SNORT AND SNORT RULES EXPLAINED

Bart Lenaerts-Bergmans - April 24, 2023

What is Snort?

Snort is an open-source network intrusion detection and prevention system (IDS/IPS) that monitors network traffic and identifies potentially malicious activities on Internet Protocol (IP) networks. Organizations can implement Snort using a rule-based language that combines protocol-, signature-, and anomaly-based inspection methods to detect malicious packets in network traffic and block potential attack vectors.

This mix is key. Commonly used signature-based methods are effective for identifying known threats, but they are not so great when it comes to unknown threats. Snort leverages behavior-based approaches as well to discover actual vulnerabilities by comparing network activity with a predefined set of Snort rules. This enables it to detect sophisticated emerging threats that may not have been previously identified through signature-based methods alone.

Snort Use Cases

Snort is primarily used by organizations looking for a platform-agnostic IDS/IPS to secure their networks against emerging threats. Companies turn to Snort for:

- Real-time network traffic monitoring
- Protocol analysis
- Content matching; collates rules by protocol, ports, and then by those with content and without
- Operating system (OS) fingerprinting
- Compatibility with any OS

But there's more to Snort than all that.

Packet Sniffing and Logging

Snort can serve as a packet sniffer that captures network traffic on a local network interface. You can also leverage Snort as a packet logger that writes captured packets to disk to debug network traffic. Or, use its network IDS/IPS capabilities to monitor network traffic in real time and examine each packet for suspicious activities or potentially malicious payloads.

Alerts and Rules

Snort can generate alerts for any unusual packets discovered in network traffic, based on the rules configured. This can help identify network threats or other risks that could lead to vulnerabilities being exploited.

The Snort rule language is very flexible, enabling you to create your own Snort rules to differentiate regular network activity from anomalous activity. This lets you add new procedures that instruct Snort to monitor the network for specific behavior and prevent potential attacks on the organization's network.

Attack Detection

Due to the flexibility of the Snort rule language and compatibility with all OSes, Snort is capable of detecting any network-based attack as long as there is a rule associated with the attack behavior. Below, we list a few types of breaches Snort can help organizations sniff out.

DoS (Denial-of-Service)/DDoS (Distributed-Denial-of-Service)

DoS/DDoS attacks involve flooding a network with illegitimate service requests to disrupt business operations. While a DoS attack is launched from a singular system, a DDoS attack is an orchestrated attack originating from multiple systems in multiple locations.

Buffer Overflow

A buffer overflow occurs when the attacker directs more incoming traffic to a network address so that the volume of network data exceeds all available bandwidth in the system.

Spoofing

GoSecure, Inc.
Exhibit 2004
CrowdStrike v. GoSecure
IPR2025-00068
Ex. 2004-001

Sometimes a hacker impersonates an authorized user or system to access a target network and steal information or perform malicious activities. This is called a spoofing attack.

Common Gateway Interface (CGI)

While CGI provides an interface between the web and the end user for rendering dynamic webpages, it is also known to contain security vulnerabilities that can be exploited by hackers. Web-based CGI scripts can frequently fall victim to input validation attacks due to not filtering malicious inputs.

Stealth Port Scans

Hackers often use stealth port scans, also known as half-open scans, to attack via open ports on the network without establishing a full connection. This involves sending a single packet via the Transmission Control Protocol (TCP) three-way handshake and terminating the process once a port is detected in the target network. This bypasses firewalls and makes the scan appear as normal network traffic.

How Does Snort Work?

Snort is based on the packet capture library (libpcap), a system-independent interface for capturing traffic that is widely used in network analyzers. Snort monitors network traffic and compares it against a Snort rule set defined by users in a config file. It applies these rules to packets in network traffic and issues alerts when it detects any anomalous activity.

Let's learn a bit more about how Snort operates and its rules.

Snort Modes

Snort can be configured to have one of three flags, which will determine its operating mode:

- **Sniffer mode (-v flag):** Snort reads the TCP/IP packets and prints out packet information on the console.
- **Packet logger mode (-l flag):** Snort logs incoming TCP/IP packets in a logging directory on disk for further analysis.
- **Network intrusion detection and prevention system (NIDS) mode (-c flag):** Snort determines whether to perform an action on network traffic based on the rule type specified in the config file.

Snort Rule Types

Snort does not evaluate rules in the order they appear in the config file; instead, it reviews them based on the rule type, which specifies the action to take when Snort finds a packet that matches the rule criteria.

The five basic rule types in Snort are:

- **Alert rules:** Snort generates an alert when a suspicious packet is detected.
- **Block rules:** Snort blocks the suspicious packet and all subsequent packets in the network flow.
- **Drop rules:** Snort drops the packet as soon as the alert is generated.
- **Logging rules:** Snort logs the packet as soon as the alert is generated.
- **Pass rules:** Snort ignores the suspicious packet and marks it as passed.

Understanding Snort Rules

It's important to properly write Snort rules so they work as intended; that is, so that they successfully identify emerging threats within your network. To do this, you need a clear grasp of Snort syntax and how the rules are formed.

While Snort rules are usually written in a single line, recent versions of Snort allow for multi-line rules; this is especially useful for more sophisticated rules that can be difficult to restrict to just one line.

Snort rules consist of two logical parts: a rule header and rule options. Let's explore.

Rule Headers

These define the action to take when any traffic that matches the rule is identified. A rule header consists of five main components:

- **Action to take:** The first component declared in a Snort rule
- **IP addresses:** The source and destination
- **Port numbers:** The source and destination
- **Direction of traffic:** -> for single direction from source to destination; <> for bidirectional
- **Inspection protocol:** "Layer 3" (IP and ICMP) and "Layer 4" (TCP and UDP) protocols

Note: Although Snort currently supports Layer 3 and 4, in Snort 3, you can also instruct Snort to only match rules to traffic for the given application-layer service (such as SSL/TLS and HTTP).

Rule Options

These define the network traffic criteria that need to be met in order for a rule to match, as well as the output when there is a match. Here are some of the rule options available:

- **Message to display when rule matches:** Explains the purpose of the rule
- **Flow state:** Specifies the session properties to check for a given packet
- **Content or pattern:** Specifies the content or pattern to find in a packet's payload or non-payload data

- **Service or application protocol:** Instructs Snort to identify the content or pattern either on the traffic of the specified application-layer service or the source and destination ports specified in the rule header
- **Snort ID (sid) and revision number (rev):** Uniquely identifies a Snort rule (sid) or uniquely identifies the revision number of a Snort rule (rev)

Example Snort Rules

Let's look at how you would write Snort rules for a DoS attack using Docker honeypots.

To uncover this attack mode, we need to write a rule to detect HTTP requests sent by a compromised Docker image over a TCP/IP connection:

```
1. alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS
```

DoS attacks are launched by flooding the server with HTTP GET requests over an established connection. So, we need to instruct Snort to check for this pattern in the rule options section:

```
1. # only detect established TCP connections from client requests<p>flow:to_server, established;
</p><p># match on HTTP GET requests</p><p>http_header; content:"Mozilla/5.0 (Windows NT
10.0|3B| Win64|3B| x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/98.0.4758.102
Safari/537.36";</p><p>http_method; content:"GET";</p>
```

As the attack mode is launched via honeypots, the rule is classified as a Network Trojan activity:

```
1. alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (<p>msg: "Detects DoS HTTP request sent by
compromised Docker image";</p><p>flow:to_server, established;</p><p>http_header;
content:"Mozilla/5.0 (Windows NT 10.0|3B| Win64|3B| x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/98.0.4758.102 Safari/537.36";</p><p>http_method; content:"GET";</p><p>classtype:trojan-
activity;</p><p>sid:8001951;</p><p>rev:20220420;</p><p>)</p>
```

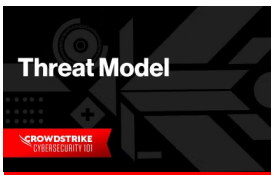
Leverage Expert-Written Snort Rules

Implementing Snort in your cybersecurity stack provides a flexible and platform-agnostic approach to securing your network against known and emerging network security threats. However, the rules must be configured to work properly.

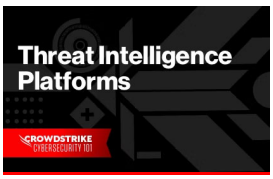
While you could write your own Snort rules for fairly straightforward use cases, keeping the rules up to date with emerging threats is a challenging task. Instead, consider using Snort and YARA rules created by experts, like the freely available Community ruleset or CrowdStrike Falcon Intelligence.

Bart is Senior Product Marketing Manager of Threat Intelligence at CrowdStrike and holds +20 years of experience in threat monitoring, detection and intelligence. After starting his career as a network security operations analyst at a Belgian financial organization, Bart moved to the US East Coast to join multiple cybersecurity companies including 3Com/Tippingpoint, RSA Security, Symantec, McAfee, Venafi and FireEye-Mandiant, holding both product management, as well as product marketing roles.

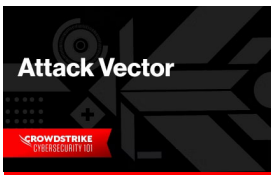
Featured Articles



Threat Model



Threat Intelligence Platforms



Attack Vectors

- Definition
- Use Cases
- How It Works
- Rule Components
- Example Rules

ur free trial now.

Total protection has never been easier. Take advantage of our free 15-day trial and explore the most popular solutions for your business:

- Protect against malware with next-gen antivirus.
- Get unrivaled visibility with USB device control.
- Defend against threats on your mobile devices.

Start free trial

CROWDSTRIKE

New to CrowdStrike?

About the Platform

Explore Products

Services

Why Choose CrowdStrike?

CrowdStrike Financial Services

Cyber Monday deals

Company

About CrowdStrike

Careers

Events

Newsroom

Partners

CrowdStrike Marketplace

Learn with CrowdStrike

2024 Global Threat Report

Cybersecurity 101

Your Threat Landscape

Tech Hub

View all resources

Contact us →

Experienced a breach? →

Copyright © 2025 Contact Us Privacy Cookies Your Privacy Choices Terms of Use Accessibility