**Summary: Examining a Suricata Signature**

**Overview of Suricata and Signature Files**

* Suricata is an **open-source signature-based IDS** that allows you to examine and create custom signatures.
* Suricata comes with pre-written **signature templates**, which are customizable for different protocols and services.

**Examining the Rules Folder**

1. Suricata's configuration files are located in the /etc/suricata directory on a Linux machine.
2. Inside the **rules folder**, you can find pre-written signature templates. These templates can be modified or new rules can be added.
3. The **custom.rules** file contains custom signatures, which can be viewed using the less command.

**Breaking Down a Suricata Signature**

1. **Signature Action**:
   * The action alert means an **alert** will be triggered if the signature conditions are met.
2. **Header**:
   * Specifies the protocol (HTTP), source and destination IP addresses, and ports.
   * For example, the signature will alert when HTTP traffic originates from the **home network** and goes to the **external network**.
3. **Rule Options** (enclosed in parentheses and separated by semicolons):
   * **Message**: The message that appears when an alert is triggered, such as "GET on wire".
   * **Flow**: Defines the direction of traffic. In this case, it matches established connections (successful connections).
   * **Content**: Specifies what to inspect in the packet. For this signature, it matches if the packet contains the text GET (an HTTP request).

**Summary of the Signature**

* The signature will trigger an **alert** if **HTTP traffic** containing the text GET is detected **leaving the home network** and **going to the external network**.

**Customizing and Testing Signatures**

* To be effective, IDS signatures must be **tailored** to fit the specific environment and **tested** to ensure they correctly detect threats while minimizing false positives.

**Summary: Suricata Logs and EVE JSON Format**

**Log Types in Suricata**

1. **Alert Logs**:
   * Generated when a **signature** is triggered, often in response to suspicious activity.
   * Contains relevant information for **security investigations**.
   * Includes **IP addresses**, **protocols**, and **signature details** like the **message** and **signature ID**.
2. **Network Telemetry Logs**:
   * Capture details of **network traffic flows** but aren't always related to security events.
   * For example, it may log events like a **connection to a specific port**.
   * These logs can help build the **context** for investigations.

**EVE JSON Format**

* **EVE JSON** (Extensible Event Format + JSON) is the format Suricata uses to output logs.
* **JSON** organizes data into **key-value pairs**, making it easy to search and extract information.

**Examples of Log Types**

1. **Alert Log Example**:
   * The **event type** is alert.
   * Contains details about the **activity** logged, such as IP addresses and protocol.
   * Also includes information about the **signature**, such as the message (e.g., related to malware detection).
2. **Network Telemetry Log Example**:
   * The **event type** is http.
   * Logs details about an **HTTP request**, such as:
     + **Hostname** (website accessed).
     + **User agent** (e.g., Mozilla 5.0 browser).
     + **Content type** (e.g., HTML text).

**Summary**

* Suricata logs data in **two main formats**: **alert logs** for security events and **network telemetry logs** for general network activity.
* Both types of logs are in the **EVE JSON format**, making it easier to process and analyze them.

Overview of Suricata

So far, you've learned about detection signatures and you were introduced to Suricata, an intrusion detection system (IDS).

In this reading, you’ll explore more about Suricata. You'll also learn about the value of writing customized signatures and configuration. This is an important skill to build in your cybersecurity career because you might be tasked with deploying and maintaining IDS tools.

**Introduction to Suricata**

[**Suricata**](https://suricata.io/) is an open-source intrusion detection system, intrusion prevention system, and network analysis tool.

**Suricata features**

There are three main ways Suricata can be used:

* **Intrusion detection system** (**IDS**): As a network-based IDS, Suricata can monitor network traffic and alert on suspicious activities and intrusions. Suricata can also be set up as a host-based IDS to monitor the system and network activities of a single host like a computer.
* **Intrusion prevention system** (**IPS**): Suricata can also function as an intrusion prevention system (IPS) to detect and block malicious activity and traffic. Running Suricata in IPS mode requires additional configuration such as enabling IPS mode.
* **Network security monitoring** (**NSM**):In this mode, Suricata helps keep networks safe by producing and saving relevant network logs. Suricata can analyze live network traffic, existing packet capture files, and create and save full or conditional packet captures. This can be useful for forensics, incident response, and for testing signatures. For example, you can trigger an alert and capture the live network traffic to generate traffic logs, which you can then analyze to refine detection signatures.

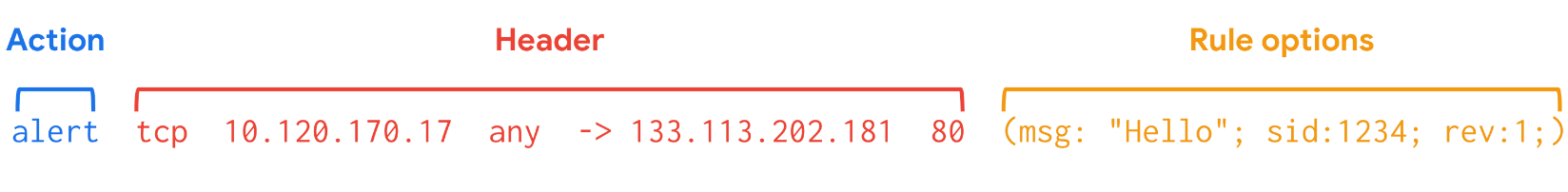
**Rules**

Rules or signatures are used to identify specific patterns, behavior, and conditions of network traffic that might indicate malicious activity. The terms rule and signature are often used interchangeably in Suricata. Security analysts use **signatures**, or patterns associated with malicious activity, to detect and alert on specific malicious activity. Rules can also be used to provide additional context and visibility into systems and networks, helping to identify potential security threats or vulnerabilities.

Suricata uses **signatures analysis**,which is a detection method used to find events of interest. Signatures consist of three components:

* **Action**: The first component of a signature. It describes the action to take if network or system activity matches the signature. Examples include: alert, pass, drop, or reject.
* **Header**: The header includes network traffic information like source and destination IP addresses, source and destination ports, protocol, and traffic direction.
* **Rule options:** The rule options provide you with different options to customize signatures.

Here's an example of a Suricata signature:



Rule options have a specific ordering and changing their order would change the meaning of the rule.

**Note**: The terms rule and signature are synonymous.

**Note:** Rule order refers to the order in which rules are evaluated by Suricata. Rules are processed in the order in which they are defined in the configuration file. However, Suricata processes rules in a different default order: pass, drop, reject, and alert. Rule order affects the final verdict of a packet especially when conflicting actions such as a drop rule and an alert rule both match on the same packet.

**Custom rules**

Although Suricata comes with pre-written rules, it is highly recommended that you modify or customize the existing rules to meet your specific security requirements.

There is no one-size-fits-all approach to creating and modifying rules. This is because each organization's IT infrastructure differs. Security teams must extensively test and modify detection signatures according to their needs.

Creating custom rules helps to tailor detection and monitoring. Custom rules help to minimize the amount of false positive alerts that security teams receive. It's important to develop the ability to write effective and customized signatures so that you can fully leverage the power of detection technologies.

**Configuration file**

Before detection tools are deployed and can begin monitoring systems and networks, you must properly configure their settings so that they know what to do. A **configuration file** is a file used to configure the settings of an application. Configuration files let you customize exactly how you want your IDS to interact with the rest of your environment.

Suricata's configuration file is **suricata.yaml**, which uses the YAML file format for syntax and structure.

**Log files**

There are two log files that Suricata generates when alerts are triggered:

* **eve.json**: The eve.json file is the standard Suricata log file. This file contains detailed information and metadata about the events and alerts generated by Suricata stored in JSON format. For example, events in this file contain a unique identifier called flow\_id  which is used to correlate related logs or alerts to a single network flow, making it easier to analyze network traffic. The eve.json file is used for more detailed analysis and is considered to be a better file format for log parsing and SIEM log ingestion.
* **fast.log**: The fast.log file is used to record minimal alert information including basic IP address and port details about the network traffic. The fast.log file is used for basic logging and alerting and is considered a legacy file format and is not suitable for incident response or threat hunting tasks.

The main difference between the eve.json file and the fast.log file is the level of detail that is recorded in each. The fast.log file records basic information, whereas the eve.json file contains additional verbose information.

**Key takeaways**

In this reading, you explored some of Suricata's features, rules syntax, and the importance of configuration. Understanding how to configure detection technologies and write effective rules will provide you with clear insight into the activity happening in an environment so that you can improve detection capability and network visibility. Go ahead and start practicing using Suricata in the upcoming activity!

**Resources for more information**

If you would like to learn more about Suricata including rule management and performance, check out the following resources:

* [Suricata user guide](https://suricata.readthedocs.io/en/latest/index.html)
* [Suricata features](https://suricata.io/features/)
* [Rule management](https://suricata.readthedocs.io/en/latest/rule-management/suricata-update.html)
* [Rule performance analysis](https://suricata.readthedocs.io/en/latest/configuration/suricata-yaml.html#engine-analysis-and-profiling)
* [Suricata threat hunting webinar](https://youtu.be/kaDGolhTu94)
* [Introduction to writing Suricata rules](https://youtu.be/tvoqFBVSShA)
* [Eve.json jq examples](https://suricata.readthedocs.io/en/latest/output/eve/eve-json-examplesjq.html)