Snort IDS/IPS Explained: What - Why you need - How it works

Cybersecurity is an important issue for both academics and practitioners, since successful cyberattacks may result in astronomical expenditures owing to the loss of confidentiality, integrity, or availability. Various security approaches have been proposed for detecting cyberattacks, with intrusion detection systems (IDS) and network-based intrusion detection systems (NIDS) being among the most prevalent.

The multitude of NIDS detection methods and approaches that have been developed are often classified as either anomaly-based (ANIDS) or signature-based (SNIDS). Anomaly-based systems evaluate the typical behavior of a system and emit alerts when the divergence from normal behavior reaches a certain threshold. Signature-based techniques check for patterns (signatures) in the studied data and provide alerts if they match known threats.

Signature-based techniques give excellent detection results for known, predefined threats. However, they are unable to identify new, unknown intrusions, even if they are minimal versions of previously identified threats.

One of the foremost signature-based intrusion detection and prevention systems is Snort in the cybersecurity world. Snort is an open-source intrusion prevention system that can analyze and log packets in real-time. Snort is the most extensively used IDS/IPS solution in the world, combining the advantages of signature, protocol, and anomaly-based inspection. With millions of downloads and approximately 400,000 registered users, Snort has become the industry standard for intrusion prevention systems (IPS).

In this article, we will cover what Snort is, what Snort is used for, what type of attacks Snort can detect, how it detects and prevents network intrusions, and how you can write a Snort rule. Lastly, we will discuss the differences between Snort and another packet sniffer, Wireshark, and the IPS tool, Suricata.

What is Snort?

Snort is a powerful and lightweight open-source network intrusion detection and preven system(IDS/IPS) developed in 1998 by Martin Roesch, the founder and former CTO of Sourcefire. Snort is currently being developed and maintained by Cisco, which acquired Sourcefire in 2013. Snort has been a pioneer in business intrusion prevention and detection software for a long time.

SNORT provides network traffic analysis and packet recording in real time. SNORT employs a rule-based language that integrates anomaly, protocol, and signature inspection techniques to identify potentially malicious behavior.

Network administrators can identify denial-of-service (DoS) and distributed DoS (DDoS) attacks, Common Gateway Interface (CGI) attacks, buffer overflows, and covert port scans using SNORT. SNORT generates a set of principles that define malicious network activity, identify malicious packets, and notify users of potential threats.

Snort is downloadable and configurable for both home and corporate usage. It can be compiled on the majority of Linux, Unix, and major BSD operating systems. Microsoft Windows versions of Snort are available as well.

Snort is built on the library packet capture(libpcap). For real-time traffic analysis, packet logging, content matching, and protocol analysis, Libpcap is a useful tool that sees widespread use in Transmission Control Protocol/Internet Protocol(TCP/IP) address traffic sniff, content searchers, and analyzers.

Is Snort a Sniffer?

Yes. Snort is a "network packet sniffer" that inspects network traffic and carefully examines each packet to find any suspicious irregularities or potentially harmful payloads. Furthermore, Snort is used not only as a packet sniffer similar to `tcpdump`, but it is also used as a packet logger that is useful for network traffic debugging and as a network intrusion detection and prevention system.

What are the Benefits of Using Snort in Your Environment?

The Snort network intrusion and detection system provides many benefits to organizations that deploy it on their networks. Detecting and preventing network security risks is the most significant advantage of Snort. Snort provides an early warning system that stops malicic attacks from propagating throughout the network and inflicting further damage. It evalu the computer resources and reports any abnormalities or anomalous tendencies. It detects known signatures or attack signatures and notifies administrators of unidentified risks. If Snort assists in preventing the problem from spreading until administrators can address it. The other primary advantages of Snort are as follows:

- **High Accuracy:** Since Snort is an open-source project, there is a constant effort to improve it and alter some of its features for greater accuracy. Several security teams enhance the software via the globally dispersed Snort Community.
- **High adaptability:** The ability to add new functionalities to Snort by accessing its source code gives Snort a significant advantage over its competitors. This method might enable Snort to manage any network security system.
- **Quick Response:** With its real-time protection mechanisms, Snort can safeguard the system from any new threats or malicious software. Cisco Talos Security Intelligence and Research Group (Talos) is one of the greatest features of Snort; they can detect brand-new attacks by upgrading Snort with new threats every hour.

Why is Snort popular?

Snort is a widely-used network intrusion detection system (IDS), because it is one of the best cyber threat hunting tools available in the cybersecurity world. A Snort is an efficient software for the real-time monitoring of network traffic. It examines every packet for potentially harmful payloads.

Another feature that makes Snort popular is that it is used for protocol analysis, content searches, and matching. And it can identify a variety of threats, like port scans, buffer overflows, etc.

Moreover, Snort has rich portability and compatibility. It is compatible with Windows, Linux, many UNIX, and all major BSD operating systems. Snort does not require you to recompile your kernel or install any software or hardware. Snort only demands that you have root capabilities to install and run it.

Snort is designed for use as a network IDS in the most traditional sense. It just compares network traffic to a set of criteria and then warns system administrators of suspect network behavior so that they may take the necessary measures.

Finally, Snort is open-source and free software. Therefore, any organization with a limited budget, like educational institutions, small and medium businesses, and even home use prefers Snort as an IDS/IPS solution.

What is the Purpose of Snort?

Snort is configurable to operate in three modes:

- **Sniffer mode** only reads the network packets and shows them in a continuous stream on the console.
- **Packet logger mode**, in which packets are logged to disk.
- **Network Intrusion Prevention Detection System (NIPDS) mode**, which conducts network traffic detection and analysis.

Snort scans and detects network packets if it is configured to function as a sniffer. These packets may also be logged to a disk file by Snort. To use Snort as a network packet sniffer, users must enable promiscuous mode on the host's network interface to monitor all network traffic on the local network interface. The monitored traffic is then written to the console.

Snort logs packets by copying the required network traffic to a disk file in packet logger mode configuration.

When Snort is configured in NIPDS mode, it detects and prevents your network from various types of cyberattacks, like denial of service, or SQL injection attacks. Snort continuously monitors network traffic and compares it to a Snort rule set specified by the user. The corresponding configuration file is named `/usr/local/etc/snort/snort.lua`. When Snort detects suspicious activity, it works as a firewall and delivers a real-time alarm to Syslog, a separate alert file, or a pop-up window. Intrusion detection and prevention is the most crucial function of Snort.

Multiple characteristics make Snort valuable for security teams to monitor their systems and identify malicious activities. Snort consists of the following features :

- **Packet Recording:** Snort's packet logger mode records packets to disk, enabling packet logging. In this mode, Snort records every packet in a hierarchical directory depending on the IP address of the host network.

- **Real-time Traffic Monitor:** Snort is used to monitor incoming and outgoing network traffic. When it detects potentially harmful packets or threats on Internet Protocol (IP) networks, it will inform users in real-time.

- **Rules Are Simple to Apply:** Snort rules are simple to establish and facilitate network monitoring and protection. Its rule language is also very adaptable, and establishing new rules is quite straightforward, allowing network administrators to distinguish betwe normal and harmful Internet traffic.

- **Content Matching:** Snort categorizes rules by protocol, such as IP and TCP, then by port, and finally by those with and without content. Content-based rules employ a multi-pattern matcher that improves efficiency, particularly for protocols such as Hypertext Transfer Protocol (HTTP). Rules without substance are constantly reviewed, which has a detrimental impact on performance.
- **OS Fingerprinting:** The premise that all systems have a unique TCP/IP stack is used in OS fingerprinting. Using this method, Snort is used to identify the OS platform employed by a network-accessing machine.
- **Protocol Analysis:** Snort is capable of doing protocol analysis, a network sniffing technique that collects data in protocol layers for further analysis. This allows the network administrator to inspect possibly harmful data packets in more detail, which is critical for protocols such as Transmission Control Protocol/Internet Protocol (TCP/IP) stack protocol definition.
- **Compatibility:** Snort is installed on all network settings and operating systems, including Linux and Windows.
- **Free:** As open-source software, Snort is accessible for free to anybody who wishes to use an IDS or IPS to monitor and secure their network.

Snort acts as a protective barrier for network systems and data by collecting and analyzing information on a network, as well as system and user behaviors, to identify possible attacks and security breaches from both inside and outside the business. Without a reliable IDS system, firms have a greater chance of falling prey to cybercrime attacks and humiliating, often costly data breaches.

What attacks can Snort detect?

Snort is used to identify the following probes and cyber attacks, but is not limited to:

- DoS/DDoS attacks
- Buffer overflow attacks
- Semantic URL attacks
- Common Gateway Interface(CGI) attacks
- Stealth port scans
- Routing attacks
- Spoofing attacks

- Server message block probes
- Efforts to get an operating system's fingerprint

Can Snort Detect Zero-day Attacks?

Yes. A research paper "Signature Based Intrusion Detection for Zero-Day Attacks: (Not) A Closed Chapter?" by Hannes Holm from the Royal Institute of Technology (KTH), Sweden shows that Snort is capable of detecting zero-day attacks. The widespread assertion that signature-based network intrusion detection systems (SNIDS) cannot identify zero-day attacks has not been confirmed. This study investigates this attribute by evaluating 356 severe attacks against the Snort, which have been set up with outdated official rules. 183 of these attacks are zero-day exploits, whereas 173 are theoretically known to the rule set. The findings of the investigation indicate that Snort can identify zero-day exploits with an average detection rate of 17%. The total detection percentage for theoretically known assaults is nonetheless higher (mean detection rate of 54%). The study then explores how zero-day vulnerabilities are found, how susceptible their signatures are to false positives, and how easily they may be circumvented. These analyses imply that a reasonable estimate of Snort's zero-day detection rate is 8.2%.

How Does Snort Detect Intrusion?

Snort monitors network traffic in real-time and analyzes it using the Misuse Detection Engine BASE. Snort analyzes the incoming and outgoing data of the packet with the signatures of the inputs specified in the rules.

Snort employs a rule-based language that integrates anomaly, protocol, and signature inspection techniques to identify possibly malicious behavior. After being downloaded and configured, Snort rules are provided in two distinct sets:

- **Snort Subscriber Ruleset:** Cisco Talos is responsible for developing, testing, and approval of the Snort Subscriber Ruleset. Users will have access to the "Snort Subscriber Rule Set" in the following ways:
    - *Subscribers:* Subscribers to the Snort Subscriber Ruleset will get the ruleset as they

  are made available to Cisco customers in real-time. This ruleset is also known by the names "VRT Ruleset" and "Talos Ruleset." Typically, this ruleset is updated on Tuesdays and Thursdays, although it may be altered at any moment to account for developing risks. At the time of publication, 12-month memberships begin at $29 for personal usage and $399 per sensor for commercial use. Personal subscriptions are restricted to students and home network users. Business subscriptions are ideal for businesses, government organizations, non-profits, colleges, etc. that need to install Snort on many devices and secure a large network.

  - *Registered users:* This ruleset is also available for free usage by individuals and organizations (however, Integrators may not use this ruleset). Under the "limited" clause of the Snort Subscriber Rule License, this ruleset is 30 days behind the Snort Subscriber Rule Set and does not include zero-day threats. The Community ruleset is included inside this rule book. If you are not a subscriber, it is advised that you use both the registered ruleset and the community ruleset. Typically, this ruleset is changed on Tuesdays and Thursdays.

- **Community Ruleset:** The Community Ruleset is created by the Snort community and its quality is assured by Cisco Talos. It is offered at no cost to all users and is licensed under the GPLv2. The authors of the community ruleset are identified in the AUTHORS file included inside the tarball. This rulebook is a subset of the subscription ruleset and is updated daily.

Cisco Systems updates freshly found attack patterns to these rulesets regularly. Through the `snort.org` website, you may obtain rules and implement them in your network. Customers with paid subscriptions get updates faster. Additionally, you may build your criteria to boost the system's detection capabilities.

Snort rules let the application execute a variety of operations for intrusion detection, including:

- **Monitor Network Traffic:** After traffic has been recorded, Snort is used to diagnose malicious packets and configuration problems.

- **Detect Network Anomalies:** Using Snort rules, network administrators may simply distinguish between normal, anticipated Internet traffic and anything unusual. Snort examines network traffic in real-time to detect harmful activities, and subsequently informs users.

- **Conduct Packet Sniffing:** Snort is used for packet sniffing, which is the collection of all data sent inside and outside a network. Collecting the individual packets that go between network devices offers a thorough examination of how traffic is transmitted.

- **Generate Alerts:** Snort notifies users according to the rule actions set in its configuration file. To get warnings, Snort rules must include criteria that describe when a packet should be regarded as odd or malicious, the risks of vulnerabilities being exploited, and whether the packet violates the organization's security policy or poses a network danger.
- **Create New Standards:** Snort facilitates the creation of new rules inside the program. This enables network administrators to modify how Snort conversion should function and the procedures it should execute. For instance, users build new rules that instruct Snort to avoid backdoor attacks, search for certain content in packets, display network statistics, choose which network to watch, and publish alarms to the console.

What are Snort Signatures?

Snort signature is any detection mechanism that depends on the presence of identifiable markings or features in exploits. Snort signatures are meant to identify known exploits because they include distinctive identifiers such as fixed offsets, ego strings, debugging information, or any other distinguishing identifier that is or is not associated with exploiting a vulnerability.

This sort of detection is often categorized as day after detection since genuine public exploits are required for it to function. This technique is used by antivirus businesses to safeguard their consumers against virus outbreaks. This sort of security is ineffective since the virus has already infected a user before signatures can be created.

How Do You Write Snort Rules?

Snort has a separate set of rules for each set of defined signatures, which are based on certain sorts of attacks. Snort rules must be on a single line. Unless the multi-line character `\` is used, multiple-line rules are not recognized by the Snort parser. Snort rules are contained in snort.conf configuration file. Snort rules have the following structure:

- **Header:** The header includes the action of the rule, the protocol, the source, and destination IP addresses, and the port. It specifies "who, where, and what to do" for the packages.
- **Options:** This section comprises warnings and information on which portions of the packages should be evaluated to determine whether the action rules should be implemented. This part is not expressly needed by every regulation.

There are five fundamental rule types in Snort:

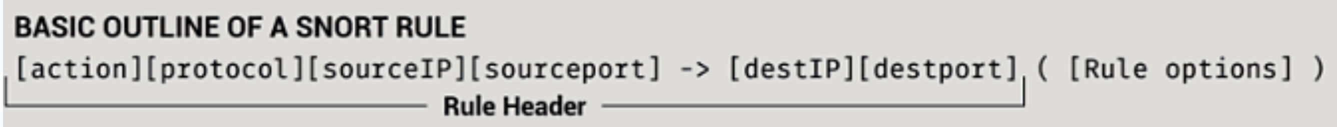
1. *Alert rules*:Snort generates an alert whenever it detects a suspect packet.
2. *Block rules*: Snort blocks the questionable packet and all packets that follow in the network flow.
3. *Drop rules*: Snort drops the packet as soon as the alert is generated, per the drop criteria.
4. *Logging rules*: Snort logs the packet immediately after an alert is generated.
5. *Pass rules*: Snort disregards and marks the dubious packet as passed.

A Snort rule *header* consists of the following parts:

- **Actions:** It represents the rule's action and may take the following values:
    - *Alert:* produce an alert using the chosen alert technique, then log the packet.
    - *Log:* log the packet
    - *Pass:* disregarded the packet
    - *Activate:* creates an alert when the action is triggered
    - *Dynamic:* stays dormant until an action activates it; then it functions as a log
- **Protocol:** It specifies the following items if the packet uses TCP, UDP, or ICMP:
- **IP addresses:** Snort lacks a method for providing hostname lookup for IP address fields in the rules file. This field indicates the source and destination IP addresses of the packet, as well as the CIDR block specifying the netmask used. Any address may be specified with the keyword "any". Specifically, CIDR/24 represents a Class C network, /16 a Class B network, and /32 a specific machine address. For example, the src_ip/mask combination 192.168.1.0/24 denotes the range of IP addresses beginning at 192.168.1.1 and ending at 192.168.1.255.
- **Port Number:** Specify the source and destination ports, respectively, in decimal format. The ":" operator may also be used to denote an interval. Regarding IP addresses and ports, the phrase "any" means that any value is acceptable.
- **Direction Operator:** It shows the travel direction (from source to destination), which may either be bidirectional (`<>`) or a single direction(`->`).

The overall Snort rule takes the following form:

```
action protocol source port -> destination port (options)
```



BASIC OUTLINE OF A SNORT RULE

```
[action][protocol][sourceIP][sourceport] -> [destIP][destport] ( [Rule options] )
```

Figure 1. *Basic Outline of a Snort Rule*

Rule options are the core of Snort's intrusion detection engine, combining versatility and simplicity of use.

All Snort rule options are separated from each other using a semicolon (;).

Rule option keywords and their arguments are separated from their arguments with a colon (:).

Snort rule *general options* are given below:

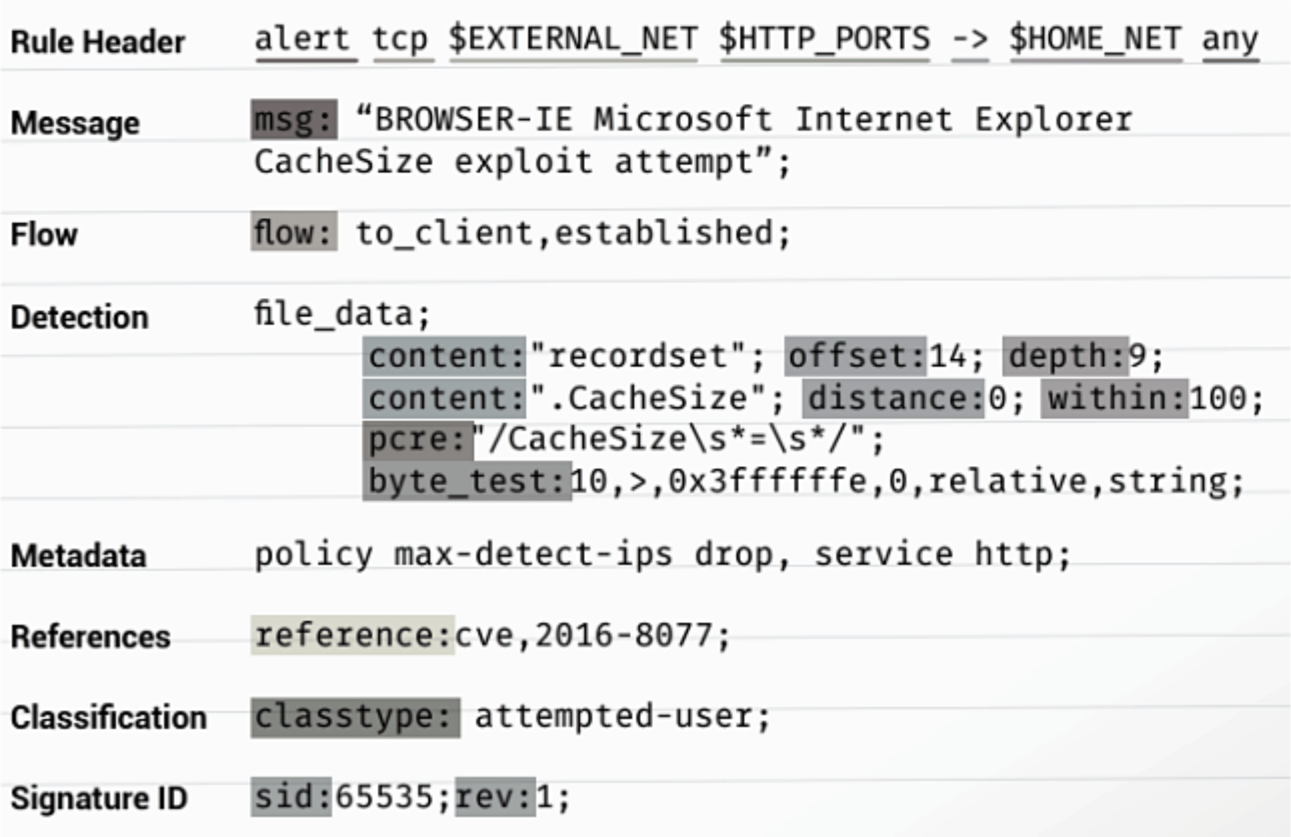
- **Message:** Typically, a relevant message provides the detection criteria of the rule. The message rule option instructs Snort on what to print when the rule is triggered. It is a basic string of words.
- **Flow:** Specifies the direction of network traffic for the rule to execute. Flow is used in combination with TCP stream reassembly. It permits restrictions to only apply to particular traffic flow directions.
- **Reference:** The reference keyword enables rules to include references to external information sources.
- **Classtype:** The `classtype` keyword is how Snort communicates the consequence of action in the event of a successful assault.
- **sid/rev:** Each rule's snort ID is a unique identifier. This data relates to facilitating the identification of rules by output plugins and should be used with the rev (revision) keyword

Notably, although most choices are optional, the SID (Snort ID) is essential and should not clash with another rule's SID. It is the distinctive identity assigned to each rule. Snort reserves SIDs from 0 to 1,000,000.

Snort rule *detection options* are given below:

- **Content:** This key feature enables the user to define rules that look for certain material inside the packet payload and trigger a response based on that data. This option data may include both text and binary information.

- **distance/offset:** These keywords provide the rule author with an indication of where to begin searching relative to the start of the payload or the start of a content match.
- **within/depth:** These keywords provide the rule author with an indication of how far ahead to search relative to the end of a prior content match, and how far back to search after that content match is discovered.
- **PCRE:** The PCRE keyword enables rules to be constructed using Perl-compatible regular expressions, enabling complicated matches as opposed to simple content matches.
- **Byte test:** The byte test option enables a rule to compare several bytes to a given binary value.



Rule Header	alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any
Message	msg: “BROWSER-IE Microsoft Internet Explorer CacheSize exploit attempt”;
Flow	flow: to_client,established;

Detection	file_data;
	content:"recordset"; offset:14; depth:9;
	content:".CacheSize"; distance:0; within:100;
	pcre:"/CacheSize\s*=\s*/";
	byte_test:10,>,0x3ffffffe,0,relative,string;

Metadata	policy max-detect-ips drop, service http;
References	reference:cve,2016-8077;
Classification	classtype: attempted-user;
Signature ID	sid:65535;rev:1;

Figure 2. *Snort rule structure*

Snort rules are not evaluated in the order in which they occur, in the snort configuration file. The default order is:

1. **Alert Rules:** It creates a notification using the alert method.
2. **Log Rules:** After creating an alert, the packet is then logged.
3. **Pass Rules:** The packet is ignored and dropped.

Some Snort rule examples are given below:

1. **Snort alerts, if the packet contains the word "Malware".**

```
alert ip any any -> any any (sid:1000001;msg:"Word Malware
found";content:"Malware";)
```

   - *alert:* this allows us to trigger an alert if a rule matches.
   - *ip:* this allows rules to be matched against any protocol (TCP, UDP, or ICMP) .
   - `any any -> any any`: this allows rules to be matched against any protocol. any source host and port to any destination host and port.
   - *Sid:1000001;msg: "Word Malware found":* the rule ID and alert message.

   The unique characteristic of this rule is the option content. As described in the Snort documentation, whenever a content option pattern match is conducted, the Boyer-Moore pattern match function is invoked and the (quite computationally intensive) test is run against the packet's content. We are just searching for Malware using a case-sensitive search.

2. **Snort notifies whenever a packet from any machine containing a UK national insurance number is transmitted to a web server.**

```
alert ip any any -> any http (sid:1000003;msg"UK national insurance
number found";pcre:"/([A-CEGHJ-PR-TW-Z][A-CEGHJ-NPR-TW-Z])(\s*[0-9])
{6}([A-D]|\s)/i";)
```

   This Snort rule example illustrates the usage of sets in PCRE. It is based on NIM391110's specs.

   - *([A-CEGHJ-PR-TW-Z][A-CEGHJ-NPR-TW-Z]):* This first capture group looks for a couple of characters. The first and second characters must not consist of the letters D, F, I, Q, U, or V. The second character is also constrained by not being O.
   - (\s[0-9]){6}: *Searches for a number between 0 and 9 six times, allowing zero to multiple spacing (s*). Due to the following group restriction being either a space or a letter, it is vital not to add an s* at the end of this rule. Otherwise, additional spaces after the rule might lead to improper matching.
   - *([A-D]|\s) :* A character or a space between A and D.
   - *i:* Flag insensitive, case-sensitive matching allowed (thus capital or lowercase characters for matching)

3. **Snort logs any UDP packet coming from everywhere to 10.10.10.0/24 network with**

destination port range 1-1024.

```
log udp any any -> 10.10.10.0/24 1:1024
```

What are the Differences Between Snort and Suricata?

Suricata has a notable history as an alternative to Snort, having debuted in beta form in 2009.

Suricata was launched to suit the requirements of contemporary infrastructure by Open Information Security Foundation (OISF). Suricata, like Snort, has IDS and IPS capabilities, as well as the ability to monitor large amounts of network traffic, automated protocol detection, a scripting language, and support for industry-standard output formats.

Additionally, Suricata supports multithreading, which theoretically enables the processing of more rules over faster networks with greater traffic volumes on the same hardware. One instance balances the processing load across every processor on a sensor-equipped to utilize Suricata, enabling commodity hardware to attain 10-gigabit rates without reducing ruleset coverage because Suricata is multithreaded. As of Snort 3, released in January 2021, Snort provides multithreading capability as well.

Suricata integrates with the scripting language Lua, which allows for more flexibility in creating rules. So that Suricata detects circumstances that would have been difficult or impossible to recognize with a conventional Snort Rule. This allows users to tailor Suricata to the complex threats often encountered by businesses.

The disadvantages of Suricata include that it is more difficult to deploy, and the community is smaller than Snort's. However, this is changing.

Snort makes it exceedingly easy to develop Snort rules that identify emerging threats using fresh threat intelligence.

According to the Snort website, "unlike signatures, rules are predicated on identifying the real vulnerability, not an exploit or a unique piece of data. Developing a rule requires an in-depth understanding of how the vulnerability truly operates."

Snort has always had strong community participation, which has resulted in a robust ruleset that is often updated. The syntax of the rules is relatively straightforward, and the struct[illegible] the software enables anybody to deploy customized rules into their IDS or share them w[illegible]e community.

Some commercial entities also generate SNORT rules, which are available for a monthly or yearly cost. Talos' SO/VRT rules (made available for free after one month) and CrowdStrike's Threat Intelligence Services are two examples.

Suricata is able to use the same rules as SNORT. Many, but not all, VRT regulations remain in effect. Emerging Threats is Suricata's own ruleset, which was previously only available to paying members but is now publicly accessible after 30 to 60 days. These Suricata rules make greater use of Suricata's additional capabilities, such as port-independent protocol identification and automated file discovery and extraction.

The widespread deployment of Snort's technology facilitates the development of rapid countermeasures to new threats. A day after the announcement of the Equifax hack, for instance, a Snort Rule was ready to monitor for the vulnerability at its core.

Snort introduced OpenAppID with version 2.9.7 in 2014. OpenAppID allows Layer 7 Detectors to identify apps. Despite the fact that the presence of a recognized program is not necessarily a direct security issue (such as Dropbox use), it enables better knowledge of what exists inside the network. By associating an AppID with a conventional SNORT IDS/IPS rule, not only may previously unknown apps be discovered but their traffic can also be rejected or flagged.

Suricata operates somewhat differently in this environment. It provides Application-Layer detection rules and can recognize HTTP or SSH traffic on non-standard ports depending on protocols, for instance. Additionally, protocol-specific log settings will be applied to these detections.

Age is the source of Snort's disadvantage. Snort is twenty years old and was built to operate on aging infrastructure. Although writing rules is relatively simple, adapting them to more sophisticated threats and the needs of high-speed networks has become difficult. Specifically, IPv6 has caused issues. Thanks to Snort 3, released in January 2021, it has support for multiple threads for packet processing, which frees up more RAM for packet processing.

Lastly, Suricate has a file extraction feature that doesn't exist on Snort. Suricata enables the extraction of files. This very handy feature enables the automated extraction of chosen files when a rule containing the "filestore" option is activated. It is possible, for instance, to extract all.pdf and.png files with a single pixel and save them in a predefined folder for additional manual inspection, VirusTotal queries, or even automatic sandboxing.

What is the Difference Between Snort and

Wireshark?

Wireshark once referred to as Ethereal, is the most popular network protocol analyzer. It operates on all platforms, including UNIX, Linux, OS X, and, Windows, and has an efficient function. Regular Wireshark users include security specialists, network professionals, developers, and educators. It is an open-source packet sniffer tool, similar to Snort. A global team of protocol specialists designed and maintains this application.

Wireshark is a free packet sniffer that has capabilities like analysis, network troubleshooting, etc. It is used to view, capture, and analyze data packets. WireShark is a very useful tool for network managers in troubleshooting computer networks. It can collect data from the LAN/WLAN and decrypt it into that format, allowing administrators to track down the reasons for poor performance and sporadic connectivity. On the other hand, Snort is used for network intrusion detection and prevention by using specified rules and informing the network security team through alert messages. Wireshark doesn't have any mechanism to detect anomalies on the network automatically and alert the administrators.

What is the History of Snort?

Martin Roesch created the first version of Snort in 1998. In 2001, he created a technological startup called Sourcefire. He took on the role of Chief Technology Officer at the company he founded.

It was marketed as a lightweight intrusion detection system that mainly operated on Unix and Unix-like operating systems. Now, it includes more features and can hardly be called lightweight. Written in the C programming language, Snort gained popularity rapidly as security experts were attracted to its configuration granularity. Snort's adoption was deemed cutting-edge. It soon became the de facto standard for network intrusion detection systems.

Check Point Software intended to purchase Sourcefire for $225 million in 2005, but dropped its bid when it became apparent that US authorities would seek to prohibit the deal.

The firm completed its first public offering in March 2007, generating $86.3 million. Sourcefire's acquisition of Clam Antivirus occurred in August of the same year. In May 2008, Sourcefire rejected a $187 million offer from security appliance provider Barracuda Networks, which had proposed to pay US$7.50 per share, a 13% premium over its then-current stock price.

Sourcefire got the 2009 "Reader Trust" award from SC Magazine for the best intrusion detection and prevention system (IDS/IPS) for Snort. In the 2012 Gartner Magic Quadrant competition for intrusion detection and prevention system appliances, the business was ranked in the "Leaders" Quadrant. NSS Labs recommended Sourcefire in 2012 for having the quickest and most accurate IPS detection.

2013 marked the beginning of a new era for Snort and Sourcefire in general since they were now owned by Cisco Systems for 2.7 billion. Multiple versions of Snort were published, and

designed to maximize efficiency while minimizing inaccuracy.

Last updated on ***Sep 16, 2023*** *by* ***Zenarmor***