which includes Medium-severity, HTTP attacks (must match both severity and protocol).
- 10 Attack details configuration is complete. Continue to create signature.

#### Create the signature

After you create the attack definition, you create the signature for the attack.

##### Task

- 1 Click **Signature-<signature name>** tab.
- 2 (Optional) Clear the **Name** and type a new name for your signature.
- 3 For this example, you can leave the **Benign Trigger Probability (BTP)** and **Target Host Architecture** with the default values.
- 4 This example is to search for a string in the HTTP URI. So, select **Request Packets** as the **Detection Window**.

5 Based on the Sensor model that you plan to use for this example, select the **Supported Device Types**.

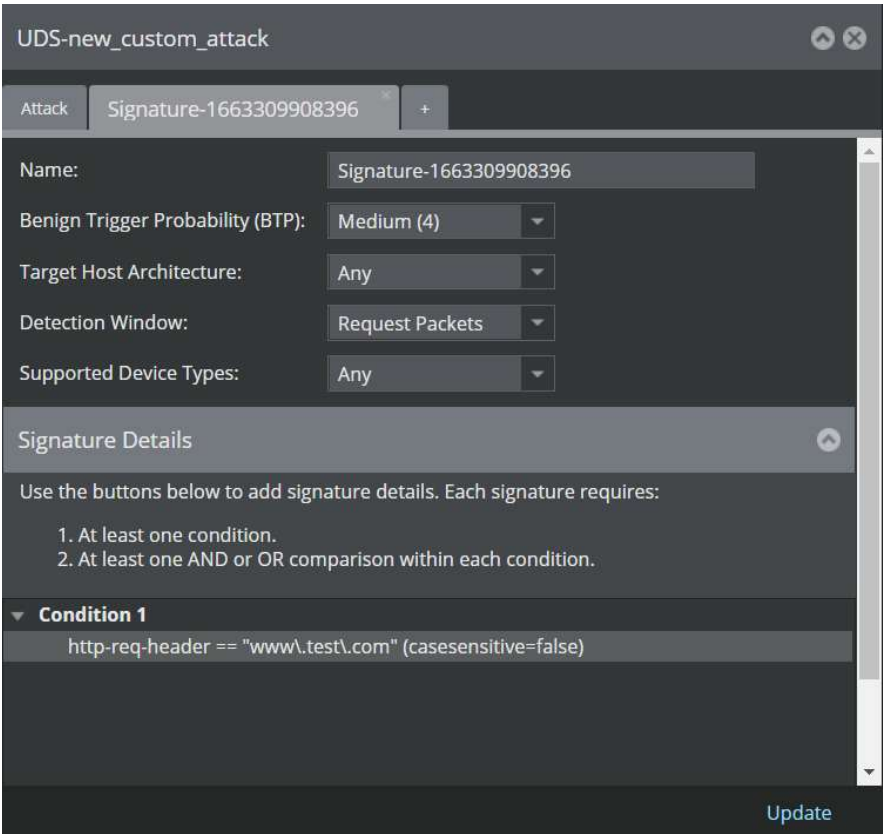


Figure 4-46 New Signature window

- 6 Add the condition to the signature.
- It is in the conditions that you specify the following details:
- The string that the Sensor should look for
  - Which section of the HTTP request should it look for the string
- 7 Proceed to add the condition.

Tasks

- *Add the condition on page 1656*

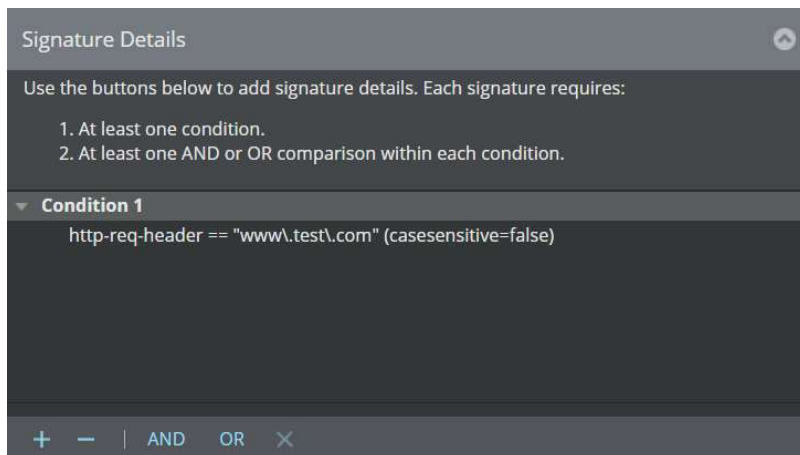
See also  
*Signature creation interface on page 1519*

Add the condition

To add the condition to the example signature:

#### Task

- 1 **Condition 1** is added automatically according to the URL you provided.
- 2 Click **Condition 1** so that it is highlighted.



- 3 Click **AND** in the Comparisons section.  
The **Add AND Comparison** dialog opens.
- 4 For this example, select **String Pattern Match** in the **Comparison Type** drop-down menu.
- 5 Select **http** from the **Protocol** list.  
Because you selected HTTP, the Custom Attack Editor displays the HTTP-specific protocol fields on the following screen.
- 6 Configure the fields for the comparison you have chosen.  
For this example, specify **req-uri-path** for the Protocol Field. This specifies that the Sensor should search in the URI of the request packet.
- 7 Select **get** as the http request method.
- 8 From the **Operator** drop-down list, select the matching criteria as **Equals** which means that the comparison criteria must be equal to the regular expression entered.
- 9 Type the pattern to match using the **Text to Match**.  
For this example, the pattern to match is either "cgi.bin/trillion.pl" or "cgi.bin/trilliant.pl", where "pl" is case-insensitive. To properly write this expression, use the following rules:
  - Add a backward slash (\) before every dot (.) to escape: cgi.bin = cgi\bin
  - Use alternatives where possible. For this example, trillion and trilliant can be written as: trilli(on|ant).
  - Use character classes to denote case insensitivity for "pl": [Pp][Ll]

The final string should appear as: `cgi\.bin/trilli(on|ant)\.[Pp][Ll]`

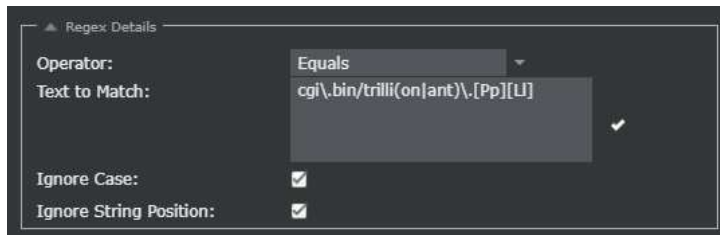


Figure 4-47 Regular Expression details

- 10 Click ☒ to verify that your expression is a valid string, and all required options are represented. Click **OK** to close the validation message window.
- 11 Click **Save**.  
Your comparison appears under **Condition 1**.
- 12 Click **Save** in the **Add Exploit Attack** window.
- 13 Verify that the attack definition is listed on the **Native Trellix IPS Format** tab.
- 14 Click **Save** to save the Trellix IPS Custom Attack in the Manager server database.
- 15 Make sure the Trellix IPS Custom Attack is saved in the database and also published in the policies.

See also

*Viewing a policy to verify inclusion of the attack on page 1604*

## Create a snort custom attack — an example

To write a Snort Custom Attack definition for the scenario explained in the previous section:

### Task

- 1 In the Custom Attack Editor, click **Snort Format** tab.
- 2 Click .
- The **New Snort Rule** text box opens.
- 3 Select one of the options from the **State** drop-down menu.
- 4 Construct the Snort rule for the scenario, which is the following:



```
alert tcp any any -> 192.168.1.1 80 (msg:"CGI: Trillion or Trilliant";content:"cgi.bin\trilli";http_uri;pcre:"/cgi.bin\trilli(on|ant).
[Pp][L]"/";priority:2;sid:2051;rev:1;)
```

Figure 4-48 Add Snort Rule window

Some points to note regarding the rule:

- Specifying the destination IP address improves the performance of the Sensor.
- As you are adding the Snort rule directly in the Custom Attack Editor, as a best practice, you need to specify the destination port number. The Sensor then checks only the HTTP request packet for this rule.
- You need to specify the msg, priority, SID, and the revision number.

**5** Select **Check for Overlap with Trellix IPS Attacks** to verify if the rule matches an existing Trellix IPS attack definition.

**6** Select **Advanced Protection Options/Web Application Server Protection** as the **Protection Category**.

- 7 Click **Add**.  
The **Attack** window opens.

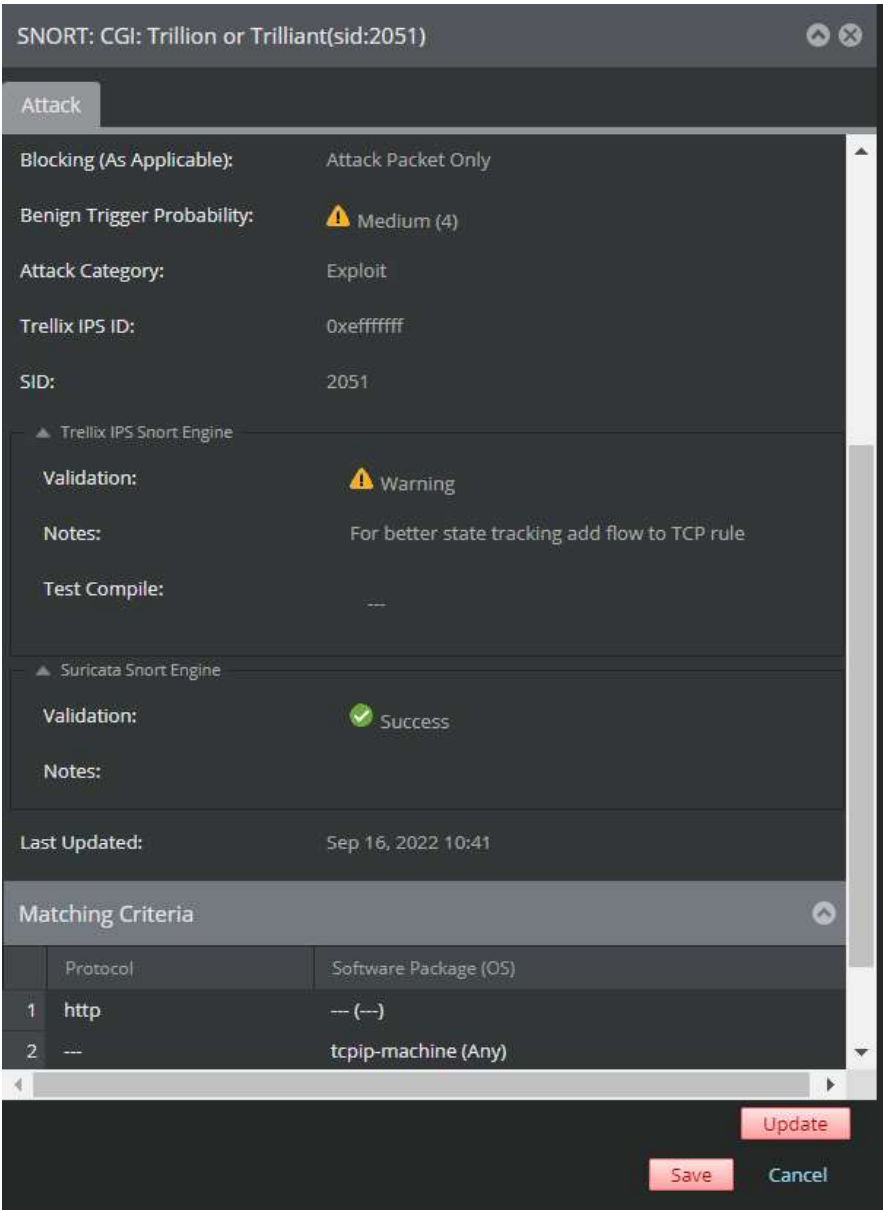


Figure 4-49 Edit Snort Attack window

- 8 Note that the Manager assigns the Attack Name based on the msg option and SID of the rule.
- 9 Set the severity to Medium. This is because the priority of the rule is 2.
- 10 Note that the Protocol is set to **http**. This is because the destination port number is 80.
- 11 Note that the **Benign Trigger Probability** is set to the default value.
- 12 Close the Attack window.
- 13 In the **Custom Attacks** window, click **Save** to save the Snort Custom Attack in the Manager server database.
- 14 Make sure the Snort Custom Attack is saved in the database and also published in the policies.

See also

*Identification of the protocol of a snort custom attack on page 1599*

*Viewing a policy to verify inclusion of the attack on page 1604*

### Custom attacks with multiple comparisons

In case of Trellix IPS Custom Attacks, you can use multiple conditions and/or comparisons within a single attack definition to increase your confidence in detecting an unknown attack. There are many configuration possibilities; the procedure that follows details the following scenario:

- One signature with two conditions. Both conditions must be met before an alert is generated.
- The first condition has a single comparison.
- The second condition employs an OR comparison; that is, either of the two comparisons in the condition signifies a positive match.

You can use various comparison methods within a condition. For example, if you employ two comparisons within a condition, one comparison can be a string match, while the second can be a fixed field check.

To create the Trellix IPS Custom Attack definition:

- Open the Custom Attack Editor.

Select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click **Custom Attacks**

### Create an attack definition for the example

To create the attack definition for this example:

#### Task

- 1 Select **Policy** | **<Admin Domain Name>** | **Intrusion Prevention** | **Policy Types** | **IPS**. Click **Custom Attacks**.

The Custom Attack Editor opens with the existing Custom Attacks listed on the Native Trellix IPS Format tab.

- 2 Click .

The **New Custom Attack** interface opens.

- 3 In the **Name** field, type a new name for your attack.

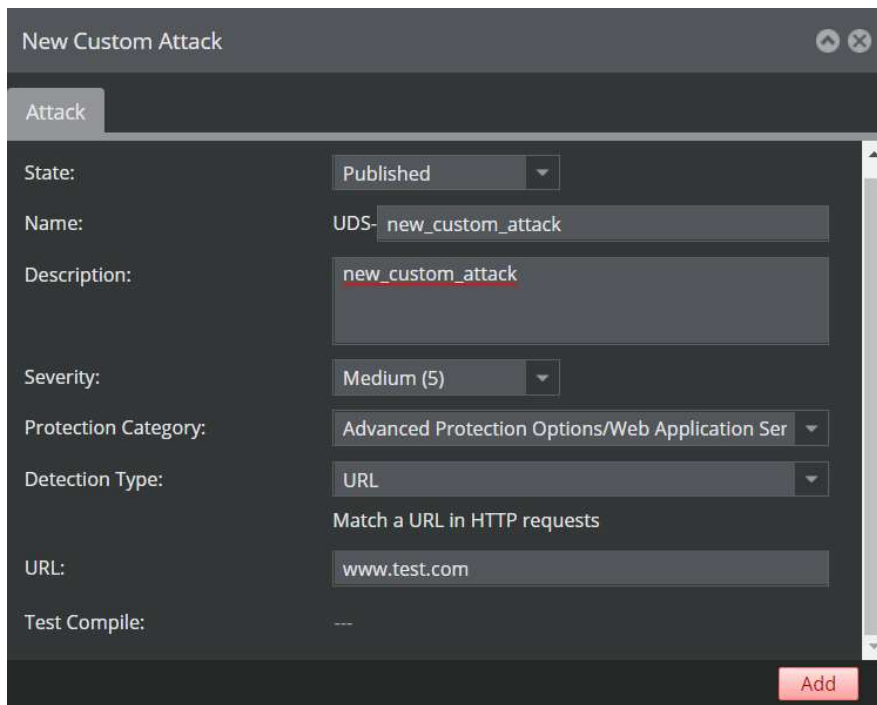
The letters "UDS" (user-defined signature) are appended to the front of the name upon completion when you save it in the Manager server.

- 4 Type a description for your attack. This area can be used for your notes or other specific information pertinent to your new attack.

- 5 Select **Advanced Protection Options/Web Application Server Protection** as the **Protection Category**.

- 6 Select **URL** from the **Detection Type** list.

- 7 Enter the **URL** that is to be detected, then click **Add**.



New Custom Attack

Attack

State: Published

Name: UDS- new\_custom\_attack

Description: new\_custom\_attack

Severity: Medium (5)

Protection Category: Advanced Protection Options/Web Application Ser

Detection Type: URL

Match a URL in HTTP requests

URL: www.test.com

Test Compile: ---

Add

- 8 In the **Matching Criteria** section, the protocol is automatically selected as HTTP.
- 9 Select **Software Package (OS)** as the **Criterion**.
- 10 Select **iis** from the **Software Package** list .
- 11 Select **windows** from the **Operating System** list and then click **Add**.

12 Verify that both HTTP and IIS appear in the Matching Criteria table.

UDS-new\_custom\_attack

Attack

Signature-1663309908396

+

Attack Target:

Server

Blocking (As Applicable):

Attack Packet Only

Benign Trigger Probability (BTP):

Medium (4)

Attack Category:

Exploit

Test Compile:

---

Trellix IPS ID:

---

Supported Device Types:

Any

Last Updated:

Sep 16, 2022 12:01

Matching Criteria

Criterion:

Software Package (C

Software Package:

iis

Operating System:

windows

Add

|   | Protocol | Software Package (OS) |   |
|---|----------|-----------------------|---|
| 1 | http     | ---                   | X |

Update

13 Click Update.

14 Attack details configuration is complete. Continue to create signature.

Tasks

- Create the signature for the example on page 1663

Create the signature for the example

After you create the attack definition, you create the signature for the attack.

Task

- 1 In the New Custom Attack interface, click on the Signature-<signature name> tab.

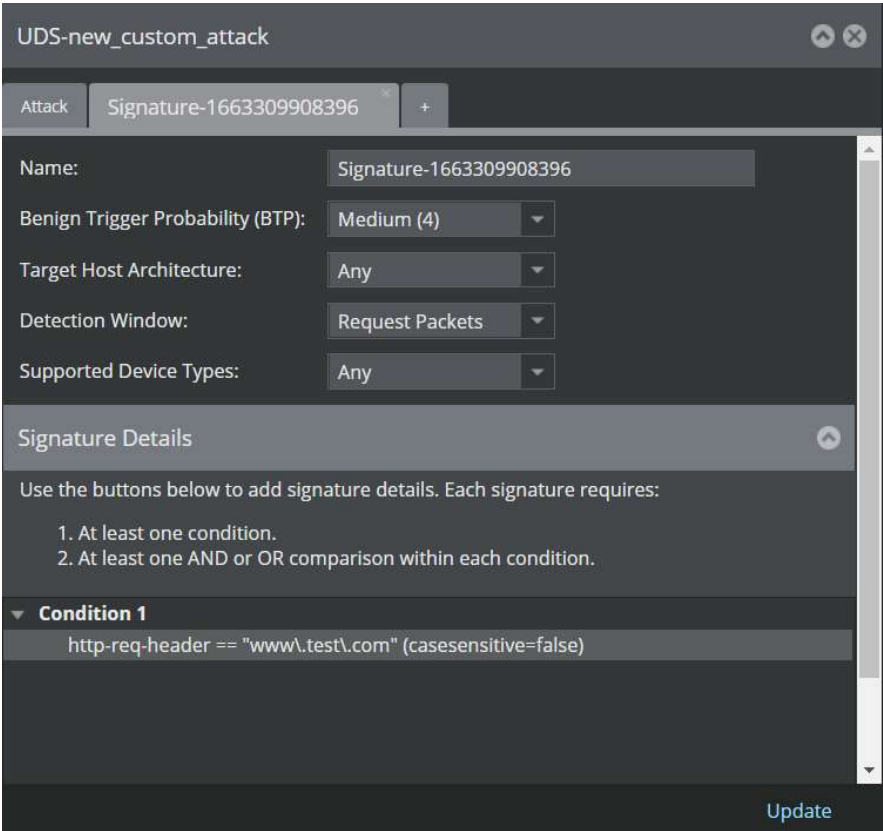


Figure 4-50 New Signature

- 2 (Optional) Clear the **Signature Name** and type a new name for your signature.
- 3 For this example, you can leave the **Benign Trigger Probability** and **Target Host Architecture** with the default values.
- 4 For this example select **Request Packets** as the **Detection Window**.
- 5 Based on the Sensor model that you plan to use for this example, select the **Target Device Type**.
- 6 Proceed to add the condition for your signature.

Tasks

- *Add the first condition and comparison for your signature on page 1664*
- *Add the second condition with OR comparisons on page 1666*

See also

*Signature creation interface on page 1519*

Add the first condition and comparison for your signature

To add the condition to the example signature:

Task

- 1 Condition 1 is added automatically with the URL.
- 2 Click Condition 1 so that it is highlighted.

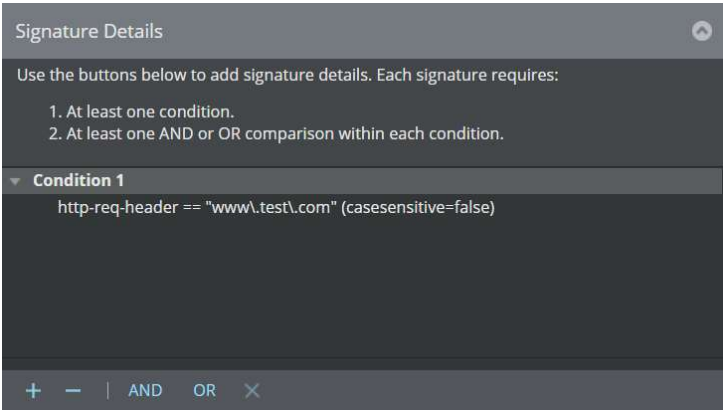


Figure 4-51 Adding Conditions

- 3 Click AND in the Signature Details section.  
The Add AND Comparison dialog opens.

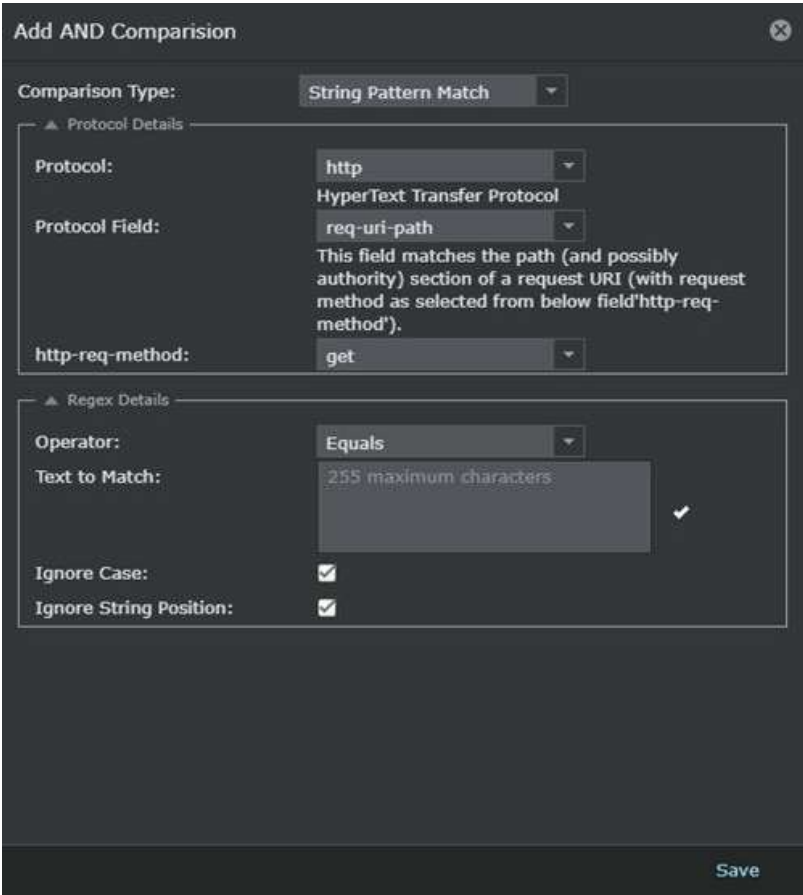


Figure 4-52 Configure Comparison window

- 4 For this example, select **String Pattern Match** from the **Comparison Type** drop-down list.
- 5 Select **http** from the Protocol List.  
Because you selected HTTP, the Custom Attack Editor displays the HTTP-specific protocol fields on the following screen.
- 6 Configure the fields for the comparison you have chosen.  
For this example, specify **req-uri-path** for the Protocol Field. This specifies that the Sensor should search in the URI of the request packet.
- 7 Select **get** as the http-request-method.
- 8 From the drop-down list in the **Regex Details** section, select the Operator as **Equals** which means that the comparison criteria has to be equal to the string pattern entered.
- 9 In the **Text to Match** text box, type the pattern to match using the Regular Expression Language rules.  
The pattern to match is "private.exe". You must add a backward slash (\) before the dot (.) to escape the dot character so that it is properly interpreted. The final string should appear as private\.exe.

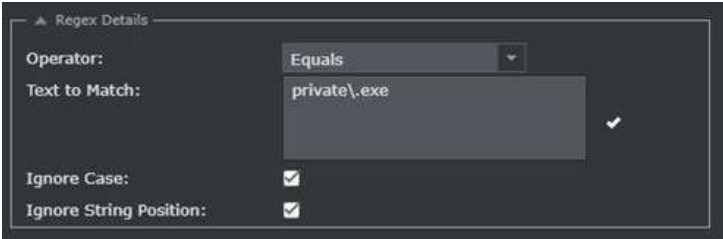


Figure 4-53 Adding the string pattern to match

- 10 Click ☒ to verify that your expression is valid, and that the appropriate pattern is represented.
- 11 Click **Save**.  
Your comparison appears under **Condition1**.

Add the second condition with OR comparisons  
To add the second condition to the example signature:

Task

- 1 In the **Signature Details** section, click **+** to add the condition.  
The first condition is minimized and the second condition reads as **[AND THEN] Condition 2**, thus signifying the first condition must match before the second condition can be tested.

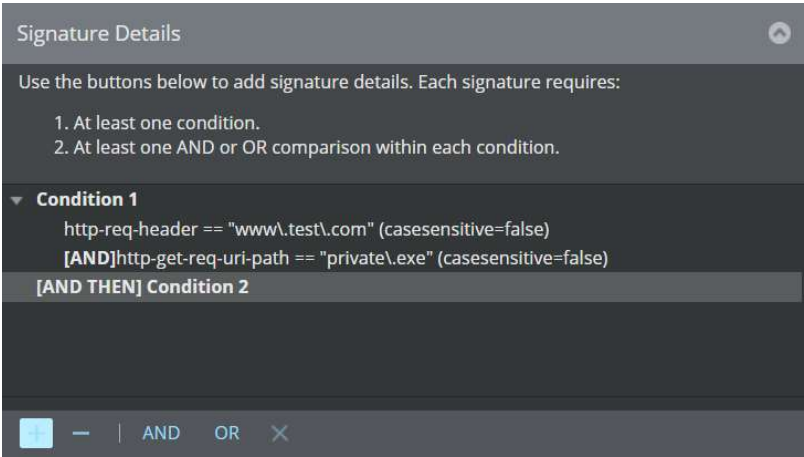


Figure 4-54 Adding conditions



- 2 Select the text of the second condition, click **OR**.
- 3 Select **Numeric Value Match** from the **Comparison Type** drop-down list.
- 4 Select **http** from the Protocol list.
- 5 Select **req-uri-length** for the Protocol Field, then **get** as the request method.
- 6 Select **Greater than** from the Operator drop-down list.

**Add AND Comparison**

Comparison Type: **Numeric Value Match**

**Protocol Details**

Protocol: **http**  
HyperText Transfer Protocol

Protocol Field: **req-uri-length**  
This field matches the URI length of a http request ( with request method as selected from below field'http-req-method').

http-req-method: **get**

**Numeric Details**

Operator: **Greater than**

Signed: ☒

Value to Match: **200**

**Save**

Figure 4-55 Configure Comparison window to select the comparator

- 7 (Optional) Check the **Signed** check box if you want the value to be signed.
- 8 Type a length value in the **Value to Match** field. For this example, type **200** (bytes) as the length over which this comparison will match.
- 9 Click **Save**; you are returned to the **New Signature** window.

- 10 Verify that your newly added comparison appears under **Condition 2** under the heading **[AND Then]**.

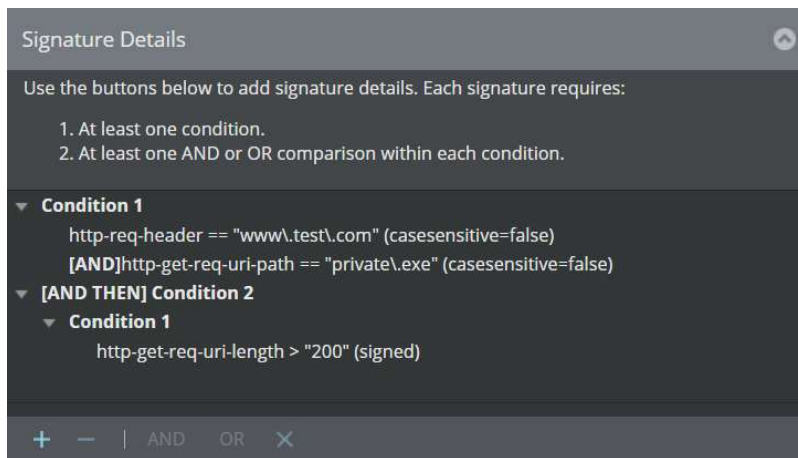


Figure 4-56 View the new condition defined

- 11 Click on **Condition 1**, then click **OR** under Comparisons.
- 12 Select **Numeric Value Match** from the Comparison List.
- 13 Select **http** from the Protocol.
- 14 Select **req-header-length** for the Protocol Field, select **content-length** as the http-req-hdr-type, then select **get** as the http-req-method.
- 15 Select **Greater than** from the Operator drop-down list.

16 Type a length value in the **Value** field. For this example, type **1** (byte) as the length over which this comparison will match.

 **Note**

As stated in the **Description** field, a value of 1 byte is significant for this comparison as the normal header length of a request should be zero.

Add OR Comparison

Comparison Type: Numeric Value Match

Protocol Details

Protocol:

http

HyperText Transfer Protocol

Protocol Field:

req-header-length

http-req-hdr-type:

content-length

http-req-method:

get

Numeric Details

Operator:

Greater than

Signed:

☒

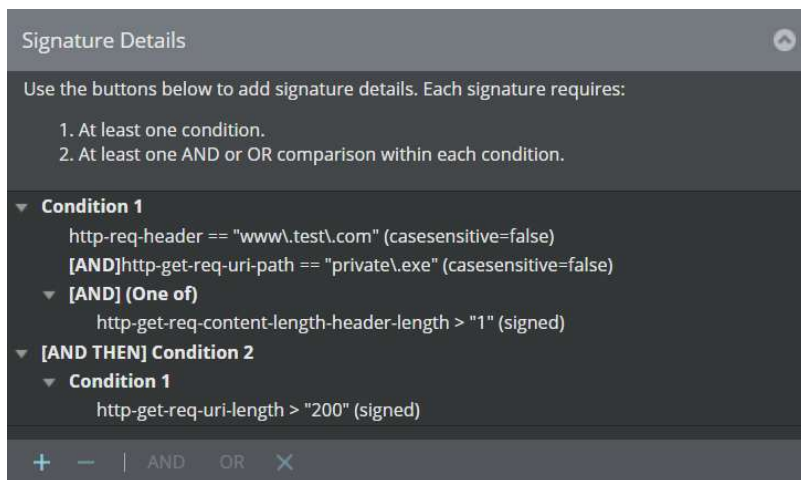
Value to Match:

1

Save

17 Click **Save**; you are returned to the **New Signature** window.

- 18 Verify that your newly added comparison appears beneath the first comparison you configured for **Condition 2**. The second comparison line is preceded by **[OR]** to signify the either-or relationship between the two comparisons.



- 19 Click **Update** in the signature details window.



#### Note

Though there are multiple conditions and comparisons, only one signature was created; so only one signature is uploaded to the Manager.

- 20 Verify that the attack definition is listed on the **Native Trellix IPS Format** tab.
- 21 Click **Save** to save the Trellix IPS Custom Attack in the Manager server database.
- 22 Make sure the Trellix IPS Custom Attack is saved in the database and also published in the policies.

See also

[Viewing a policy to verify inclusion of the attack on page 1604](#)

### Equivalent snort custom attack

It is not possible to create an equivalent Snort Custom Attack for all the conditions used in this example, because Snort does not support `http-get-req-header-length` option.

## Management of custom attacks from the Central Manager

Using the Trellix IPS Central Manager, you can manage custom attacks for the corresponding Managers. Functionally, the Custom Attack Editor of a Central Manager is the same as that of a Manager. So, see the earlier sections in this document for information on how to use the Editor for Trellix IPS Custom Attacks and Snort Custom Attacks.

The summary of managing custom attacks from the Central Manager is as follows:

- 1 Create the custom attack definitions using the Custom Attack Editor in the Central Manager.
- 2 Save the custom attacks in the Central Manager server database.

- 3 Synchronize the policies of the constituent Managers.
  - a From the Resource Tree of the Central Manager, select **Manager List | Policy Synchronization**.
  - b Select the Synchronization Type for the required Managers and then click **Synchronize**.
- 4 View the status of the synchronization. Check the **Status of Activities** section on the Home page of the Central Manager.
- 5 Update the Sensors with the configuration change.
  - a Log on to the required Manager.
  - b Update the relevant Sensors with the policy change.
  - c Repeat this process for the other Managers.

See also

[Managing Snort custom attacks on page 1587](#)

### Important notes

This section lists some important notes related to managing custom attacks in the Central Manager.

- You can create custom attack definitions for a Manager in the Central Manager. When you configure the Manager in the Central Manager the custom attack definitions are published to the Manager.
- You can create or edit custom attack definitions in a Manager.
- All custom attacks sent from the Central Manager to the Manager as part of synchronization are automatically saved in the Manager database and also published in the relevant policies.
- In a Manager, when an attack matching the custom attack definition is detected, you can view the alert in the **Attack Log** page of the Manager and Central Manager.
- In the Custom Attack Editor of the Manager, you cannot modify or delete any custom attack that was sent from the Central Manager. You cannot change the State of the Central Manager attack definitions in the Manager.
- When you synchronize the Central Manager with the constituent Managers, only the custom attacks with State as **Published** are sent to the Managers.
- If you remove a Manager from the Central Manager, all custom attacks sent from the Central Manager to the Manager are removed from the Manager database. To remove these attacks from the corresponding Sensors as well, you need to do a configuration update.
- Recall the custom attack capacity per Sensor type:
  - You can use up to 4500 Trellix IPS Custom Attacks and 5000 Snort Custom Attacks per NS-series Sensor.

Factor this in when you send custom attacks from the Central Manager to the corresponding Managers.

