Table of Contents

1 About
   1.1 Security Onion ............................................. 1
   1.2 Security Onion Solutions, LLC .......................... 2
   1.3 Documentation ............................................ 2

2 Introduction .................................................... 5
   2.1 Overview .................................................. 7
   2.2 Analysis Tools ........................................... 8
   2.3 Deployment Scenarios .................................... 13
   2.4 Conclusion ............................................... 13

3 Getting Started ................................................ 15
   3.1 Architecture .............................................. 16
   3.2 Hardware Requirements .................................. 23
   3.3 Partitioning ............................................... 28
   3.4 Release Notes ............................................. 30
   3.5 Download .................................................. 37
   3.6 VMWare ..................................................... 37
   3.7 VirtualBox ................................................ 39
   3.8 Booting Issues .......................................... 39
   3.9 Installation .............................................. 40
   3.10 Configuration ........................................... 41
   3.11 After Installation ....................................... 47

4 Security Onion Console (SOC) ................................ 49
   4.1 Alerts ...................................................... 51
   4.2 Hunt ...................................................... 55
   4.3 PCAP ....................................................... 59
   4.4 Kibana ..................................................... 61
   4.5 Grafana ................................................... 64
   4.6 CyberChef ................................................ 67
   4.7 Playbook ................................................ 68
   4.8 Fleet ....................................................... 72
   4.9 TheHive ................................................... 73
   4.10 ATT&CK Navigator ...................................... 74

5 Analyst VM ....................................................... 77
1.1 Security Onion

Security Onion is a free and open source Linux distribution for threat hunting, enterprise security monitoring, and log management. It includes TheHive, Playbook and Sigma, Fleet and osquery, CyberChef, Elasticsearch, Logstash, Kibana, Suricata, Zeek, Wazuh, and many other security tools. Security Onion has been downloaded over 1 million times and is being used by security teams around the world to monitor and defend their enterprises. Our easy-to-use Setup wizard allows you to build an army of distributed sensors for your enterprise in minutes!

Note: Security Onion started in 2008 and was originally based on the Ubuntu Linux distribution. Throughout the years, the Security Onion version tracked the version of Ubuntu it was based on. For example, the last major version of Security Onion was based on Ubuntu 16.04 and so it was called Security Onion 16.04. Security Onion is now container based and thus no longer limited to just Ubuntu. To signify this change, Security Onion now has its own versioning scheme and this new platform is Security Onion 2.

Here are some high level system differences between Security Onion 2 and the older legacy versions:

- Move from Ubuntu packages to containers
- Support both CentOS 7 and Ubuntu 18.04
- Change pcap collection tool from netsniff-ng to Google Stenographer
- Upgrade to Elastic Stack 7.x and support the Elastic Common Schema (ECS)
- Remove unsigned kernel module PF_RING and completely replace with AF_PACKET
- Suricata completely replaces Snort. (We may elect to add Snort back after Snort 3.0 is officially released.)
- Sgui, Squert, and capME are removed
- Storage Nodes are now known as Search Nodes
- Incorporate new tech: TheHive, Strelka, support for Sigma rules, Grafana/influx (independent health monitoring/alerting), Fleet (osquery management), Playbook (detection playbook tool), Onion Hunt (hunting tool), Security Onion Console (PCAP collection tool)
For more information about Security Onion not contained in this Documentation, please see our community site at https://securityonion.net.

1.2 Security Onion Solutions, LLC

Doug Burks started Security Onion as a free and open source project in 2008 and then founded Security Onion Solutions, LLC in 2014.

**Important:** Security Onion Solutions, LLC is the only official provider of hardware appliances, training, and professional services for Security Onion.

For more information about these products and services, please see our company site at https://securityonionsolutions.com.

1.3 Documentation

**Warning:** We’ve started updating this documentation for Security Onion 2. However, please note that this is a work in progress. Many pages have not been updated yet and thus may have incorrect or missing information.

1.3.1 Formats

This documentation is published online at https://securityonion.net/docs/2.3. If you are viewing an offline version of this documentation but have Internet access, you might want to switch to the online version at https://securityonion.net/docs/2.3 to see the latest version.

This documentation is also available in PDF format at https://readthedocs.org/projects/securityonion/downloads/pdf/2.3/.

Many folks have asked for a printed version of our documentation. Whether you work on airgapped networks or simply want a portable reference that doesn’t require an Internet connection or batteries, this is what you’ve been asking for. Thanks to Richard Bejtlich for writing the inspiring foreword! Proceeds go to the Rural Technology Fund! Please note that the print version is currently based on the older Security Onion 16.04 documentation.

https://securityonion.net/book

1.3.2 Authors

Security Onion Solutions is the primary author and maintainer of this documentation. Some content has been contributed by members of our community. Thanks to all the folks who have contributed to this documentation over the years!

1.3.3 Contributing

We welcome your contributions to our documentation! We will review any suggestions and apply them if appropriate.
If you are accessing the online version of the documentation and notice that a particular page has incorrect information, you can submit corrections by clicking the Edit on GitHub button in the upper right corner of each page.

To submit a new page, you can submit a pull request (PR) to the 2.3 branch of the securityonion-docs repo at https://github.com/Security-Onion-Solutions/securityonion-docs.

Pages are written in RST format and you can find several RST guides on the Internet including https://thomas-cokelaer.info/tutorials/sphinx/rest_syntax.html.

### 1.3.4 Naming Convention

Our goal is to allow you to easily guess and type the URL of the documentation you want to go to.

For example, if you want to read more about Suricata, you can type the following into your browser: https://securityonion.net/docs/suricata.

To achieve this goal, new documentation pages should use the following naming convention:

- all lowercase
- .rst file extension
- ideally, the name of the page should be one simple word (for example: suricata.rst)
- try to avoid symbols if possible
- if symbols are required, use hyphens (NOT underscores)
Network Security Monitoring (NSM) is, put simply, monitoring your network for security related events. It might be proactive, when used to identify vulnerabilities or expiring SSL certificates, or it might be reactive, such as in incident response and network forensics. Whether you’re tracking an adversary or trying to keep malware at bay, NSM provides context, intelligence and situational awareness of your network. Enterprise Security Monitoring (ESM) takes NSM to the next level and includes endpoint visibility and other telemetry from your enterprise. There are some commercial solutions that get close to what Security Onion provides, but very few contain the vast capabilities of Security Onion in one package.

In the diagram below, we see Security Onion in a traditional enterprise network with a firewall, workstations, and servers. You can use Security Onion to monitor north/south traffic to detect an adversary entering an environment, establishing command-and-control (C2), or perhaps data exfiltration. You’ll probably also want to monitor east/west traffic to detect lateral movement. As more and more of our network traffic becomes encrypted, it’s important to fill in those blind spots with additional visibility in the form of endpoint telemetry. Security Onion can consume logs from your servers and workstations so that you can then hunt across all of your network and host logs at the same time.
Many assume NSM is a solution they can buy to fill a gap; purchase and deploy solution XYZ and problem solved. The belief that you can buy an NSM denies the fact that the most important word in the NSM acronym is “M” for Monitoring. Data can be collected and analyzed, but not all malicious activity looks malicious at first glance. While automation and correlation can enhance intelligence and assist in the process of sorting through false positives and malicious indicators, there is no replacement for human intelligence and awareness. I don’t want to disillusion you. Security Onion isn’t a silver bullet that you can setup, walk away from and feel safe. Nothing is and if that’s what you’re looking for you’ll never find it. Security Onion will provide visibility into your network traffic and context around alerts and anomalous events, but it requires a commitment from you the defender to review alerts, monitor the network activity, and most importantly, have a willingness, passion, and desire to learn.

2.1 Overview

Security Onion seamlessly weaves together three core functions:

- full packet capture
- network and endpoint detection
- powerful analysis tools

**Full-packet capture** is accomplished via Stenographer. Stenographer captures all the network traffic your Security Onion sensors see and stores as much of it as your storage solution will hold (it has a built-in mechanism to purge old data before your disks fill to capacity). Full packet capture is like a video camera for your network, but better because not only can it tell us who came and went, but also exactly where they went and what they brought or took with them (exploit payloads, phishing emails, file exfiltration). It’s a crime scene recorder that can tell us a lot about the victim and the white chalk outline of a compromised host on the ground. There is certainly valuable evidence to be found on the victim’s body, but evidence at the host can be destroyed or manipulated; the camera doesn’t lie, is hard to deceive, and can capture a bullet in transit.

**Network and endpoint detection** analyzes network traffic or host systems, respectively, and provide log and alert data for detected events and activity. Security Onion provides multiple options:

- Rule-driven NIDS. For rule-driven network intrusion detection, Security Onion 2 uses Suricata. Rule-based systems look at network traffic for fingerprints and identifiers that match known malicious, anomalous or otherwise suspicious traffic. You might say that they’re akin to antivirus signatures for the network, but they’re a bit deeper and more flexible than that.

- Protocol metadata. For analysis-driven network intrusion detection, Security Onion offers Zeek. Unlike rule-based systems that look for needles in the haystack of data, Zeek says, “Here’s all your data and this is what I’ve seen. Do with it what you will and here’s a framework so you can.” Zeek monitors network activity and logs any connections, DNS requests, detected network services and software, SSL certificates, and HTTP, FTP, IRC, SMTP, SSH, SSL, and Syslog activity that it sees, providing a real depth and visibility into the context of data and events on your network. Additionally, Zeek includes analyzers for many common protocols and by default has the capacity to check MD5 sums for HTTP file downloads against Team Cymru’s Malware Hash Registry project. Beyond logging activity and traffic analyzers, the Zeek framework provides a very extensible way to analyze network data in real time. The input framework allows you to feed data into Zeek, which can be scripted, for example, to read a comma delimited file of C-level employee usernames and correlate that against other activity, such as when they download an executable file from the Internet. The file analysis framework provides protocol independent file analysis, allowing you to capture files as they pass through your network and automatically pass them to a sandbox or a file share for antivirus scanning. The flexibility of Zeek makes it an incredibly powerful ally in your defense.

- For endpoint detection, Security Onion offers Wazuh, a free, open source HIDS for Windows, Linux and Mac OS X. When you add the Wazuh agent to endpoints on your network, you gain invaluable visibility from endpoint to your network’s exit point. Wazuh performs log analysis, file integrity checking, policy monitoring, rootkit detection, real-time alerting and active response. As an analyst, being able to correlate host-based events with
network-based events can be the difference in identifying a successful attack. A new addition to Security Onion 2 is osquery, which is another free and open source endpoint agent. In addition, Security Onion can collect data via Syslog or other agent transport like Beats.

2.2 Analysis Tools

With full packet capture, IDS alerts, Zeek data, and endpoint telemetry, there is an incredible amount of data available at your fingertips. Fortunately, Security Onion tightly integrates the following tools to help make sense of this data.

2.2.1 Security Onion Console (SOC)

Security Onion Console (SOC) is the first thing you see when you log into Security Onion. It includes a new Alerts interface which allows you to see all of your NIDS and HIDS alerts.

Security Onion Console (SOC) also includes a new Hunt interface for threat hunting which allows you to query not only your NIDS/HIDS alerts but also Zeek logs and system logs.
Security Onion Console (SOC) also includes an interface for full packet capture (PCAP) retrieval.
2.2.2 TheHive

TheHive is the case management interface. As you are working in Alerts, Hunt, or Kibana, you may find alerts or logs that are interesting enough to send to TheHive and create a case. Other analysts can collaborate with you as you work to close that case.
2.2.3 Kibana

*Kibana*, created by the team at Elastic, allows us to quickly analyze and pivot between all of the different data types generated by Security Onion through a “single pane of glass”. This includes not only NIDS/HIDS alerts, but also Zeek logs and system logs collected via syslog or other agent transport. Kibana can pivot to full packet capture via *Security Onion Console (SOC)*.
2.2.4 CyberChef

*CyberChef* allows you to decode, decompress, and analyze artifacts.
2.2.5 Playbook

Playbook is a web application that allows you to create a Detection Playbook, which itself consists of individual plays. These plays are fully self-contained and describe the different aspects around the particular detection strategy.

2.3 Deployment Scenarios

Analysts around the world are using Security Onion today for many different architectures. The Security Onion Setup wizard allows you to easily configure the best installation scenario to suit your needs.

2.4 Conclusion

So we have full packet capture, Suricata rule-driven intrusion detection, Zeek event-driven intrusion detection and Wazuh host-based intrusion detection, all running out of the box once you run Security Onion setup. These disparate systems with various dependencies and complexities all run seamlessly together and would otherwise take hours, days or weeks to assemble and integrate on their own. What was once a seemingly impossible task is now as easy as answering a few questions.
Getting Started

If you’re ready to get started with Security Onion, you may have questions like:

**How many machines do I need?**

Depending on what you’re trying to do, you may need anywhere from one machine to thousands of machines. The *Architecture* section will help you decide.

**What kind of hardware does each of those machines need?**

This could be anything from a small virtual machine to a large rack mount server with lots of CPU cores, lots of RAM, and lots of storage. The *Hardware Requirements* section provides further details.

**Which ISO image should I download?**

You can download our Security Onion ISO image or a standard CentOS 7 or Ubuntu 18.04 ISO image. We recommend our Security Onion ISO image for most use cases, but you should review the *Partitioning*, *Release Notes*, and *Download* sections for more information.

**If I just want to try Security Onion in a virtual machine, how do I create a virtual machine?**

See the *VMWare* and *VirtualBox* sections.

**What if I have trouble booting the ISO image?**

Check out the *Booting Issues* section.

**Once I’ve booted the ISO image, how do I install it?**

The *Installation* section has steps for our Security Onion ISO image and for standard CentOS 7 and Ubuntu 18.04 ISO images.

**After installation, how do I configure Security Onion?**

The *Configuration* section covers many different use cases.

**Is there anything I need to do after configuration?**

See the *After Installation* section.
3.1 Architecture

If you’re going to deploy Security Onion, you should first decide on what type of deployment you want. This could be anything from a temporary Evaluation installation in a small virtual machine on your personal laptop all the way to a large scalable enterprise deployment consisting of a manager node, multiple search nodes, and lots of forward nodes. This section will discuss what those different deployment types look like from an architecture perspective.

3.1.1 Import

The simplest architecture is an Import node. An import node is a single standalone box that runs just enough components to be able to import a pcap using `so-import-pcap`. When you run `so-import-pcap`, it analyzes the pcap using Suricata and Zeek and the resulting logs are picked up by Filebeat and sent to Elasticsearch where they are parsed and indexed. You can then view those logs in Security Onion Console (SOC).

3.1.2 Evaluation

The next architecture is Evaluation. It’s a little more complicated than Import because it has a network interface dedicated to sniffing live traffic from a TAP or span port. Processes monitor the traffic on that sniffing interface and generate logs. Filebeat collects those logs and sends them directly to Elasticsearch where they are parsed and indexed. Evaluation mode is designed for quick installations to temporarily test out Security Onion. It is not designed for production usage at all.
3.1.3 Standalone

Standalone is similar to Evaluation in that all components run on one box. However, instead of Filebeat sending logs directly to Elasticsearch, it sends them to Logstash, which sends them to Redis for queuing. A second Logstash pipeline pulls the logs out of Redis and sends them to Elasticsearch, where they are parsed and indexed.

This type of deployment is typically used for testing, labs, POCs, or very low-throughput environments. It’s not as
scalable as a distributed deployment.
3.1.4 Distributed

A standard distributed deployment includes a **manager node**, one or more **forward nodes** running network sensor components, and one or more **search nodes** running Elastic search components. This architecture may cost more upfront, but it provides for greater scalability and performance, as you can simply add more nodes to handle more traffic or log sources.

- Recommended deployment type
- Consists of a manager node, one or more forward nodes, and one or more search nodes.
There is the option to utilize only two node types – the **manager node** and one or more **heavy nodes**, however, this is not recommended due to performance reasons, and should only be used for testing purposes or in low-throughput environments.

- Recommended only if a standard distributed deployment is not possible.
• Consists of a manager node and one or more heavy nodes.

3.1.5 Node Types

Management

The manager node runs its own local copy of Elasticsearch, which manages cross-cluster search configuration for the deployment. This includes configuration for heavy nodes and search nodes (where applicable), but not forward nodes, as they do not run Elastic Stack components. An analyst connects to the server from a client workstation (typically a Security Onion virtual machine installation) to execute queries and retrieve data.

The manager node runs the following components:

• Elasticsearch
• Logstash

3.1. Architecture
Forward Node

When using a forward node, Elastic Stack components are not installed. Filebeat forwards all logs to Logstash on the manager node, where they are stored in Elasticsearch on the manager node or a search node (if the manager node has been configured to use a search node). From there, the data can be queried through the use of cross-cluster search.

Forward Nodes run the following components:

- Zeek
- Suricata
- Stenographer
- Wazuh

Search Node

When using a search node, Security Onion implements distributed deployments using Elasticsearch’s cross cluster search. When you run Setup and choose Search Node, it will create a local Elasticsearch instance and then configure the manager node to query that instance. This is done by updating _cluster/settings on the manager node so that it will query the local Elasticsearch instance.

Search nodes primarily collect logs from other nodes and store them for searching.

Search Nodes run the following components:

- Elasticsearch
- Logstash
- Curator
- Wazuh

Heavy Node

Similar to search nodes, heavy nodes extend the storage and processing capabilities of the manager node. However, heavy nodes also perform sensor duties and thus have lower performance overall.

Heavy Nodes run the following components:

- Elasticsearch
- Logstash
- Curator
- Zeek
- Suricata
- Stenographer
Fleet Standalone Node

A Fleet Standalone Node is ideal when there are a large amount of osquery endpoints deployed. It reduces the amount of overhead on the manager node by transferring the workload associated with managing osquery endpoints to a dedicated system. It is also useful for off-network osquery endpoints that do not have remote access to the Manager node as it can be deployed to the DMZ and TCP/8090 made accessible to your off-network osquery endpoints.

If the Manager Node was originally setup with Fleet, your grid will automatically switch over to using the Fleet Standalone Node instead as a grid can only have one Fleet instance active at a time.

Fleet Standalone Nodes run the following components:

- Fleet

3.2 Hardware Requirements

The Architecture section should have helped you determine how many machines you will need for your deployment. This section will help you determine what kind of hardware specs each of those machines will need.

3.2.1 CPU Architecture

Security Onion only supports x86-64 architecture (standard Intel or AMD 64-bit processors).

**Warning:** We do not support ARM or any other non-x86-64 processors!

3.2.2 Minimum Specs

If you just want to import a pcap using `so-import-pcap`, then you can configure Security Onion 2 as an Import Node with the following minimum specs:

- 4GB RAM
- 2 CPU cores
- 200GB storage

For all other configurations, the minimum specs for running Security Onion 2 are:

- 12GB RAM
- 4 CPU cores
- 200GB storage

**Note:** These minimum specs are for EVAL mode with minimal services running. These requirements may increase drastically as you enable more services, monitor more traffic, and consume more logs. For more information, please see the detailed sections below.
3.2.3 Production Deployments

Security Onion 2 is a new platform with more features than previous versions of Security Onion. These additional features result in higher hardware requirements. For best results, we recommend purchasing new hardware to meet the new requirements.

Tip: If you’re planning to purchase new hardware, please consider official Security Onion appliances from Security Onion Solutions (https://securityonionsolutions.com). Our custom appliances have already been designed for certain roles and traffic levels and have Security Onion pre-installed. Purchasing from Security Onion Solutions will save you time and effort and help to support development of Security Onion as a free and open source platform!

3.2.4 Storage

We only support local storage. Remote storage like SAN/iSCSI/FibreChannel/NFS increases complexity and points of failure, and has serious performance implications. You may be able to make remote storage work, but we do not provide any support for it. By using local storage, you keep everything self-contained and you don’t have to worry about competing for resources. Local storage is usually the most cost efficient solution as well.

3.2.5 NIC

You’ll need at least one wired network interface dedicated to management (preferably connected to a dedicated management network). We recommend using static IP addresses where possible.

If you plan to sniff network traffic from a tap or span port, then you will need one or more interfaces dedicated to sniffing (no IP address). The installer will automatically disable NIC offloading functions such as tso, gso, and gro on sniffing interfaces to ensure that Suricata and Zeek get an accurate view of the traffic.

Make sure you get good quality network cards, especially for sniffing. Most users report good experiences with Intel cards.

Security Onion is designed to use wired interfaces. You may be able to make wireless interfaces work, but we don’t recommend or support it.

3.2.6 UPS

Like most IT systems, Security Onion has databases and those databases don’t like power outages or other ungraceful shutdowns. To avoid power outages and having to manually repair databases, please consider a UPS.

3.2.7 Elastic Stack

Please refer to the Architecture section for detailed deployment scenarios.

We recommend placing all Elastic storage (/nsm/elasticsearch) on SSD or fast spinning disk in a RAID 10 configuration.

3.2.8 Standalone Deployments

In a standalone deployment, the manager components and the sensor components all run on a single box, therefore, your hardware requirements will reflect that. You’ll need at minimum 16GB RAM, 4 CPU cores, and 200GB storage.
This deployment type is recommended for evaluation purposes, POCs (proof-of-concept) and small to medium size single sensor deployments. Although you can deploy Security Onion in this manner, it is recommended that you separate the backend components and sensor components.

- **CPU**: Used to parse incoming events, index incoming events, search metadata, capture PCAP, analyze packets, and run the frontend components. As data and event consumption increases, a greater amount of CPU will be required.
- **RAM**: Used for Logstash, Elasticsearch, disk cache for Lucene, Suricata, Zeek, etc. The amount of available RAM will directly impact search speeds and reliability, as well as ability to process and capture traffic.
- **Disk**: Used for storage of indexed metadata. A larger amount of storage allows for a longer retention period. It is typically recommended to retain no more than 30 days of hot ES indices.

Please refer to the [Architecture](#) section for detailed deployment scenarios.

### 3.2.9 Manager node with local log storage and search

In an enterprise distributed deployment, a manager node will store logs from itself and forward nodes. It can also act as a syslog destination for other log sources to be indexed into Elasticsearch. An enterprise manager node should have 8 CPU cores at a minimum, 16-128GB RAM, and enough disk space (multiple terabytes recommended) to meet your retention requirements.

- **CPU**: Used to parse incoming events, index incoming events, search metadata. As consumption of data and events increases, more CPU will be required.
- **RAM**: Used for Logstash, Elasticsearch, and disk cache for Lucene. The amount of available RAM will directly impact search speeds and reliability.
- **Disk**: Used for storage of indexed metadata. A larger amount of storage allows for a longer retention period. It is typically recommended to retain no more than 30 days of hot ES indices.

Please refer to the [Architecture](#) section for detailed deployment scenarios.

### 3.2.10 Manager node with separate search nodes

This deployment type utilizes search nodes to parse and index events. As a result, the hardware requirements of the manager node are reduced. An enterprise manager node should have at least 4-8 CPU cores, 16GB RAM, and 200GB to 1TB of disk space. Many folks choose to host their manager node in their VM farm since it has lower hardware requirements than sensors but needs higher reliability and availability.

- **CPU**: Used to receive incoming events and place them into Redis. Used to run all the front end web components and aggregate search results from the search nodes.
- **RAM**: Used for Logstash and Redis. The amount of available RAM directly impacts the size of the Redis queue.
- **Disk**: Used for general OS purposes and storing Kibana dashboards.

Please refer to the [Architecture](#) section for detailed deployment scenarios.

### 3.2.11 Search Node

Search nodes increase search and retention capacity with regard to Elasticsearch. These nodes parse and index events, and provide the ability to scale horizontally as overall data intake increases. Search nodes should have at least 4-8 CPU cores, 16-64GB RAM, and 200GB of disk space or more depending on your logging requirements.

- **CPU**: Used to parse incoming events and index incoming events. As consumption of data and events increases, more CPU will be required.
• RAM: Used for Logstash, Elasticsearch, and disk cache for Lucene. The amount of available RAM will directly impact search speeds and reliability.

• Disk: Used for storage of indexed metadata. A larger amount of storage allows for a longer retention period. It is typically recommended to retain no more than 30 days of hot ES indices.

Please refer to the Architecture section for detailed deployment scenarios.

### 3.2.12 Forward Node (Sensor)

A forward node runs sensor components only, and forwards metadata to the manager node. All PCAP stays local to the sensor, and is accessed through use of an agent.

• CPU: Used for analyzing and storing network traffic. As monitored bandwidth increases, a greater amount of CPU will be required. See below.

• RAM: Used for write cache and processing traffic.

• Disk: Used for storage of PCAP and metadata. A larger amount of storage allows for a longer retention period.

Please refer to the Architecture section for detailed deployment scenarios.

### 3.2.13 Heavy Node (Sensor with ES components)

A heavy node runs all the sensor components AND Elastic components locally. This dramatically increases the hardware requirements. In this case, all indexed metadata and PCAP are retained locally. When a search is performed through Kibana, the manager node queries this node’s Elasticsearch instance.

• CPU: Used to parse incoming events, index incoming events, search metadata. As monitored bandwidth (and the amount of overall data/events) increases, a greater amount of CPU will be required.

• RAM: Used for Logstash, Elasticsearch, and disk cache for Lucene. The amount of available RAM will directly impact search speeds and reliability.

• Disk: Used for storage of indexed metadata. A larger amount of storage allows for a longer retention period. It is typically recommended to retain no more than 30 days of hot ES indices.

Please refer to the Architecture section for detailed deployment scenarios.

### 3.2.14 Sensor Hardware Considerations

The following hardware considerations apply to sensors. If you are using a heavy node or standalone deployment type, please note that it will dramatically increase CPU/RAM/Storage requirements.

**Virtualization**

We recommend dedicated physical hardware (especially if you’re monitoring lots of traffic) to avoid competing for resources. Sensors can be virtualized, but you’ll have to ensure that they are allocated sufficient resources.

**CPU**

Suricata and Zeek are very CPU intensive. The more traffic you are monitoring, the more CPU cores you’ll need. A very rough ballpark estimate would be 200Mbps per Suricata worker or Zeek worker. So if you have a fully saturated 1Gbps link and are running Suricata and Zeek, then you’ll want at least 5 Suricata instances and 5 Zeek workers,
which means you’ll need at least 10 CPU cores for Suricata and Zeek with additional CPU cores for Stenographer and/or other services.

**RAM**

RAM usage is highly dependent on several variables:

- the services that you enable
- the **kinds** of traffic you’re monitoring
- the **actual amount of traffic** you’re monitoring (example: you may be monitoring a 1Gbps link but it’s only using 200Mbps most of the time)
- the amount of packet loss that is “acceptable” to your organization

For best performance, over provision RAM so that you can fully disable swap.

The following RAM estimates are a rough guideline and assume that you’re going to be running Suricata, Zeek, and Stenographer (full packet capture) and want to minimize/eliminate packet loss. Your mileage may vary!

If you just want to quickly evaluate Security Onion in a VM, the bare minimum amount of RAM needed is 12GB. More is obviously better!

If you’re deploying Security Onion in production on a small network (100Mbps or less), you should plan on 16GB RAM or more. Again, more is obviously better!

If you’re deploying Security Onion in production to a medium network (100Mbps - 1000Mbps), you should plan on 16GB - 128GB RAM or more.

If you’re deploying Security Onion in production to a large network (1000Mbps - 10Gbps), you should plan on 128GB - 256GB RAM or more.

If you’re buying a new server, go ahead and max out the RAM (it’s cheap!). As always, more is obviously better!

**Storage**

Sensors that have full packet capture enabled need LOTS of storage. For example, suppose you are monitoring a link that averages 50Mbps, here are some quick calculations: 50Mb/s = 6.25 MB/s = 375 MB/minute = 22,500 MB/hour = 540,000 MB/day. So you’re going to need about 540GB for one day’s worth of pcaps (multiply this by the number of days you want to keep on disk for investigative/forensic purposes). The more disk space you have, the more PCAP retention you’ll have for doing investigations after the fact. Disk is cheap, get all you can!

**Packets**

You need some way of getting packets into your sensor interface(s). If you’re just evaluating Security Onion, you can replay **PCAPs for Testing**. For a production deployment, you’ll need a tap or SPAN/monitor port. Here are some inexpensive tap/span solutions:

Sheer Simplicity and Portability (USB-powered):
http://www.dual-comm.com/port-mirroring-LAN_switch.htm

Dirt Cheap and Versatile:
https://mikrotik.com/product/RB260GS

3.2. Hardware Requirements
Netgear GS105E (requires Windows app for config):
https://www.netgear.com/support/product/GS105E.aspx

Netgear GS105E v2 (includes built-in web server for config):
https://www.netgear.com/support/product/GS105Ev2

low cost TAP that uses USB or Ethernet port:
http://www.midbittech.com

More exhaustive list of enterprise switches with port mirroring:
http://www.miarec.com/knowledge/switches-port-mirroring

Enterprise Tap Solutions:
  • Net Optics / Ixia
  • Arista Tap Aggregation Feature Set
  • Gigamon
  • cPacket
  • Bigswitch Monitoring Fabric
  • Garland Technologies Taps
  • APCON
  • Profitap

Further Reading
For large networks and/or deployments, please also see https://github.com/pevma/SEPTun.

3.3 Partitioning

Now that you understand Hardware Requirements, we should next discuss disk partitioning. If you’re installing Security Onion for a production deployment, you’ll want to pay close attention to partitioning to make sure you don’t fill up a partition at some point.

3.3.1 Minimum Storage

As the Hardware Requirements section mentions, the MINIMUM requirement is 200GB storage. This is to allow 100GB for /nsm and 100GB for the rest of /.

3.3.2 ISO

If you use our Security Onion ISO image, it will automatically partition your disk for you. If you instead use CentOS 7 or Ubuntu 18.04, you will most likely need to manually modify their default partition layout.
3.3.3 LVM

You may want to consider Logical Volume Management (LVM) as it will allow you to more easily change your partitioning in the future if you need to. As of Security Onion 2.0.3, our Security Onion ISO image uses LVM by default.

3.3.4 /boot

You probably want a dedicated /boot partition of at least 500MB at the beginning of the drive.

3.3.5 /nsm

The vast majority of data will be written to /nsm, so you’ll want to dedicate the vast majority of your disk space to that partition. You’ll want at least 100GB.

3.3.6 /

/ (the root partition) currently contains /var/lib/docker/ (more on that below) and thus you’ll want at least 100GB.

3.3.7 Docker

Docker images are currently written to /var/lib/docker/. The current set of Docker images uses 27GB on disk. If you’re planning a production deployment, you should plan on having enough space for another set of those Docker images for in-place updates.

3.3.8 Other

If you install using a standard CentOS 7 or Ubuntu 18.04 ISO, then those installers may try to dedicate a large amount of space to /home. You may need to adjust this to ensure that it is not overly large and wasting valuable disk space.

3.3.9 Example

Here’s an example of how our current Security Onion ISO image partitions a 1TB disk:

- 500MB /boot partition at the beginning of the drive
- the remainder of the drive is an LVM volume that is then partitioned as follows:
  - 630GB /nsm
  - 300GB /
  - 2GB /tmp
  - 8GB swap
3.4 Release Notes

Before downloading, please review the notes for this release.

Security Onion 2 is now generally available and is at version 2.3.10!

3.4.1 2.3.10 Changes

- UEFI installs with multiple disks should work as intended now.
- Telegraf scripts will now make sure they are not already running before execution.
- You are now prompted during setup if you want to change the docker IP range. If you change this it needs to be the same on all nodes in the grid.
- Soup will now download the new containers before stopping anything. If anything fails it will now exit and leave the grid at the current version.
- All containers are now hosted on quay.io to prevent pull limitations. We are now using GPG keys to determine if the image is from Security Onion.
- Osquery installers have been updated to osquery 4.5.1
- Fix for bug where Playbook was not removing the Elastalert rules for inactive Plays
- Exifdata reported by Strelka is now constrained to a single multi-valued field to prevent mapping explosion (scan.exiftool).
- Resolved issue with Navigator layer(s) not loading correctly.
- Wazuh authd is now started by default on port 1515/tcp.
- Wazuh API default credentials are now removed after setup. Scripts have been added for API user management.
- Upgraded Salt to 3002.2 due to CVEs.
- If salt-minion is unable to apply states after the defined threshold, we assume salt-minion is in a bad state and the salt-minion service will be restarted.
- Fixed bug that prevented mysql from installing for Fleet if Playbook wasn’t also installed.
- so-status will now show STARTING or WAIT_START, instead of ERROR if so-status is run before a salt high-state has started or finished for the first time after system startup
- Stenographer can now be disabled on a sensor node by setting the pillar steno:enabled:false in its minion.sls file or globally if set in the global.sls file
- Added so-ssh-harden script that runs the commands listed in SSH.
- NGINX now redirects the browser to the hostname/IP address/FQDN based on global:url_base
- MySQL state now waits for MySQL server to respond to a query before completing
- Added Analyst option to network installs
- Acknowledging (and Escalating) alerts did not consistently remove the alert from the visible list; this has been corrected.
- Escalating alerts that have a rule.case_template field defined will automatically assign that case template to the case generated in TheHive.
- Alerts and Hunt interface quick action bar has been converted into a vertical menu to improve quick action option clarity. Related changes also eliminated the issues that occurred when the quick action bar was appearing to the left of the visible browser area.
• Updated Go to newer version to fix a timezone, daylight savings time (DST) issue that resulted in Alerts and Hunt interfaces not consistently showing results.
• Improved Hunt and Alert table sorting.
• Alerts interface now allows absolute time searches.
• Alerts interface ‘Hunt’ quick action is now working as intended.
• Alerts interface ‘Ack’ icon tooltip has been changed from ‘Dismiss’ to ‘Acknowledge’ for consistency.
• Hunt interface bar charts will now show the quick action menu when clicked instead of assuming the click was intended to add an include filter.
• Hunt interface quick action will now cast a wider net on field searches.
• Now explicitly preventing the use of a dollar sign ($) character in web user passwords during setup.
• Cortex container will now restart properly if the SO host was not gracefully shutdown.
• Added syslog plugin to the logstash container; this is not in-use by default but available for those users that choose to use it.
• Winlogbeat download package is now available from the SOC Downloads interface.
• Upgraded Kratos authentication system.
• Added new Reset Defaults button to the SOC Profile Settings interface which allows users to reset all local browser SOC customizations back to their defaults. This includes things like default sort column, sort order, items per page, etc.

3.4.2 Known Issues

• For Ubuntu, non master nodes, you may need to ssh to each node and run `salt-call state.highstate` in order initiate the update. To verify if this needs to be done on remote nodes, from the master, run `salt \\* pkg.version salt-minion` after 30 minutes following the initial soup update. If the node does not return that it running Salt 3002.2, then the node will need to manually be highstated locally from the node to complete the update.

• During soup, you may see the following during the first highstate run, it can be ignored: Rendering SLS '<some_sls_here>' failed: Jinja variable 'list object' has no attribute 'values'. The second highstate will complete without that error.

• During install or soup, there is a false positive failure condition that can occur. It is caused by [ERROR ] Failed to add job <job_name> to schedule.. This error indicates that Salt was unable to add a job to a schedule. If you see this in setup or soup log, it can be confirmed if this is false positive or not by running `salt-call schedule.list` on the node that saw the error. If the job isn’t in the schedule list, run `salt-call state.highstate` and check if the job was added after it completes.

3.4.3 2.3.2 Changes

• Elastic components have been upgraded to 7.9.3.
• Fixed an issue where curator was unable to delete a closed index.
• Cheat sheet is now available for airgap installs.
3.4.4 2.3.1 Changes

- Fixed a SOC issue in airgap mode that was preventing people from logging in.
- Downloading Elastic features images will now download the correct images.
- Winlogbeat download no longer requires Internet access.
- Adjusted Alerts quick action bar to allow searching for a specific value while remaining in Alerts view.
- /nsm will properly display disk usage on the standalone Grafana dashboard.
- The manager node now has syslog listener enabled by default (you’ll still need to allow syslog traffic through the firewall of course).
- Fixed an issue when creating host groups with so-firewall.

3.4.5 Known Issues

- It is still possible to update your grid from any release candidate to 2.3. However, if you have a true production deployment, then we recommend a fresh image and install for best results.
- In 2.3.0 we made some changes to data types in the elastic index templates. This will cause some errors in Kibana around field conflicts. You can address this in 2 ways:
  - Delete all the data on the ES nodes (preserving all of your other settings such as BPFs) by running `sudo so-elastic-clear` on all the search nodes.
  - Re-index the data. This is not a quick process but you can find more information at https://docs.securityonion.net/en/2.3/elasticsearch.html#re-indexing
- Please be patient as we update our documentation. We have made a concerted effort to update as much as possible but some things still may be incorrect or omitted. If you have questions or feedback, please start a discussion at https://securityonion.net/discuss.
- Once you update your grid to 2.3, any new nodes that join the grid must be 2.3 so if you try to join an older node it will fail. For best results, use the latest 2.3 ISO (or 2.3 installer from github) when joining to a 2.3 grid.
- Shipping Windows Eventlogs with Osquery will fail intermittently with utf8 errors logged in the Application log. This is scheduled to be fixed in Osquery 4.5.
- When running soup to upgrade from older versions to 2.3, there is a Salt error that may occur during the final highstate. This error is related to the patch_os_schedule and can be ignored as it should not occur again in subsequent highstates.
- When Search Nodes are upgraded from older versions to 2.3, there is a chance of a race condition where certificates are missing. This will show errors in the manager log to the remote node. To fix this run the following on the search node that is having the issue:
  - Stop elasticsearch - `sudo so-elasticsearch-stop`
  - Run the SSL state - `sudo salt-call state.apply ssl`
  - Restart elasticsearch - `sudo so-elasticsearch-restart`
- If you are upgrading from RC1 you might see errors around registry:2 missing. This error does not break the actual upgrade. To fix, run the following on the manager:
  - Stop the Docker registry - `sudo docker stop so-dockerregistry`
  - Remove the container - `sudo docker rm so-dockerregistry`
  - Run the registry state - `sudo salt-call state.apply registry`
3.4.6 2.3.0 Changes

- We have a new Alerts interface for reviewing alerts and acknowledging or escalating them. Escalating creates a new case in TheHive. Please note that TheHive no longer receives alerts directly.
- Kibana no longer presents the option to create alerts from events, but instead allows creation of cases from events.
- Our Security Onion ISO now works for UEFI as well as Secure Boot.
- Airgap deployments can now be updated using the latest ISO. Please read this documentation carefully.
- Suricata has been updated to version 5.0.4.
- Zeek has been updated to version 3.0.11.
- Stenographer has been updated to the latest version.
- soup will now attempt to clean up old docker images to free up space.
- Hunt actions can be customized via hunt.actions.json.
- Hunt queries can be customized via hunt.queries.json.
- Hunt event fields can be customized via hunt.eventfields.json.
- Alerts actions can be customized via alerts.actions.json.
- Alerts queries can be customized via alerts.queries.json.
- Alerts event fields can be customized via alerts.eventfields.json.
- This help documentation is now viewable offline for airgap installations.
- The script so-user-add will now validate the password is acceptable before attempting to create the user.
- Playbook and Grafana no longer use static passwords for their admin accounts.
- Analyst VM now comes with NetworkMiner 2.6 installed.
- Strelka YARA matches now generate alerts that can be viewed through the Alerts interface.

3.4.7 2.2.0 Changes

- Setup now includes an option for airgap installations
- Playbook now works properly when installed in airgap mode
- Added so-analyst script to create an analyst workstation with GNOME desktop, Chromium browser, Wireshark, and NetworkMiner
- Upgraded Zeek to version 3.0.10 to address a recent security issue
- Upgraded Docker to latest version
- Re-worked IDSTools to make it easier to modify
- Added so-* tools to the default path so you can now tab complete
- so-status can now be run from a manager node to get the status of a remote node. Run salt <target> so.status
- Salt now prevents states from running on a node that it shouldn’t so you can’t, for example, accidentally apply the elasticsearch state on a forward node
- Added logic to check for Salt mine corruption and recover automatically
• Collapsed Hunt filter icons and action links into a new quick action bar that will appear when a field value is clicked; actions include:
  – Filtering the hunt query
  – Pivot to PCAP
  – Create an alert in TheHive
  – Google search for the value
  – Analyze the value on VirusTotal.com

• Fixed minor bugs in Hunt user interface relating to most-recently used queries, tooltips, and more
• so-user-add now automatically adds users to Fleet and TheHive (in addition to SOC)
• Introduced so-user-disable and so-user-enable commands which allows administrators to lock out users that are no longer permitted to use Security Onion
• Added icon to SOC Users list representing their active or locked out status
• Removed User delete action from SOC interface in favor of disabling users for audit purposes
• Prune old PCAP job data from sensors once the results are streamed back to the manager node
• Hunt filtering to a specific value will search across all fields instead of only the field that was originally clicked
• Limiting PCAP jobs to extract at most 2GB from a sensor to avoid users accidentally requesting unreasonably large PCAP via the web interface
• so-test is back - run it to easily replay PCAPs and verify that all the components are working as expected
• New Elasticsearch subfield (.security) based on the new community-driven analyzer from @neu5ron - https://github.com/neu5ron/es_stk
  • Playbook now uses the new .security subfield for case-insensitive wildcard searches

3.4.8 2.1.0 Changes

• Fixed an issue where the console was timing out and making it appear that the installer was hung
• Introduced Import node type ideal for running so-import-pcap to import pcap files and view the resulting logs in Hunt or Kibana
• Moved static.sls to global.sls to align the name with the functionality
• Traffic between nodes in a distributed deployment is now fully encrypted
• Playbook
  – Elastalert now runs active Plays every 3 minutes
  – Changed default rule-update config to only import Windows rules from the Sigma Community repo
  – Lots of bug fixes & stability improvements
• Ingest Node parsing updates for Osquery and Winlogbeat - implemented single pipeline for Windows eventlogs & sysmon logs
• Upgraded Osquery to 4.4 and re-enabled auto-updates
• Upgraded to Salt 3001.1
• Upgraded Wazuh to 3.13.1
• Hunt interface now shows the timezone being used for the selected date range
• Fixed Cortex initialization so that TheHive integration and initial user set is correctly configured
• Improved management of TheHive/Cortex credentials
• SOC now allows for arbitrary, time-bounded PCAP job creation, with optional filtering by host and port

3.4.9 2.0.3 Changes

• Resolved an issue with large drives and the ISO install
• Modified ISO installation to use Logical Volume Management (LVM) for disk partitioning
• Updated Elastic Stack components to version 7.8.1
• Updated Zeek to version 3.0.8

3.4.10 2.0.2 Changes

• Sensoroni fails on 2.0.1 ISO EVAL installation #1089
  https://github.com/Security-Onion-Solutions/securityonion/issues/1089

3.4.11 2.0.1 Changes

• Security Fix: variables.txt from ISO install stays on disk for 10 days
  https://github.com/Security-Onion-Solutions/securityonion/issues/1067
• Security Fix: Remove user values from static.sls
  https://github.com/Security-Onion-Solutions/securityonion/issues/1068
• Fix distributed deployment sensor interval issue allowing PCAP
• Support for passwords that start with special characters
  https://github.com/Security-Onion-Solutions/securityonion/issues/1058
• Minor soup updates

3.4.12 2.0.0 Changes

• This version requires a fresh install, but there is good news - we have brought back soup! From now on, you should be able to run soup on the manager to upgrade your environment to RC2 and beyond!
• Re-branded 2.0 to give it a fresh look
• All documentation has moved to our docs site
• soup is alive! Note: This tool only updates Security Onion components. Please use the built-in OS update process to keep the OS and other components up to date
• so-import-pcap is back! See the docs here
• Fixed issue with so-features-enable
• Users can now pivot to PCAP from Suricata alerts
• ISO install now prompts users to create an admin/sudo user instead of using a default account name
• The web email & password set during setup is now used to create the initial accounts for TheHive, Cortex, and Fleet
• Fixed issue with disk cleanup
• Changed the default permissions for /opt/so to keep non-priviledged users from accessing salt and related files
• Locked down access to certain SSL keys
• Suricata logs now compress after they roll over
• Users can now easily customize shard counts per index
• Improved Elastic ingest parsers including Windows event logs and Sysmon logs shipped with WinLogbeat and Osquery (ECS)
• Elastic nodes are now “hot” by default, making it easier to add a warm node later
• so-allow now runs at the end of an install so users can enable access right away
• Alert severities across Wazuh, Suricata and Playbook (Sigma) have been standardized and copied to event.severity:
  – 1-Low / 2-Medium / 3-High / 4-Critical
• Initial implementation of alerting queues:
  – Low & Medium alerts are accessible through Kibana & Hunt
  – High & Critical alerts are accessible through Kibana, Hunt and sent to TheHive for immediate analysis
• ATT&CK Navigator is now a statically-hosted site in the nginx container
• Playbook
  – All Sigma rules in the community repo (500+) are now imported and kept up to date
  – Initial implementation of automated testing when a Play’s detection logic has been edited (i.e., Unit Testing)
  – Updated UI Theme
  – Once authenticated through SOC, users can now access Playbook with analyst permissions without login
• Kolide Launcher has been updated to include the ability to pass arbitrary flags - new functionality sponsored by SOS
• Fixed issue with Wazuh authd registration service port not being correctly exposed
• Added option for exposure of Elasticsearch REST API (port 9200) to so-allow for easier external querying/integration with other tools
• Added option to so-allow for external Strelka file uploads (e.g., via strelka-fileshot)
• Added default YARA rules for Strelka – default rules are maintained by Florian Roth and pulled from https://github.com/Neo23x0/signature-base
• Added the ability to use custom Zeek scripts
• Renamed “master server” to “manager node”
• Improved unification of Zeek and Strelka file data
3.5 Download

Now that you’ve reviewed the Release Notes in the previous section, you’re ready to download Security Onion! You can either download our Security Onion ISO image (based on CentOS 7) or download a standard CentOS 7 64-bit or Ubuntu 18.04 ISO image and then add our Security Onion components. Please keep in mind that we only support CentOS 7 and Ubuntu 18.04.

Tip: For most use cases, we recommend using our Security Onion ISO image as it’s the quickest and easiest method.

Warning: ALWAYS verify the checksum of ANY downloaded ISO image! Regardless of whether you’re downloading our Security Onion ISO image or a standard CentOS or Ubuntu ISO image, you should ALWAYS verify the downloaded ISO image to ensure it hasn’t been tampered with or corrupted during download.

- If downloading our Security Onion 2.3 ISO image, you can find the download link and verification instructions here: https://github.com/Security-Onion-Solutions/securityonion/blob/master/VERIFY_ISO.md
- If downloading an Ubuntu or CentOS ISO image, please verify that ISO image using whatever instructions they provide.

See also:
If you’re going to create a bootable USB from one of the ISO images above, there are many ways to do that. One popular choice that seems to work well for many folks is Balena Etcher which can be downloaded at https://www.balena.io/etcher/.

3.6 VMWare

3.6.1 Overview

In this section, we’ll cover creating a virtual machine (VM) for our Security Onion 2 ISO image in VMware Workstation Pro and VMware Fusion. These steps should be fairly similar for most VMware installations. If you don’t already have VMware, you can download VMware Workstation Player from http://www.vmware.com/products/player/playerpro-evalex.html.

Note: With the sniffing interface in bridged mode, you will be able to see all traffic to/from the host machine’s physical NIC. If you would like to see ALL the traffic on your network, you will need a method of forwarding that traffic to the interface to which the virtual adapter is bridged. This can be achieved by switch port mirroring (SPAN), or through the use of a tap.

3.6.2 Workstation Pro

VMware Workstation is available for many different host operating systems, including Windows and several popular Linux distros. Follow the steps below to create a VM in VMware Workstation Pro for our Security Onion ISO image:

1. From the VMware main window, select File >> New Virtual Machine.
2. Select Typical installation >> Click Next.
3. Installer disc image file >> SO ISO file path >> Click Next.
4. Choose Linux, CentOS 7 64-Bit and click Next.
5. Specify virtual machine name and click Next.
6. Specify disk size (minimum 200GB), store as single file, click Next.
7. Customize hardware and increase Memory (minimum 12GB for most use cases) and Processors (minimum 4 CPU cores for most use cases).
8. Network Adapter (NAT or Bridged – if you want to be able to access your Security Onion machine from other devices in the network, then choose Bridged, otherwise choose NAT to leave it behind the host) – in this tutorial, this will be the management interface.
9. Add >> Network Adapter (Bridged) - this will be the sniffing (monitor) interface.
10. Click Close.
11. Click Finish.
12. Power on the virtual machine and then follow the installation steps for your desired installation type in the Installation section.

### 3.6.3 Fusion

VMware Fusion is available for Mac OS. For more information about VMware Fusion, please see [https://www.vmware.com/products/fusion.html](https://www.vmware.com/products/fusion.html).

Follow the steps below to create a VM in VMware Fusion for our Security Onion ISO image:

1. From the VMware Fusion main window, click File and then click New.
2. Select the Installation Method appears. Click Install from disc or image and click Continue.
3. Create a New Virtual Machine appears. Click Use another disc or disc image..., select our ISO image, click Open, then click Continue.
4. Choose Operating System appears. Click Linux, click CentOS 7 64-bit, then click Continue.
5. Choose Firmware Type appears. Click Legacy BIOS and then click Continue.
6. Finish screen appears. Click the Customize Settings button.
7. Save As screen appears. Give the VM a name and click the Save button.
8. Settings window appears. Click Processors & Memory.
9. Processors & Memory screen appears. Increase the number of processors to at least 4 and memory to at least 12GB depending on your use case. Click the Add Device... button.
10. Add Device screen appears. Click Network Adapter and click the Add... button.
11. Network Adapter 2 screen appears. This will be the sniffing (monitor) interface. Select your desired network adapter configuration. Click the Show All button.
13. Hard Disk (SCSI) screen appears. Increase the disk size to at least 200GB depending on your use case. Click the Apply button.
15. At the window for your new VM, click the Play button to power on the virtual machine.
16. Follow the installation steps for your desired installation type in the Installation section.
3.7 VirtualBox

In this section, we’ll cover installing Security Onion on VirtualBox. You’ll need a computer with at least 16GB of RAM (so that we can dedicate at least 12GB RAM to the VM) for best results. You can download a copy of VirtualBox for Windows, Mac OS X, or Linux at http://www.virtualbox.org.

3.7.1 Creating VM

Launch VirtualBox and click the “New” button. First we’ll provide a name for our virtual machine (“Security Onion” for example) and specify the type (“Linux”) and version (“CentOS” or “CentOS 64 bit”), then click “Continue.” We’ll next define how much memory we want to make available to our virtual machine. You should dedicate at least 12GB RAM to the Security Onion VM.

Next we’ll create a virtual hard drive. Specify “Create a virtual hard drive now” then click “Create” to choose the hard drive file type “VDI (VirtualBox Disk Image)” and “Continue.” For storage, we have the options of “Dynamically allocated” or “Fixed size.” For a client virtual machine, “Dynamically allocated” is the best choice as it will grow the hard disk up to whatever we define as the maximum size on an as needed basis until full, at which point Security Onion’s disk cleanup routines will work to keep disk space available. If you happen to be running a dedicated sensor in a virtual machine, I would suggest using “Fixed size,” which will allocate all of the disk space you define up front and save you some disk performance early on. Once you’ve settled on the storage allocation, click “Continue” and provide a name from your hard disk image file and specify the location where you want the disk file to be created if other than the default location. For disk size, you’ll want at least 200GB so you have enough capacity for retrieving/testing packet captures and downloading system updates. Click “Create” and your Security Onion VM will be created.

At this point, you can click “Settings” for your new virtual machine so we can get it configured. You might want to increase your display virtual memory to 128MB of RAM, but most other settings should be fine. We do, however, need to do a couple of things. First, mount the Security Onion ISO file so our VM can boot from it to install Linux. Click the “Storage” icon, then under “Controller: IDE” select the “Empty” CD icon. To the right, you’ll see “CD/DVD Drive” with “IDE Secondary” specified with another CD icon. Click the icon, then select “Choose a virtual CD/DVD disk file” and browse to where you downloaded the Security Onion ISO file, select it then choose “Open.” Next click “Network” then “Adapter 2.” You’ll need to click the checkbox to enable it then attach it to “Internal Network.” Under the “Advanced” options, set “Promiscuous Mode” to “Allow All.” Click “Ok” and we are ready to install the operating system.

Hit the “Start” button with your new virtual machine selected and after a few seconds the boot menu will load.

Follow the installation steps for your desired installation type in the Installation section.

**Tip:** You’ll notice two icons on the top right in VirtualBox Manager when you select your virtual machine: Details and Snapshots. Click “Snapshots” then click the camera icon and give your snapshot a name and description. Once we have a snapshot, we’ll be able to make changes to the system and revert those changes back to the state we are preserving.

3.7.2 Guest Additions

If you want to install VirtualBox Guest Additions, please see https://wiki.centos.org/HowTos/Virtualization/VirtualBox/CentOSguest.

3.8 Booting Issues

If you have trouble booting an ISO image, here are some troubleshooting steps:
• Verify the downloaded ISO image using hashes or GPG key.
• Verify that your machine is x86-64 architecture (standard Intel or AMD 64-bit).
• If you’re trying to run a 64-bit virtual machine, verify that your 64-bit processor supports virtualization and that virtualization is enabled in the BIOS.
• If you’re trying to create a bootable USB from an ISO image, try using Balena Etcher which can be downloaded at https://www.balena.io/etcher/.
• Certain display adapters may require the `nomodeset` option passed to the kernel (see https://unix.stackexchange.com/questions/353896/linux-install-goes-to-blank-screen).
• If you’re still having problems with our 64-bit ISO image, try downloading the standard CentOS 7 64-bit ISO image or Ubuntu 18.04 64-bit ISO image and seeing if they run. If they don’t, then you should double-check your 64-bit compatibility.
• If all else fails but standard CentOS 7 64-bit or Ubuntu 18.04 64-bit installs normally, then you can always install our components on top of them as described on the Installation page.

### 3.9 Installation

Having downloaded your desired ISO according to the Download section, it’s now time to install! There are separate sections below to walk you through installing using our Security Onion ISO image (based on CentOS 7) or installing standard CentOS 7 or Ubuntu 18.04 and then installing our components on top.

**Tip:** For most use cases, we recommend using our Security Onion ISO image as it’s the quickest and easiest method.

#### 3.9.1 Installation using Security Onion ISO Image

If you want to install Security Onion using our ISO image:

1. Review the Hardware Requirements and Release Notes sections.
2. Download and verify our Security Onion ISO image.
3. Boot the ISO in a machine that meets the minimum hardware specs.
4. Follow the prompts to complete the installation and reboot.
5. Login using the username and password you set in the installer.
6. Proceed to the Configuration section.

#### 3.9.2 Installation on Ubuntu or CentOS

If you want to install Security Onion on CentOS 7 or Ubuntu 18.04 (not using our Security Onion ISO image), follow these steps:

1. Review the Hardware Requirements section.
2. Download the ISO image for your preferred flavor of Ubuntu 18.04 64-bit or CentOS 7 64-bit, verify the ISO image, and boot from it.
3. Follow the prompts in the installer. If you’re building a production deployment, you’ll probably want to use LVM and dedicate most of your disk space to `/nsm` as discussed in the Partitioning section.
4. Reboot into your new installation.

5. Login using the username and password you specified during installation.

6. If using CentOS 7 Minimal, install `git`:
   
   ```
   sudo yum -y install git
   ```

7. Once you have `git`, then do the following:
   
   ```
   git clone https://github.com/Security-Onion-Solutions/securityonion
cd securityonion
sudo bash so-setup-network
   ```

8. Proceed to the `Configuration` section.

### 3.10 Configuration

Now that you’ve installed Security Onion, it’s time to configure it!

**Note:** Setup uses keyboard navigation and you can use arrow keys to move around. Certain screens may provide a list and ask you to select one or more items from that list. You can use the space bar to select items and the Enter key to proceed to the next screen.

**Warning:** If you use DHCP and your IP address changes, this can cause problems. If you want to use DHCP, make sure that you have a DHCP reservation so that your IP address does not change. Otherwise, use a static IP address to be safe.

Security Onion is designed for many different use cases. Here are just a few examples!

**Tip:** If this is your first time using Security Onion and you just want to try it out, we recommend the Import option as it’s the quickest and easiest way to get started.

#### 3.10.1 Import

One of the easiest ways to get started with Security Onion is using it to forensically analyze one or more pcap files. Just install Security Onion in `Import` mode and then run `so-import-pcap` giving it the full path to one or more pcap files. For more information, please see the `so-import-pcap` section.

#### 3.10.2 Evaluation

Evaluation Mode is ideal for classroom or small lab environments. Evaluation is **not** designed for production usage. Choose `EVAL`, follow the prompts (see screenshots below), and then proceed to the `After Installation` section.

#### 3.10.3 Production Server - Standalone

Standalone is similar to Evaluation in that it only requires a single box, but Standalone is more ready for production usage. Choose `STANDALONE`, follow the prompts, and then proceed to the `After Installation` section.
3.10.4 Production Server - Distributed Deployment

If deploying a distributed environment, install and configure the manager node first and then join the other nodes to it. For best performance, the manager node should be dedicated to just being a manager for the other nodes (the manager node should have no sniffing interfaces of its own). Please note that all nodes will need to be able to connect to the manager node on ports 22, 4505, and 4506.

Build the manager by following the prompts. Save the soremote password so that you can join nodes to the manager.

Build search nodes and join them to the manager node using the soremote password.

Build forward nodes and join them to the manager node using the soremote password.

Proceed to the After Installation section.

3.10.5 Screenshots

The following screenshots are from an EVAL installation with all services enabled. Your screens may be different depending on what options you choose.
3.10. Configuration
Please add NICs to the Monitor Interface

ens34

<Ok> <Cancel>

Enter your HOME_NET separated by ,

192.168.0.0/16,172.16.0.0/12

<Ok> <Cancel>

Please keep in mind the more services that you enable the more RAM that is required.

<Ok>
3.10. Configuration

Security Onion Documentation, Release 2.3

Select Components to install

- GRAFANA  Enable Grafana for system monitoring
- OSQUERY  Enable Fleet with osquery
- WAZUH  Enable Wazuh
- THEHIVE  Enable TheHive
- PLAYBOOK  Enable Playbook
- STRELKA  Enable Strelka

<Ok>  <Cancel>

Security Onion Setup

Do you want to enable the default YARA rules for Strelka?

<Yes>  <No>

Security Onion Install

Please enter an email address to create an administrator account for the web interface.

<Ok>  <Cancel>
3.11 After Installation

3.11.1 Adjust firewall rules using so-allow

Depending on what kind of installation you did, the Setup wizard may have already walked you through adding firewall rules to allow your analyst IP address(es). If you need to allow other IP addresses, you can manually run `so-allow`.

3.11.2 Services

- Verify services are running:
  ```bash
sudo so-status
  ```

3.11.3 Other

- Full-time analysts may want to connect using a dedicated Analyst VM.
- Any IDS/NSM system needs to be tuned for the network it’s monitoring. Please see the Tuning section.
- Configure the OS to use your preferred NTP server.
Once you’ve run `so-allow` and allowed your IP address, you can then connect to Security Onion Console (SOC) with your web browser. We recommend chromium or chromium-based browsers such as Google Chrome. Other browsers may work, but chromium-based browsers provide the best compatibility.

Depending on the options you chose in the installer, connect to the IP address or hostname of your Security Onion installation. Then login using the email address and password that you specified in the installer.
Once logged in, you’ll see links on the left side for analyst tools like Alerts, Hunt, PCAP, Kibana, CyberChef, Playbook, TheHive, and ATT&CK Navigator. While Alerts, Hunt, and PCAP are native to SOC itself, the remaining tools are external and will spawn separate browser tabs.
Tip: SOC gives you access to a variety of tools and they all complement each other very well. For example, here’s one potential workflow:

- Check **Grafana** to make sure your system is healthy.
- Go to the **Alerts** page and review unacknowledged alerts.
- Once you’ve found an alert that you want to investigate, you might want to expand your search and look for additional logs relating to the source and destination IP addresses, so pivot to **Hunt** for more information. If any of those additional logs look interesting, you might then want to pivot to **PCAP** to look at the full packet capture for that stream.
- Send alert to **TheHive** and document any indicators of compromise (IOCs) found in the previous step.
- Go to **Fleet** and perform a wider search for those IOCs across all **osquery** endpoints.
- Use **CyberChef** to further analyze and decode additional host artifacts.
- Develop a play in **Playbook** that will automatically alert on IOCs moving forward and update your coverage in **ATT&CK Navigator**.
- Finally, return to **TheHive** and document the entire investigation and close the case.

### 4.1 Alerts

**Security Onion Console (SOC)** gives you access to our new Alerts interface. This interface gives you an overview of the alerts that Security Onion is generating and allows you to quickly drill down into details, pivot to **Hunt** or **PCAP**, and escalate alerts to **TheHive**.

#### 4.1.1 Toggles

The top of the page has toggles for **Acknowledged** and **Escalated**:

<table>
<thead>
<tr>
<th>Rule Name</th>
<th>Event Module</th>
<th>Event Severity</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPL2B</td>
<td>sicorta</td>
<td>low</td>
</tr>
<tr>
<td>ET MALWARE Backdoor family PCRec/Goost. Cnc traffic (OUTBOUND) 102</td>
<td>sicorta</td>
<td>high</td>
</tr>
<tr>
<td>ET MALWARE Backdoor family PCRec/Goost. Cnc traffic</td>
<td>sicorta</td>
<td>high</td>
</tr>
<tr>
<td>ET JAS Hash - Possible Malware - Various Trickbot/Kevo/Drhdx</td>
<td>sicorta</td>
<td>low</td>
</tr>
<tr>
<td>ET MALWARE ABUSE.CH SSL Blacklist Malicious SSL certificate detected [Dridex/Trickbot/Cnc]</td>
<td>sicorta</td>
<td>high</td>
</tr>
<tr>
<td>ET POLICY OpenSSL Demo CA - Internet Widgee Fyi (I)</td>
<td>sicorta</td>
<td>low</td>
</tr>
<tr>
<td>ET HUNTING Suspicious HTTP Request to .dcm domain</td>
<td>sicorta</td>
<td>medium</td>
</tr>
<tr>
<td>ET HUNTING Suspicious GET To gats.php with no Referer</td>
<td>sicorta</td>
<td>medium</td>
</tr>
<tr>
<td>ET MALWARE Generic get()</td>
<td>sicorta</td>
<td>high</td>
</tr>
<tr>
<td>ET INFO Suspicious Windows NT version 9 User Agent</td>
<td>sicorta</td>
<td>high</td>
</tr>
<tr>
<td>ET MALWARE Ghidra Remote Access Trojan Encrypted Session To Cnc Server</td>
<td>sicorta</td>
<td>high</td>
</tr>
</tbody>
</table>
• Enabling the **Acknowledged** toggle will only show alerts that have previously been acknowledged by an analyst.

• Enabling the **Escalated** toggle will only show alerts that have previously been escalated by an analyst to *TheHive*.

### 4.1.2 Query Bar

The query bar defaults to **Group By Name, Module** which groups the alerts by `rule.name` and `event.module`. If you want to send your current Alerts query to *Hunt*, you can click the crosshair icon to the right of the query bar.

```
Group By Name, Module
```

Under the query bar, you’ll notice colored bubbles that represent the individual components of the query and the fields to group by. If you want to remove part of the query, you can click its corresponding bubble to remove it and run a new search.

You can click the dropdown box to select other queries which will group by other fields.

```
Group By Name, Module
```

<table>
<thead>
<tr>
<th>Group By Name, Module</th>
<th>Group: event.severity_label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group By Sensor, Source IP/Port, Destination IP/Port, Name</td>
<td>Group: event.severity_label</td>
</tr>
<tr>
<td>Group By Source IP, Name</td>
<td>Group: event.severity_label</td>
</tr>
<tr>
<td>Group By Source Port, Name</td>
<td>Group: event.severity_label</td>
</tr>
<tr>
<td>Group By Destination IP, Name</td>
<td>Group: event.severity_label</td>
</tr>
<tr>
<td>Group By Destination Port, Name</td>
<td>Group: event.severity_label</td>
</tr>
<tr>
<td>Ungroup</td>
<td>Group: event.severity_label</td>
</tr>
<tr>
<td><strong>5,292 ET HUNTING Suspicious HTTP Request to .bit domain</strong></td>
<td>Group: event.severity_label</td>
</tr>
</tbody>
</table>
4.1.3 Time Picker

By default, Alerts searches the last 24 hours. If you want to search a different time frame, you can change it in the upper right corner of the screen.

4.1.4 Data Table

The remainder of the page is a data table that starts in the grouped view and can be switched to the detailed view. Both views have some functionality in common:

- Clicking the table headers allows you to sort ascending or descending.
- Clicking the bell icon acknowledges an alert. That alert can then be seen by selecting the Acknowledged toggle at the top of the page. In the Acknowledged view, clicking the bell icon removes the acknowledgement.
- Clicking the blue exclamation icon escalates the alert to TheHive and creates a case. The case can then be seen in TheHive interface. If you need to find that original escalated alert in the Alerts page, you can enable the Escalated toggle (which will automatically enable the Acknowledged toggle as well).
- Clicking a value in the table brings up a menu of actions for that value. The plus and minus magnifying glass icons to the left allow you to include or exclude (respectively) those values in your query. If present, the down arrow icon allows you to drill into that value (more on that in the next section). The groupby icon allows to add that particular field as a groupby in the query. The crosshair icon starts a new search for the value in Hunt. The G and VT on the right end of the actions menu look up the value at Google and VirusTotal (respectively).
- You can adjust the Rows per page setting in the bottom right and use the left and right arrow icons to page through the table.

Grouped View

By default, alerts are grouped by whatever criteria is selected in the query bar. Clicking a field value and then clicking the down arrow icon allows you to drilldown into that value which switches to the detailed view.
If you click a value in the grouped view and then click the down arrow icon on the quick bar, it will drill down to the detailed view. This shows all search results and allows you to then drill into individual search results as necessary. Clicking the table headers allows you to sort ascending or descending. Starting from the left side of each row, there is an arrow which will expand the result to show all of its fields. To the right of that arrow is the Timestamp field. Next, a few standard fields are shown: rule.name, event.severity_label, source.ip, source.port, destination.ip, and destination.port. Depending on what kind of data you’re looking at, there may be some additional data-specific fields as well.

When you click the down arrow to expand a row in the Events table, it will show all of the individual fields from that event. Field names are shown on the left and field values on the right. When looking at the field names, there is an icon to the left that will add that field to the groupby section of your query. You can click on values on the right to bring up the action menu to refine your search or pivot to other pages.

Show PCAP for this event

@timestamp 2020-10-01T10:15:34.590Z
destination.geo.continent_name Asia
destination.geo.country_iso_code HK
destination.geo.country_name Hong Kong
destination.geo.ip 58.64.132.141
destination.geo.location.lat 22.25
destination.geo.location.lon 114.1657
destination.geo.timezone Asia/Hong Kong
destination.ip 58.64.132.141
destination.port
ecs.version
event.category network
4.2 Hunt

*Security Onion Console (SOC)* gives you access to our new Hunt interface. This interface allows you to hunt through all of the data in *Elasticsearch* and is highly tuned for stacking, pivoting, data expansion, and data reduction.

4.2.1 Auto Hunt

The top of the page has a toggle for Auto Hunt which defaults to enabled. When enabled, Hunt will automatically submit your query any time you change filters, groupings, or date ranges.

4.2.2 Query Bar

The easiest way to get started is to click the query drop down box and select one of the pre-defined queries. These pre-defined queries cover most of the major data types that you would expect to see in a Security Onion deployment: NIDS alerts from *Suricata*, HIDS alerts from *Wazuh*, protocol metadata logs from *Zeek* or *Suricata*, endpoint logs, and firewall logs. Each of the entries in the drop down list will show the actual query followed by a description of what that query does.
4.2.3 Time Picker

By default, Hunt searches the last 24 hours. If you want to search a different time frame, you can change it in the upper right corner of the screen. You can use the default relative time or click the clock icon to change to absolute time.

4.2.4 Visualization

The first section of output contains a Most Occurrences visualization, a timeline visualization, and a Fewest Occurrences visualization. Bar charts are clickable, so you can click a value to update your search criteria. Aggregation defaults to 10 values, so Most Occurrences is the Top 10 and Fewest Occurrences is the Bottom 10 (long tail). The number of aggregation values is controlled by the Fetch Limit setting in the Group Metrics section.
4.2.5 Group Metrics

The middle section of output is the Group Metrics section and it’s a data table that allows you to stack (aggregate) arbitrary fields. Group metrics are controlled by the `groupby` parameter in the search bar. Clicking the table headers allows you to sort ascending or descending.

Clicking a value in the Group Metrics table brings up a menu of actions for that value. The plus and minus magnifying glass icons to the left allow you to include or exclude (respectively) those values in your query. The third magnifying glass starts a new query for just the value itself. The G and VT on the right end of the actions menu look up the value at Google and VirusTotal (respectively).

The default Fetch Limit for the Group Metrics table is 10. If you need to see more than the top 10, you can increase the Fetch Limit and then page through the output using the left and right arrow icons or increase the Rows per page setting.

<table>
<thead>
<tr>
<th>Group Metrics</th>
<th>Fetch Limit</th>
<th>Filter Results</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

4.2.6 Events

The third and final section of output is a data table that contains all search results and allows you to drill into individual search results as necessary. Clicking the table headers allows you to sort ascending or descending. Starting from the left side of each row, there is an arrow which will expand the result to show all of its fields. To the right of that arrow is the Timestamp field. Next, a few standard fields are shown: source.ip, source.port, destination.ip, destination.port, log.id.uid (Zeek unique identifier), network.community_id (Community ID), and event.dataset. Depending on what kind of data you’re looking at, there may be some additional data-specific fields as well.

Clicking a value in the Events table brings up a menu of actions for that value. The plus and minus magnifying glass icons to the left allow you to include or exclude (respectively) those values in your query. The third magnifying glass starts a new query for just the value itself. The fourth icon takes you to pcap for the stream. The fifth icon (bell) creates an alert for the event. The G and VT on the right end of the actions menu look up the value at Google and VirusTotal (respectively).

The default Fetch Limit for the Events table is 100. If you need to see more than 100 events, you can increase the Fetch Limit and then page through the output using the left and right arrow icons or increase the Rows per page setting.
When you click the down arrow to expand a row in the Events table, it will show all of the individual fields from that event. Field names are shown on the left and field values on the right. When looking at the field names, there is an icon to the left that will add that field to the groupby section of your query. You can click on values on the right to bring up the action menu to refine your search or pivot to other pages.
4.2.7 Statistics

The bottom left corner of the page shows statistics about the current query including the speed of the backend data fetch and the total round trip time.

The backend data fetch took 0.035 seconds. The total round trip took 0.06 seconds.

4.2.8 OQL

Onion Query Language (OQL) starts with standard Lucene query syntax and then allows you to add optional segments that control what Hunt does with the results from the query. The `groupby` segment tells Hunt to group by (aggregate) a particular field. So, for example, if you want to group by destination IP address, you can add `| groupby destination.ip` to your search (assuming it didn’t already have a groupby statement). The `groupby` segment supports multiple aggregations so you can add more fields that you want to group by, separating those fields with spaces. For example, to group by destination IP address and then destination port, you could use `| groupby destination.ip destination.port`.

4.2.9 Videos

See also:

To see Hunt in action, check out these Youtube videos:

https://www.youtube.com/watch?v=TZ96aBEVhFU
https://www.youtube.com/watch?v=0bwwZyedqdA
https://www.youtube.com/watch?v=Is2shLAOyJs
https://www.youtube.com/watch?v=Y-nZInToH8s

4.3 PCAP

*Security Onion Console (SOC)* gives you access to our new PCAP interface. This interface allows you to access your full packet capture that was recorded by *Stenographer*.

You can pivot to PCAP from *Alerts, Hunt*, and *Kibana*. Alternatively, you can go directly to the PCAP interface and then put in your search criteria to search for a particular stream.
Security Onion will then locate the stream and render a high level overview of the packets.
If there are many packets in the stream, you can use the LOAD MORE button, Rows per page setting, and arrows to navigate through the list of packets.

You can drill into individual rows to see the actual payload data. There are buttons at the top of the table that control what data is displayed in the individual rows. By disabling Show all packet data and HEX, we can get an ASCII transcript.

Finally, you can also download the pcap by clicking the button on the right side of the table header.

### 4.4 Kibana

From https://www.elastic.co/kibana:

Kibana is a free and open user interface that lets you visualize your Elasticsearch data and navigate the Elastic Stack. Do anything from tracking query load to understanding the way requests flow through your apps.
4.4.1 Screenshot

![Screenshot of Security Onion console](image)

4.4.2 Diagnostic Logging

Kibana logs to `/opt/so/log/kibana/kibana.log`.

4.4.3 Dashboards

We’ve included the old 16.04 dashboards in case you have any old 16.04 data. The new Security Onion 2 dashboards are all named with the `Security Onion` prefix and they should be used for any new data going forward.

If you ever need to reload dashboards, you can run the following command on your manager:

```
so-kibana-config-load
```

4.4.4 Pivoting

Kibana uses multiple hyperlinked fields to accelerate investigations and decision-making:

**Transcript**

When present, clicking the hyperlinked `_id` field allows an analyst to pivot to full packet capture via our PCAP interface. You can usually find the `_id` field as the rightmost column in the log panels at the bottom of the dashboards:
You can also find the `_id` field by drilling into a row in the log panel.

<table>
<thead>
<tr>
<th>Time</th>
<th>source.ip</th>
<th>source.port</th>
<th>destination.ip</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct 7, 2020 @ 14:59:09.213</td>
<td>192.168.6.10</td>
<td>53209</td>
<td>192.168.129.36</td>
</tr>
</tbody>
</table>

Indicator Dashboard

Several fields are hyperlinked to the Indicator dashboard to allow you to get all the information you can about a particular indicator. Here are just a few:

- `uid`
- `source.ip`
- `source.port`
- `destination.ip`
- `destination.port`

### 4.4.5 Search Results

Search results in the dashboards and through Discover are limited to the first 10 results for a particular query. If you don’t feel like this is adequate after narrowing your search, you can adjust the value for `discover:sampleSize`. 

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>@timestamp</code></td>
<td>Oct 7, 2020 @ 14:59:09.213</td>
</tr>
<tr>
<td><code>Push to TheHive</code></td>
<td>Click to create a case in TheHive</td>
</tr>
<tr>
<td><code>_id</code></td>
<td>okhuBHU93PK9naqemKw</td>
</tr>
</tbody>
</table>
in Kibana by navigating to Stack Management -> Advanced Settings and changing the value. It may be best to change this value incrementally to see how it affects performance for your deployment.

### 4.4.6 Timestamps

By default, Kibana will display timestamps in the timezone of your local browser. If you would prefer timestamps in UTC, you can go to Management -> Advanced Settings and set `dateFormat:tz` to UTC.

### 4.4.7 Configuration

Kibana’s configuration can be found in `/opt/so/conf/kibana/`. However, please keep in mind that most configuration is managed with Salt, so if you manually make any modifications in `/opt/so/conf/kibana/`, they may be overwritten at the next salt update.

### 4.4.8 More Information

See also:

For more information about Kibana, please see [https://www.elastic.co/kibana](https://www.elastic.co/kibana).

### 4.5 Grafana

Once you’ve logged into Security Onion Console (SOC), you can then click the Grafana link to see system health information.
On a distributed deployment, you will default to the manager dashboard. There are also dashboards for other node types. Once you’ve accessed the node dashboards, they should be added to Recently viewed dashboards which is accessible by simply clicking the Dashboards icon.
4.5.1 Accounts

By default, you will be viewing Grafana as an anonymous user. If you want to make changes to the default Grafana dashboards, you will need to log into Grafana with username admin and the randomized password found via sudo salt-call pillar.get secrets.

4.5.2 Configuration

Grafana configuration can be found in /opt/so/conf/grafana/etc/. However, please keep in mind that most configuration is managed with Salt, so if you manually make any modifications in /opt/so/conf/grafana/etc/, they may be overwritten at the next salt update. The default configuration options can be seen in /opt/so/saltstack/default/salt/grafana/defaults.yaml. Any options not specified in here, will use the Grafana default.

If you want to configure and enable SMTP for Grafana, place the following in the global.sls file. If you have files referenced in the config file, those can be placed in /opt/so/saltstack/default/salt/grafana/etc/files/. Those files will be then be placed in /opt/so/conf/grafana/etc/files on the minion and mapped to /etc/grafana/config/files/ within the container.

grafana:
  config:
    smtp:
      enabled: true
      host: smtphost.mydomain:25
      user: myuser
      # If the password contains # or ; you have to wrap it with triple quotes,
      # wrapped by single quotes. Ex """"#password;"""
      password: mypassword
      # cert_file: /etc/grafana/config/files/smtp_cert_file.crt

(continues on next page)
4.5.3 More Information

See also:

For more information about Grafana, please see https://grafana.com/.

4.6 CyberChef

From https://github.com/gchq/CyberChef:

The Cyber Swiss Army Knife
CyberChef is a simple, intuitive web app for carrying out all manner of “cyber” operations within a web browser. These operations include simple encoding like XOR or Base64, more complex encryption like AES, DES and Blowfish, creating binary and hexdumps, compression and decompression of data, calculating hashes and checksums, IPv6 and X.509 parsing, changing character encodings, and much more.

The tool is designed to enable both technical and non-technical analysts to manipulate data in complex ways without having to deal with complex tools or algorithms. It was conceived, designed, built and incrementally improved by an analyst in their 10% innovation time over several years. Every effort has been made to structure the code in a readable and extendable format, however it should be noted that the analyst is not a professional developer.
4.6.1 Screenshot

4.6.2 Accessing

To access CyberChef, log into Security Onion Console (SOC) and click the CyberChef hyperlink.

4.6.3 More Information

See also:

For more information about CyberChef, please see https://github.com/gchq/CyberChef.

4.7 Playbook

4.7.1 Overview

Playbook is a web application available for installation on Manager nodes. Playbook allows you to create a Detection Playbook, which itself consists of individual Plays. These Plays are fully self-contained and describe the different aspects around a particular detection strategy.
The key components of a Play are:

1. Objective & Context - what exactly are we trying to detect and why?
2. What are the follow-up actions required to validate and/or remediate when results are seen?
3. The actual query needed to implement the Play’s objective. In our case, the ElastAlert / Elasticsearch configuration.

Any results from a Play (low, medium, high, critical severity) are available to view within Hunt or Kibana. High or critical severity results from a Play will generate an Alert within TheHive.

The final piece to Playbook is automation. Once a Play is made active, the following happens:

- The required ElastAlert config is put into production
- Case Template for TheHive is created
- ATT&CK Navigator layer is updated to reflect current coverage

### 4.7.2 Getting Started

You can access Playbook by logging into Security Onion Console (SOC) and clicking the Playbook link. You will see over 500 plays already created that have been imported from the Sigma Community repository of rules at https://github.com/Neo23x0/sigma/tree/master/rules.

### 4.7.3 Creating a new Play

Plays are based on Sigma rules - from https://github.com/Neo23x0/sigma:
Security Onion Documentation, Release 2.3

Sigma is a generic and open signature format that allows you to describe relevant log events in a straightforward manner. The rule format is very flexible, easy to write and applicable to any type of log file. The main purpose of this project is to provide a structured form in which researchers or analysts can describe their once developed detection methods and make them shareable with others.

To create a new play, click on the Sigma Editor menu link. Either Load a sample Sigma rule or paste one into the Sigma field and click Convert. This will convert the Sigma into a query that you can use in Hunt or Kibana to confirm that it will work for your target log.

Refer to Log Sources & Field Names for details around what field names to use in the Sigma etc.

Once you are ready to create the Play, click Create Play From Sigma. If the Play creation is successful, you will be redirected to the newly created Play - it will have a status of Draft.

The lifecycle of a Play is as follows:

1. Draft (Initial state)
2. Active (In Production)
3. Inactive (Temporarily moved out of production)
4. Archived (Play has been superseded/retired)

A Play can also have the status of Disabled, which means that it is broken in some way and should not be made Active.

4.7.4 Editing a Play

Click on Edit to edit a Play. There will only be a few fields that you can modify - to make edits to the others (Title, Description, etc), you will need to edit the Sigma inside the Sigma field. Keep in mind that the Sigma is YAML formatted, so if you have major edits to make it is recommended to lint it and/or Convert it through the Sigma Editor to confirm that it is formatted correctly. Be sure to remove the prepended and postpended Playbook-specific syntax highlighting before linting/converting.{{\template{View Sigma}<\pre><\code class="yaml">

Once you save your changes, Playbook will update the rest of the fields to match your edits, including regenerating the Elastalert rule if needed.

4.7.5 Putting a Play into Production

When you are ready to start alerting on your Play, change the Status of the play to Active. This will create TheHive case template and the ElastAlert config. Any edits made to the Play in Playbook will automatically update the ElastAlert configuration and TheHive case template.

The Elastalert rules are located under /opt/so/rules/elastalert/playbook/<PlayID>.yml. Elastalert rules created by Playbook will run every 3 minutes, with a buffer_time of 15 minutes.

Performance testing is still ongoing. Initial testing has shown that on a lightly-used Standalone install with 16GB of RAM (4GB allocated to the Elasticsearch Heap), 300 Plays can be active without issues. We recommend avoiding the Malicious Nishang PowerShell Commandlets play as it can cause serious performance problems. You may also want to avoid others with a status of experimental.

4.7.6 Viewing Playbook Alerts

When results from your Plays are found (ie alerts), they are available to view within Alerts.
4.7.7 User Accounts

By default, once a user has authenticated through SOC they can access Playbook without having to login again to the app itself. This anonymous access has the permissions of the analyst role. If you need your team to login with individual user accounts, you can disable this anonymous access and create new user accounts and add them to the analyst group which will give them all the relevant permissions.

If you need administrator access to Playbook, you can login as admin with the randomized password found via `sudo salt-call pillar.get secrets`. However, the Playbook UI is designed to be used with a user that has an analyst role. Using an admin account will be very confusing to newcomers to Playbook, since many of the fields will now be shown/editable and it will look much more cluttered.

4.7.8 Misc Notes

`so-playbook-sync` runs every 5 minutes. This script queries Playbook for all active plays and then checks to make sure that there is an ElastAlert config and TheHive case template for each play. It also runs through the same process for inactive plays.

There is currently a bug when it comes to disabling plays. If you disable plays in the web interface but they continue to run, you may need to manually delete the yaml files in `/opt/so/rules/elastalert/playbook/`. Then restart ElastAlert as follows:

```
sudo so-elastalert-restart --force
```

4.7.9 Log Sources & Field Names

Sigma support currently extends to the following log sources in Security Onion:

- `osquery`
- network (via Zeek logs)
- Windows Eventlogs and Sysmon (shipped with `osquery` or winglobeat)

The pre-loaded Plays depend on Sysmon and Windows Eventlogs shipped with winlogbeat or osquery.

For best compability, use the following Sigma Taxonomy:

- Process Creation: https://github.com/Neo23x0/sigma/wiki/Taxonomy#process-creation-events
- Network: https://github.com/Neo23x0/sigma/wiki/Taxonomy#specific

The current Security Onion Sigmac field mappings can be found here: https://github.com/Security-Onion-Solutions/securityonion-image/blob/master/soctopus/soctopus/playbook/securityonion-baseline.yml

4.7.10 Adding Additional Rulesets

As previously mentioned, the pre-loaded Plays come from the community Sigma repository (https://github.com/Neo23x0/sigma/tree/master/rules). The default config is to only pull in the Windows rules. The rest of the rules from the community repository can be pulled in by editing a pillar value under `/opt/so/saltstack/local/pillar/global.sls`

```
soctopus:
  playbook:
    rulesets:
```

4.7. Playbook
Security Onion Documentation, Release 2.3

- windows

Add one or more of the following:

application, apt, cloud, compliance, generic, linux, network, proxy, web

These are based on the top level directories from the Sigma community repository rule’s folder.

Next, restart SOCTopus (so-soctopus-restart) and have Playbook pull in the new rules with so-playbook-ruleupdate - this can take a few minutes to complete if pulling in a large amount of new rules.

4.7.11 Diagnostic Logging

Playbook logs can be found in /opt/so/log/playbook/.

4.8 Fleet

From https://www.kolide.com/fleet/:

Expand osquery capabilities from a single machine to your entire fleet. Query dynamic sets of hosts, and watch the data stream in for immediate analysis and investigation. Export results for a closer look in your favorite tools.

4.8.1 Usage

If you selected to enable Fleet during the setup, you can now login to Fleet using the email address and password that you entered during the installer. You can edit the password or add a new Fleet user within Fleet itself.

Custom osquery packages were generated for you during setup and you can find them under Downloads in Security Onion Console (SOC). Before you install a package on an endpoint, use sudo so-allow on your manager node to configure the SO firewall to allow inbound osquery connections.

4.8.2 Configuration

Fleet configuration can be found in /opt/so/conf/fleet/. However, please keep in mind that if you make any changes to this directory they may be overwritten since the configuration is managed with Salt.
4.8.3 Diagnostic Logging

Fleet logs can be found in `/opt/so/log/fleet/`.

4.8.4 More Information

See also:
For more information about osquery, please see the osquery section.
For more information about Fleet, please see https://www.kolide.com/fleet/.

4.9 TheHive

From https://thehive-project.org/:

A scalable, open source and free Security Incident Response Platform, tightly integrated with MISP (Malware Information Sharing Platform), designed to make life easier for SOCs, CSIRTs, CERTs and any information security practitioner dealing with security incidents that need to be investigated and acted upon swiftly.

4.9.1 Usage

As you are working in Alerts, Hunt, or Kibana, you may find alerts or logs that are interesting enough to send to TheHive and create a case. Other analysts can collaborate with you as you work to close that case.

In Alerts and Hunt, you can use the blue triangle with an exclamation point to escalate to TheHive.
Clicking the escalate button will escalate the data from the row as it is displayed. This means that if you’re looking at an aggregated view, you will get limited details in the resulting escalated case. If you want more details to be included in the case, then first drill into the aggregation and escalate one of the individual items in that aggregation.

In Kibana you will see a scripted field named *Push to TheHive* with a value of *Click to create a case in TheHive*. This will use the API to add this new event to *TheHive*.

<table>
<thead>
<tr>
<th>Table</th>
<th>JSON</th>
</tr>
</thead>
<tbody>
<tr>
<td>☐ timestamp</td>
<td>Oct 7, 2020 @ 14:59:09.213</td>
</tr>
<tr>
<td>☐ Push to TheHive</td>
<td>Click to create a case in TheHive</td>
</tr>
</tbody>
</table>

### 4.9.2 Configuration

TheHive reads its configuration from /opt/so/conf/thehive/. However, please keep in mind that if you make any changes to this directory they may be overwritten since the configuration is managed with *Salt*.

### 4.9.3 Diagnostic Logging

TheHive logging can be found at /opt/so/log/thehive/.

### 4.9.4 More Information

**See also:**

For more information about TheHive, please see https://thehive-project.org/.

### 4.10 ATT&CK Navigator

From https://github.com/mitre-attack/attack-navigator:

The ATT&CK Navigator is designed to provide basic navigation and annotation of ATT&CK matrices, something that people are already doing today in tools like Excel. We’ve designed it to be simple and generic - you can use the Navigator to visualize your defensive coverage, your red/blue team planning, the frequency of detected techniques or anything else you want to do. The Navigator doesn’t care - it just allows you to manipulate the cells in the matrix (color coding, adding a comment, assigning a numerical
value, etc.). We thought having a simple tool that everyone could use to visualize the matrix would help make it easy to use ATT&CK.

The principal feature of the Navigator is the ability for users to define layers - custom views of the ATT&CK knowledge base - e.g. showing just those techniques for a particular platform or highlighting techniques a specific adversary has been known to use. Layers can be created interactively within the Navigator or generated programmatically and then visualized via the Navigator.

### 4.10.1 Accessing

To access Navigator, log into Security Onion Console (SOC) and then click the Navigator hyperlink on the left side.

### 4.10.2 Default Layer - Playbook

The default layer is titled Playbook and is automatically updated when a Play from Playbook is made active/inactive. This allows you to see your Detection Playbook coverage across the ATT&CK framework.

Right-clicking any Technique and selecting View Related Plays will open Playbook with a pre-filtered view of any plays that are tagged with the selected Technique.

### 4.10.3 Configuration

Navigator reads its configuration from /opt/so/conf/navigator/. However, please keep in mind that if you make any changes here they may be overwritten since the config is managed with Salt.
4.10.4 More Information

See also:

For more information about ATT&CK Navigator, please see https://github.com/mitre-attack/attack-navigator.
Full-time analysts may want to create a dedicated Analyst VM. This allows you to investigate pcaps and other potentially malicious artifacts without impacting your Security Onion deployment or your workstation.

Starting in Security Onion 2.3, the `so-analyst-install` script will install a full GNOME desktop environment including Chromium web browser, NetworkMiner, Wireshark, and other analyst tools. `so-analyst-install` is totally independent of the standard setup process, so you can run it before or after setup or not run setup at all if all you really want is the Analyst VM itself.
To connect from the Analyst VM to your manager node, you will need to run `so-allow` on the manager node and choose the analyst option to allow the traffic through the host-based Firewall.

5.1 NetworkMiner

From https://www.netresec.com/?page=networkminer:

NetworkMiner is an open source Network Forensic Analysis Tool (NFAT) for Windows (but also works in Linux / Mac OS X / FreeBSD). NetworkMiner can be used as a passive network sniffer/packet capturing tool in order to detect operating systems, sessions, hostnames, open ports etc. without putting any traffic on the network. NetworkMiner can also parse PCAP files for off-line analysis and to regenerate/reassemble transmitted files and certificates from PCAP files.

NetworkMiner makes it easy to perform advanced Network Traffic Analysis (NTA) by providing extracted artifacts in an intuitive user interface. The way data is presented not only makes the analysis simpler, it also saves valuable time for the analyst or forensic investigator.

5.1.1 Usage

NetworkMiner is a part of our Analyst VM installation. `so-analyst-install` automatically registers NetworkMiner as a pcap handler, so if you download a pcap file from PCAP, you can simply click on it to open in NetworkMiner.
5.1.2 Screenshot

![NetworkMiner Screenshot](image)

5.1.3 More Information

See also:

For more information about NetworkMiner, please see [https://www.netresec.com/?page=networkminer](https://www.netresec.com/?page=networkminer).

5.2 Wireshark

From [https://www.wireshark.org/](https://www.wireshark.org/):

Wireshark is the world’s foremost and widely-used network protocol analyzer. It lets you see what’s happening on your network at a microscopic level and is the de facto (and often de jure) standard across many commercial and non-profit enterprises, government agencies, and educational institutions. Wireshark development thrives thanks to the volunteer contributions of networking experts around the globe and is the continuation of a project started by Gerald Combs in 1998.

5.2.1 Usage

Wireshark is a part of our Analyst VM installation.
5.2.2 Screenshot

See also:

For more information about Wireshark, please see https://www.wireshark.org/.
When you log into Security Onion Console (SOC), you may see network-based IDS alerts from Suricata, protocol metadata logs from Zeek, file analysis logs from Strelka, or full packet capture from Stenographer. How is that data generated and stored? This section covers the various processes that Security Onion uses to analyze and log network traffic.
6.1 AF-PACKET

AF-PACKET is built into the Linux kernel and includes fanout capabilities enabling it to act as a flow-based load balancer. This means, for example, if you configure Suricata for 4 AF-PACKET threads then each thread would receive about 25% of the total traffic that AF-PACKET is seeing.

Warning: If you try to test AF-PACKET fanout using tcpreplay locally, please note that load balancing will not work properly and all (or most) traffic will be handled by the first worker in the AF-PACKET cluster. If you need to test AF-PACKET load balancing properly, you can run tcpreplay on another machine connected to your AF-PACKET machine.

The following processes use AF-PACKET for packet acquisition:

Stenographer
Suricata
Zeek
6.1.1 More Information

See also:
For more information about AF-PACKET, please see https://www.kernel.org/doc/Documentation/networking/packet_mmap.txt.

6.2 Stenographer

From https://github.com/google/stenographer:

Stenographer is a full-packet-capture utility for buffering packets to disk for intrusion detection and incident response purposes. It provides a high-performance implementation of NIC-to-disk packet writing, handles deleting those files as disk fills up, and provides methods for reading back specific sets of packets quickly and easily.

Stenographer uses AF-PACKET for packet acquisition.

6.2.1 Output

Stenographer writes full packet capture to /nsm/pcap/. It will automatically start purging old data once the partition reaches 90%.

6.2.2 Analysis

You can access full packet capture via PCAP:

Alerts, Hunt, and Kibana allow you to easily pivot to the PCAP page.
6.2.3 Configuration

Stenographer reads its configuration from `/opt/so/conf/steno/`. However, please keep in mind that if you make any changes to this directory they may be overwritten since the configuration is managed with Salt.

6.2.4 Diagnostic Logging

Diagnostic logging for Stenographer can be found at `/opt/so/log/stenographer/`.

6.2.5 Disabling

If you need to disable Stenographer, you can set the Salt pillar `steno:enabled:false` in the `global.sls` or in the sensor’s `minion.sls` file.

6.2.6 More Information

See also:

For more information about stenographer, please see https://github.com/google/stenographer.

6.3 Suricata

From https://suricata-ids.org:

Suricata is a free and open source, mature, fast and robust network threat detection engine. Suricata inspects the network traffic using a powerful and extensive rules and signature language, and has powerful Lua scripting support for detection of complex threats.

Suricata NIDS alerts can be found in Alerts, Hunt, and Kibana. Here’s an example of Suricata NIDS alerts in Alerts:
If enabled, Suricata metadata (protocol logs) can be found in *Hunt* and *Kibana*.

### 6.3.1 Community ID

Security Onion enables Suricata’s native support for *Community ID*.

### 6.3.2 Performance

Suricata uses *AF-PACKET* to allow you to spin up multiple workers to handle more traffic.

To change the number of Suricata workers:

- Stop sensor processes:
  ```bash
sudo so-suricata-stop
  ```
- Edit `/opt/so/saltstack/local/pillar/minions/$SENSORNAME_$ROLE.sls` and change the `suriprocs` variable to the desired number of workers.
- Start sensor processes:
  ```bash
  sudo so-suricata-start
  ```

For best performance, Suricata processes should be pinned to specific CPUs. In most cases, you’ll want to pin sniffing processes to the same CPU that your sniffing NIC is bound to. You can use the affinity settings in `suricata.yaml` as shown in https://suricata.readthedocs.io/en/latest/configuration/suricata-yaml.html#threading.


### 6.3.3 Configuration

You can configure Suricata’s `suricata.yaml` using *Salt*. The defaults for this have been defined in https://github.com/Security-Onion-Solutions/securityonion/blob/master/salt/suricata/defaults.yaml. Under `suricata:config`, the pillar structure follows the same YAML structure of the `suricata.yaml` file.

For example, suppose you want to change Suricata’s `EXTERNAL_NET` setting from the default of `any` to `!$HOME_NET`. You could add the following to the global pillar file (`/opt/so/saltstack/local/pillar/global.sls`) or minion pillar file (`/opt/so/saltstack/local/pillar/minions/$SENSORNAME_$ROLE.sls`) on the manager:

```yaml
suricata:
  config:
    vars:
      address-groups:
        EXTERNAL_NET: "!$HOME_NET"
```

From the manager, then run:
```
sudo salt $SENSORNAME_$ROLE state.highstate
```

Some of the settings normally found in `suricata.yaml` can be found in the sensor pillar instead of the Suricata pillar. These options are: `HOMENET`, `default-packet-size`, and the CPU affinity settings for pinning the processes to CPU cores or how many processes to run.

If you would like to configure/manage IDS rules, please see the *Managing Rules* and *Managing Alerts* sections.

6.3. Suricata

85
6.3.4 Thresholding

To enable thresholding for SIDS, reference the example pillar at https://github.com/Security-Onion-Solutions/securityonion/blob/master/pillar/thresholding/pillar.example.


This pillar can be added to Salt in either the global pillar file (/opt/so/saltstack/local/pillar/global.sls) or minion pillar file (/opt/so/saltstack/local/pillar/minions/$SENSORNAME_$ROLE.sls).

6.3.5 Diagnostic Logging

If you need to troubleshoot Suricata, check /opt/so/log/suricata/suricata.log.

6.3.6 Stats

For detailed Suricata statistics, check /opt/so/log/suricata/stats.log.

6.3.7 More Information

See also:

For more information about Suricata, please see https://suricata-ids.org.

6.4 Zeek

Zeek is formerly known as Bro. From https://www.zeek.org/:

Zeek is a powerful network analysis framework that is much different from the typical IDS you may know. (Zeek is the new name for the long-established Bro system. Note that parts of the system retain the “Bro” name, and it also often appears in the documentation and distributions.)

Zeek logs are sent to Elasticsearch for parsing and storage and can then be found in Hunt and Kibana. Here’s an example of Zeek conn (connection) logs in Hunt:
6.4.1 Community ID

Security Onion enables Zeek’s native support for **Community ID**.

6.4.2 Performance

Zeek uses **AF-PACKET** so that you can spin up multiple Zeek workers to handle more traffic.

To change the number of AF-PACKET workers for **Zeek**:

- **Stop Zeek**:
  ```
  sudo so-zeek-stop
  ```

- **Edit** `/opt/so/saltstack/local/pillar/minions/$SENSORNAME_$ROLE.sls` and change the `zeek_lbprocs` variable to the desired number of cores.

- **Start Zeek**:
  ```
  sudo so-zeek-start
  ```

For best performance, Zeek should be pinned to specific CPUs. In most cases, you’ll want to pin sniffing processes to the same CPU that your sniffing NIC is bound to. You can do this using the `pin_cpus` setting as shown at [https://docs.zeek.org/en/stable/configuration/#using-pf-ring](https://docs.zeek.org/en/stable/configuration/#using-pf-ring).
6.4.3 Email

To configure email notifications, please see the Email Configuration section.

6.4.4 Syslog

To forward Zeek logs to an external syslog collector, please see the Syslog Output section.

6.4.5 Intel

You can add your own intel to `/opt/so/saltstack/local/salt/zeek/policy/intel/intel.dat` on the manager and then run `sudo salt $SENSORNAME_$ROLE state.highstate`. When writing this file, ensure there are no leading/trailing spaces or lines, and that only a single tab is used to separate fields. If you experience an error, or do not notice `/nsm/zeek/logs/current/intel.log` being generated, try having a look in `/nsm/zeek/logs/current/reporter.log` for clues. You may also want to restart Zeek after making changes by running `sudo so-zeek-restart`.

For more information, please see:

https://github.com/weslambert/securityonion-misp

6.4.6 Custom Scripts

Custom scripts can be added to `/opt/so/conf/zeek/policy/custom/<$custom-module>` on the manager. Once the script module is created, the configuration for `local.zeek` will need to be updated. In Security Onion 2, this configuration is abstracted into a SaltStack pillar. For example, we would copy `/opt/so/saltstack/default/pillar/zeek/init.sls` to `/opt/so/saltstack/local/pillar/zeek/init.sls`, and add our custom module to be loaded by Zeek (alternatively, the pillar could be modified in the `global.sls` file. More details can be found here here: https://docs.securityonion.net/en/latest/zeek.html#configuration):

```
local:
  '@load':
    - misc/loaded-scripts
    - tuning/defaults
    - misc/capture-loss
    - misc/stats
    - frameworks/software/vulnerable
    - frameworks/software/version-changes
    - protocols/ftp/software
    - protocols/smtp/software
    - protocols/ssh/software
    - protocols/http/software
    - protocols/dns/detect-external-names
    - protocols/ftp/detect
    - protocols/conn/known-hosts
    - protocols/conn/known-services
    - protocols/ssl/known-certs
    - protocols/ssl/validate-certs
    - protocols/ssl/log-hostcerts-only
```

(continues on next page)
One the configuration has been updated, Zeek can be restarted with `so-zeek-restart` on applicable nodes to pick up the changes. Finally, `/nsm/zeek/logs/current/loaded_scripts.log` can be checked to ensure the new module has been loaded.

Ex. `grep mynewmodule /nsm/zeek/logs/current/loaded_scripts.log`

### 6.4.7 Logs

Zeek logs are stored in `/nsm/zeek/logs`. They are collected by `Filebeat`, parsed by and stored in `Elasticsearch`, and viewable in `Hunt` and `Kibana`.

We configure Zeek to output logs in JSON format. If you need to parse those JSON logs from the command line, you can use `jq`.

If you want to specify what Zeek logs are ingested, you can use `so-zeek-logs`.

Zeek monitors your network traffic and creates logs, such as:

**conn.log**

- TCP/UDP/ICMP connections
- For more information, see:
  
  https://docs.zeek.org/en/latest/scripts/base/protocols/conn/main.zeek.html#type-Conn::Info

**dns.log**

- DNS activity
- For more information, see:
  

**ftp.log**

- FTP activity
- For more information, see:
http.log

- HTTP requests and replies
- For more information, see:

ssl.log

- SSL/TLS handshake info
- For more information, see:

notice.log

- Zeek notices
- For more information, see:

... and others, which can be researched here:

As you can see, Zeek log data can provide a wealth of information to the analyst, all easily accessible through Hunt or Kibana.

6.4.8 Configuration

You can use Salt to manage Zeek’s local.zeek, node.cfg and zeekctl.cfg:

local.zeek: The allowed options for this file are @load, @load-sigs and redef. An example of configuring this pillar can be seen below.

node.cfg: The pillar items to modify this file are located under the sensor pillar in the minion pillar file. The options that can be customized in the file include: interface, lb_procs, pin_cpus, and af_packet_buffer_size.

zeekctl.cfg: An example of customizing this can be seen below. The allowed options can be seen in https://github.com/Security-Onion-Solutions/securityonion/blob/dev/salt/zeek/files/zeekctl.cfg.jinja.

Here is an example of how we would modify local.zeek. We can see the default pillar assignments used for local.zeek in /opt/so/saltstack/default/pillow/zeek/init.sls. This file should never be modified as it could be updated in the future and any modification made would be overwritten. The global or minion pillar files should be used for making changes as they are stored in /opt/so/saltstack/local/, and that directory isn’t overwritten during a Security Onion code update.
In this file, there are two keys under zeek, `zeekctl` and `local`. We will be using `zeek:local` for this example since we are modifying the `zeek.local` file. We will address `zeek:zeekctl` in another example where we...
modify the `zeekctl.cfg` file.

Under `zeek:local`, there are three keys: `@load`, `@load-sigs`, and `redef`. In the pillar definition, `@load` and `@load-sigs` are wrapped in quotes due to the `@` character. Under each of the keys, there is a list of items that will be added to the `local.zeek` file with the appropriate directive of either `@load`, `@load-sigs` or `redef`. In order to modify either of the lists, the entire list must be redefined in either the global or minion pillar file.

If we have a node where `protocols/ssh/detect-bruteforcing` is generating a lot of noise and we want to tell Zeek to stop loading that script, we would do the following. Since we just want to turn it off for that specific node, we would open `/opt/so/saltstack/local/pillar/minions/<minionid>.sls`. At the bottom, we would append the following:

```zeek
zeek:
  local:
    '@load':
    - misc/loaded-scripts
    - tuning/defaults
    - misc/capture-loss
    - misc/stats
    - frameworks/software/vulnerable
    - frameworks/software/version-changes
    - protocols/ftp/software
    - protocols/smtp/software
    - protocols/ssh/software
    - protocols/http/software
    - protocols/dns/detect-external-names
    - protocols/ftp/detect
    - protocols/conn/known-hosts
    - protocols/conn/known-services
    - protocols/ssl/known-certs
    - protocols/ssl/validate-certs
    - protocols/ssl/log-hostcerts-only
    - protocols/ssh/geo-data
    - protocols/ssh/interesting-hostnames
    - protocols/http/detect-sqli
    - frameworks/files/hash-all-files
    - frameworks/files/detect-MHR
    - policy/frameworks/notice/extend-email/hostnames
    - ja3
    - hassh
    - intel
    - cve-2020-0601
    - securityonion/bpfconf
    - securityonion/communityid
    - securityonion/file-extraction
```

We redefined the `@load` list in the minion pillar file, but we left out the `protocols/ssh/detect-bruteforcing`. This will override the value defined in the `/opt/so/saltstack/default/pillar/zeek/init.sls` and the global pillar file if it is defined there, and prevent the script from being added to the `local.zeek` file. If we wanted to add a script to be loaded, then we would add out script to the list. Since we aren’t changing `@load-sigs` or `redef`, then we do not need to add them here. Once the file is saved, and the node checks in with the manager, the `local.zeek` file will be updated and the `so-zeek` docker container will be restarted.

Let’s see an example of how we would modify the `zeekctl.cfg` file. From the example above, we know that the default pillar values are set for `zeek` in `/opt/so/saltstack/default/pillar/zeek/init.sls`. The default pillar values for `zeekctl.cfg` are as follows:
zeek:
  zeekctl:
    MailTo: root@localhost
    MailConnectionSummary: 1
    MinDiskSpace: 5
    MailHostUpDown: 1
    LogRotationInterval: 3600
    LogExpireInterval: 0
    StatsLogEnable: 1
    StatsLogExpireInterval: 0
    StatusCmdShowAll: 0
    CrashExpireInterval: 0
    SitePolicyScripts: local.zeek
    LogDir: /nsm/zeek/logs
    SpoolDir: /nsm/zeek/spool
    CfgDir: /opt/zeek/etc
    CompressLogs: 1

For anything not defined here, Zeek will use its own defaults. The options that are allowed to be managed with the pillar can be found at https://github.com/Security-Onion-Solutions/securityonion/blob/master/salt/zeek/files/zeekctl.cfg.jinja.

In order to add or modify an option in zeekctl, we will need to modify either the global or minion pillar file. For example, if we wanted to turn log compression off and change the timeout for Broker communication events to 20 seconds globally, we would add the following to the global pillar file.

zeek:
  zeekctl:
    compresslogs: 0
    commtimeout: 20

Since zeek:zeekctl is a dictionary with dictionary values, we do not need to redefine the entire pillar here like we did for zeek:local above. Once the pillar file is saved and the node checks in with the manager, the zeekctl.cfg file will be updated and the so-zeek container will be restarted.

6.4.9 More Information

See also:

For more information about Zeek, please see https://www.zeek.org/.

6.5 Strelka

From https://github.com/target/strelka:

Strelka is a real-time file scanning system used for threat hunting, threat detection, and incident response. Based on the design established by Lockheed Martin’s Laika BOSS and similar projects (see: related projects), Strelka’s purpose is to perform file extraction and metadata collection at huge scale.

Zeek extracts files from network traffic to /nsm/zeek/extracted/complete/. Strelka then analyzes those files and they end up in /nsm/strelka/processed/.

You can find Strelka logs in Hunt and Kibana. Here’s an example of Strelka logs in Hunt:
6.5.1 Configuration

Strelka reads its configuration from `/opt/so/conf/strelka/`. However, please keep in mind that if you make any changes to this directory they may be overwritten since the configuration is managed with *Salt*.

6.5.2 More Information

See also:

For more information about Strelka, please see [https://github.com/target/strelka](https://github.com/target/strelka).
When you logged into Security Onion Console (SOC), you may have seen some host logs from Wazuh. Security Onion can also consume many other kinds of host logs as well. You can send logs to Security Onion via osquery, Beats, Wazuh, or Syslog.

For Windows endpoints, you can optionally augment the standard Windows logging with Sysmon and/or Autoruns. Those additional logs can then be transported by whatever mechanism you chose above.

7.1 osquery

From https://osquery.io/:

Osquery uses basic SQL commands to leverage a relational data-model to describe a device.

7.1.1 Fleet

Security Onion includes Kolide Fleet to manage your osquery deployment. For more information, please see the Fleet section.

7.1.2 Agents - Deployment

To deploy an osquery agent to an endpoint, go to the Security Onion Console (SOC) Downloads page and download the proper osquery agent for the operating system of that endpoint. Use so-allow to allow the osquery agent to connect to port 8090 on the manager. Then install the osquery agent and it should check into the manager and start showing up in Fleet.

Osquery will attempt to connect to the Manager via the Manager’s IP or Hostname - whichever was selected during the Manager setup. If the hostname is used, the endpoints need to be able to resolve that hostname to the Manager’s IP. See this value by running the following command on the Manager: sudo salt-call pillar.get global:url_base. If this value ever changes, the osquery packages under Downloads will need to be regenerated.
All the packages (except for the macOS PKG) are customized for the specific Grid they were downloaded from, and include all the necessary configuration to connect to that Grid. The macOS package is a stock Launcher package, and will require additional configuration once it has been deployed.

**7.1.3 Agents - Updating**

Security Onion uses Kolide Launcher as a management wrapper around Osquery. This allows for a simpler configuration as well as auto-updates of Launcher and Osquery from the Kolide TUF service. Launcher will check every hour to see if an update is available and, if so, will download and install it. This is the default configuration, but can be changed within the osquery Flags file.

In an airgap environment where the endpoints do not have Internet access, updated Osquery packages can be downloaded from the Security Onion Console and used to update the endpoints. Osquery packages are periodically updated on the Manager as new versions of Osquery are released.

**7.1.4 Agents - Troubleshooting**

Agent logs on Windows endpoints can be found under the Application channel in the Windows Eventlog - source is Launcher.

**7.1.5 Agents - Regenerating Install Packages**

To regenerate packages, run the following on the Manager (it will take up to 5 minutes to rebuild the packages):

```
sudo salt-call state.apply fleet.event_gen-packages
```

**7.1.6 Hunt or Kibana**

All osquery logs can be found by using the following query:

```
event.module: osquery
```

**Kibana Dashboard: Host Data --> Modules/Osquery**

This dashboard gives an overview of the osquery logs in the system. As long as the default osquery configuration is used, this dashboard should work out of the box regardless of how you schedule or name your queries and packs.

**7.1.7 Shipping Windows Eventlogs**

Windows Eventlogs from the local Windows system can be shipped with osquery to Security Onion. Current parsing support extends to core Windows Eventlog channels (Security, Application, System) as well as Sysmon under the default channel location. These logs will show up in Security Onion as event.dataset: windows_eventlog or event.dataset: sysmon.

- Confirm that you can successfully live query the logs: `SELECT * FROM windows_events limit 10;`
- Save a new query: Query -> Manage Queries -> Create New Query `SELECT * FROM windows_events;` -> Save
• Add the new query to a query pack that targets a Windows host - how often it should run depends on log volume on the local host; start off with 180 seconds, differential logging: Packs -> Manage Packs -> Select + Edit Pack (Modify Targets for Windows only if needed, Modify Logging options as needed)

• Save pack + Enable pack, if needed.


7.1.8 Community ID

We sponsored the development of Community ID support for osquery:
https://dactiv.lle/blog/correlate-osquery-network-connections/

7.1.9 More Information

See also:
For more information about osquery, please see https://osquery.io/.

7.2 Beats

We can use Elastic Beats to facilitate the shipping of endpoint logs to Security Onion’s Elastic Stack. Currently, testing has only been performed with Filebeat (multiple log types) and Winlogbeat (Windows Event logs).

Note: In order to receive logs from Beats, Security Onion must be running Logstash. Evaluation Mode and Import Mode do not run Logstash, so you’ll need Standalone or a full Distributed Deployment.

7.2.1 Encryption

Warning: Beats communication with Logstash is not encrypted by default. If you require encryption, please consult the appropriate Elastic documentation to configure the use of TLS.

7.2.2 so-allow

Run sudo so-allow and select the b option to allow your Beats agents to send their logs to Logstash port 5044/tcp.

7.2.3 Winlogbeat

Navigate to the Downloads page in Security Onion Console (SOC) and download the linked Winlogbeat agent. This will ensure that you get the correct version of Winlogbeat for your Elastic version. Please note that the hyperlink simply points to the standard Winlogbeat download from the Elastic site.

Install Winlogbeat and copy winlogbeat.example.yml to winlogbeat.yml if necessary. Then configure winlogbeat.yml as follows:
• Make sure that the setup.dashboards.enabled setting is commented out or disabled.
• Disable the output.elasticsearch output.
• Enable the output.logstash output and configure it to send logs to port 5044 on your management node.
• If you are shipping Sysmon logs, confirm that your Winlogbeat configuration does NOT use the Elastic Sysmon module as Security Onion will do all the necessary parsing.

7.2.4 Installation

To install a Beat, follow the instructions provided for the respective Beat, with the exception of loading the index template, as Security Onion uses its own template file to manage Beats fields.

Filebeat

Winlogbeat

If installing Filebeat on a Linux distribution, you will want to ensure that the service is started after a reboot. We can ensure this by running the following commands after install:

```
sudo update-rc.d filebeat defaults
sudo update-rc.d filebeat enable
```

7.2.5 Log files

Filebeat

Windows: C:\Program Files\Filebeat\filebeat.log
Linux: /var/log/filebeat/filebeat

Winlogbeat

C:\Program Files\Winlogbeat\winlogbeat.log

7.2.6 Data

In Kibana, you can find Beats data on the Host dashboard or by searching for _index:"*:so-beats-*" in Discover.

In Hunt, you can find Beats data by searching for _index:"*:so-beats-*".
7.3 Wazuh

7.3.1 Description

From https://wazuh.com/:

Wazuh is a free, open source and enterprise-ready security monitoring solution for threat detection, integrity monitoring, incident response and compliance.

7.3.2 Usage

Security Onion utilizes Wazuh as a Host Intrusion Detection System (HIDS) on each of the Security Onion nodes.

The Wazuh components include:

manager - runs inside of so-wazuh Docker container and performs overall management of agents
API - runs inside of so-wazuh Docker container and allows for remote management of agents, querying, etc.
agent - runs directly on each host and monitors logs/activity and reports to manager

The Wazuh API runs at TCP port 55000 locally, and currently uses the default credentials of user:foo and password:bar for authentication. Keep in mind, the API port is not exposed externally by default. Therefore, firewall rules need to be in place to reach the API from another location other than the Security Onion node on which the targeted Wazuh manager is running.

Since the manager runs inside a Docker container, many of the Wazuh binaries that you might want to run will need to be run inside the Docker container. For example, to run agent_upgrade:

```
sudo so-wazuh-agent-upgrade
```

7.3.3 Configuration

The main configuration file for Wazuh is /opt/so/conf/wazuh/ossec.conf.

7.3.4 Email

If you want to configure Wazuh to send email, please see the Email Configuration section.

7.3.5 Syslog

If you want to send Wazuh logs to an external syslog collector, please see the Syslog Output section.

7.3.6 Active Response

Sometimes, Wazuh may recognize legitimate activity as potentially malicious and engage in Active Response to block a connection. This may result in unintended consequences such as blocking of trusted IPs. To prevent this from occurring, you can add your IP address to a safe list and change other settings in /opt/so/conf/wazuh/ossec.conf in the <!-- Active response --> section. so-allow does this for you automatically when you allow analyst connections.
7.3.7 Tuning Rules

You can add new rules and modify existing rules in `/opt/so/rules/hids/local_rules.xml`.

7.3.8 Adding Agents

The Wazuh agent is cross platform and you can download agents for Windows/Unix/Linux/FreeBSD from the Wazuh website:

https://documentation.wazuh.com/3.13/installation-guide/packages-list/index.html

**Note:** It is important to ensure that you download the agent that matches the version of your Wazuh server. For example, if your Wazuh server is version 3.13.1, then you will want to deploy Wazuh agent version 3.13.1.

You can verify the version of your current Wazuh server using the following command:

```
sudo docker exec -it so-wazuh dpkg -l | grep wazuh
```

Once you’ve installed the Wazuh agent on the host(s) to be monitored, then perform the steps defined here:


Please keep in mind that when you run `manage_agents` you will need to do so inside the `so-wazuh` container like this:

```
sudo so-wazuh-agent-manage
```

You also may need to run `so-allow` to allow traffic from the IP address of your Wazuh agent(s).

7.3.9 Maximum Number of Agents

Security Onion is configured to support a maximum number of 14000 Wazuh agents reporting to a single Wazuh manager.

7.3.10 Automated Deployment

If you would like to automate the deployment of Wazuh agents, the Wazuh server includes `ossec-authd`. You can read more about `ossec-authd` at https://documentation.wazuh.com/3.13/user-manual/reference/daemons/ossec-authd.html.

When using `ossec-authd`, be sure to add a firewall exception for agents to access port 1515/tcp on the Wazuh manager node by running `so-allow` and choosing the `r` option.

7.3.11 More Information

See also:

For more information about Wazuh, please see https://documentation.wazuh.com/3.13/.
7.4 Syslog

If you want to collect syslog from other devices, you’ll need to run `so-allow` and then choose the `syslog` option to allow the port through the firewall.

If you need to add custom parsing for those syslog logs, we recommend using *Elasticsearch* ingest parsing.

7.5 Sysmon

From https://technet.microsoft.com/en-us/sysinternals/sysmon:

System Monitor (Sysmon) is a Windows system service and device driver that, once installed on a system, remains resident across system reboots to monitor and log system activity to the Windows event log. It provides detailed information about process creations, network connections, and changes to file creation time. By collecting the events it generates using Windows Event Collection or SIEM agents and subsequently analyzing them, you can identify malicious or anomalous activity and understand how intruders and malware operate on your network.

7.5.1 Integration

Josh Brower wrote a great paper on integrating sysmon into Security Onion:

(Please note that the paper is a few years old and was therefore written for an older version of Security Onion.)

7.5.2 Configuration

SwiftOnSecurity has a great sysmon config file to use as a starting point:
https://github.com/SwiftOnSecurity/sysmon-config

7.5.3 Downloads

Download sysmon here:
https://download.sysinternals.com/files/Sysmon.zip

Download SwiftOnSecurity’s example sysmon config here:

7.5.4 Winlogbeat

If you are shipping Sysmon logs via Winlogbeat, confirm that your Winlogbeat configuration does NOT use the Elastic Sysmon module. Security Onion will do all the necessary parsing.
7.5.5 More Information

See also:

For more information about sysmon, please see:

Also see “How to Go from Responding to Hunting with Sysinternals Sysmon”:

TrustedSec has a great Community Guide on Sysmon:
https://github.com/trustedsec/SysmonCommunityGuide

7.6 Autoruns

From https://docs.microsoft.com/en-us/sysinternals/downloads/autoruns:

This utility, which has the most comprehensive knowledge of auto-starting locations of any startup monitor, shows you what programs are configured to run during system bootup or login, and when you start various built-in Windows applications like Internet Explorer, Explorer and media players. These programs and drivers include ones in your startup folder, Run, RunOnce, and other Registry keys. Autoruns reports Explorer shell extensions, toolbars, browser helper objects, Winlogon notifications, auto-start services, and much more. Autoruns goes way beyond other autostart utilities.

7.6.1 Integration

Pertinax

Josh Brower developed a great project called Pertinax to normalize autoruns data and integrate it into Security Onion:
https://github.com/defensivedepth/Pertinax/wiki/Introduction

Execute autoruns and ar-normalize.ps1 as shown here:
https://github.com/defensivedepth/Pertinax/wiki/Reference%20Architecture

AutorunsToWinEventLog

Another method for integrating Autoruns into your logging infrastructure is AutorunsToWinEventLog:
https://github.com/palantir/windows-event-forwarding/tree/master/AutorunsToWinEventLog

7.6.2 Downloads

Download Autoruns here:
https://download.sysinternals.com/files/Autoruns.zip
Download ar-normalize.ps1 here:
https://raw.githubusercontent.com/defensivedepth/Pertinax/master/normalize/ar-normalize.ps1

### 7.6.3 More Information

**See also:**

For more information about Autoruns, please see https://docs.microsoft.com/en-us/sysinternals/downloads/autoruns.
Once logs are generated by network sniffing processes or endpoints, where do they go? How are they parsed? How are they stored? That’s what we’ll discuss in this section.

**8.1 Ingest**

Here’s an overview of how logs are ingested in various deployment types.

**8.1.1 Import**

Core Pipeline: Filebeat [IMPORT Node] → ES Ingest [IMPORT Node]
Logs: Zeek, Suricata

**8.1.2 Eval**

Core Pipeline: Filebeat [EVAL Node] → ES Ingest [EVAL Node]
Logs: Zeek, Suricata, Wazuh, Osquery/Fleet

Osquery Shipper Pipeline: Osquery [Endpoint] → Fleet [EVAL Node] → ES Ingest via Core Pipeline
Logs: WEL, Osquery, syslog

**8.1.3 Standalone**

Logs: Zeek, Suricata, Wazuh, Osquery/Fleet, syslog
Logs: WEL, Sysmon

8.1.4 Fleet Standalone

Logs: Osquery

8.1.5 Manager Node

Core Pipeline: Filebeat [Fleet | Forward] → Logstash [Manager] → ES Ingest [S]
Logs: Zeek, Suricata, Wazuh, Osquery/Fleet, syslog

Logs: WEL

8.1.6 Manager + Search

Logs: Zeek, Suricata, Wazuh, Osquery/Fleet, syslog

Logs: Local Wazuh, Osquery/Fleet

Logs: WEL

8.1.7 Heavy

Pipeline: Filebeat [Heavy Node] → Logstash [Heavy] → ES Ingest [Heavy]
Logs: Zeek, Suricata, Wazuh, Osquery/Fleet, syslog

8.1.8 Search

Pipeline: Redis [Search] → Logstash [Search] → ES Ingest [Search]
Logs: Zeek, Suricata, Wazuh, Osquery/Fleet, syslog

8.1.9 Forward

Logs: Zeek, Suricata, Wazuh, syslog
8.2 Filebeat

From https://www.elastic.co/beats/filebeat:

Filebeat helps you keep the simple things simple by offering a lightweight way to forward and centralize logs and files.

In Security Onion 2, Filebeat collects logs from the filesystem. On an Evaluation installation, Filebeat sends those logs directly to Elasticsearch. For other installation types, Filebeat sends to Logstash.

8.2.1 Configuration

You can configure Filebeat inputs and output using Salt. An example of the filebeat pillar can be seen at https://github.com/Security-Onion-Solutions/securityonion/blob/master/salt/filebeat/pillar.example

Any inputs that are added to the pillar definition will be in addition to the default defined inputs. In order to prevent a Zeek log from being used as input, the zeeklogs:enabled pillar will need to be modified. The easiest way to do this is via so-zeek-logs. Alternatively, you can do this manually starting with the default definition at https://github.com/Security-Onion-Solutions/securityonion/blob/master/pillar/zeeklogs.sls. Copy the contents of this file and place it in either the global or minion pillar file depending on if you want the changes to be global or specific to that individual node. If there is a log file that you would like to disable, move that entry from the enabled list to the disabled list. Be sure to follow the proper indentation for YAML.

8.2.2 Diagnostic Logging

Filebeat’s log can be found in /opt/so/log/filebeat/.

8.2.3 More Information

See also:

For more information about Filebeat, please see https://www.elastic.co/beats/filebeat.

8.3 Logstash

From https://www.elastic.co/products/logstash:

Logstash is a free and open server-side data processing pipeline that ingests data from a multitude of sources, transforms it, and then sends it to your favorite “stash.”

In Security Onion 2, Logstash transports unparsed logs to Elasticsearch which then parses and stores those logs.

8.3.1 Configuration

You can configure Logstash using Salt. Here are a few of the settings which you may need to tune in /opt/so/saltstack/local/pillar/minions/$MINION_$ROLE.sls under logstash_settings.
**ls_pipeline_batch_size**

The maximum number of events an individual worker thread will collect from inputs before attempting to execute its filters and outputs. Larger batch sizes are generally more efficient, but come at the cost of increased memory overhead. This is set to 125 by default.

**ls_pipeline_workers**

The number of workers that will, in parallel, execute the filter and output stages of the pipeline. If you find that events are backing up, or that the CPU is not saturated, consider increasing this number to better utilize machine processing power. By default this value is set to the number of cores in the system.

For more information, please see https://www.elastic.co/guide/en/logstash/current/logstash-settings-file.html.

**lsheap**

If total available memory is 8GB or greater, Setup sets the Logstash heap size to 25% of available memory, but no greater than 4GB.

For more information, please see https://www.elastic.co/guide/en/elasticsearch/guide/current/heap-sizing.html# compressed_oops.

You may need to adjust the value depending on your system’s performance. The changes will be applied the next time the minion checks in. You can force it to happen immediately by running `sudo salt-call state.apply logstash` on the actual node or by running `sudo salt $SENSORNAME_$ROLE state.apply logstash` on the manager node.

### 8.3.2 Parsing

Since Logstash no longer parses logs in Security Onion 2, modifying existing parsers or adding new parsers should be done via Elasticsearch.

### 8.3.3 Adding New Logs

If you want to add a new log to the list of logs that are sent to Elasticsearch for parsing, you can update the logstash pipeline configurations by adding to `/opt/so/saltstack/local/salt/logstash/pipelines/config/custom/`.

If you are modifying or adding a new manager pipeline, then add the following to your `global.sls` file:

```yaml
logstash:
  pipelines:
    manager:
      config:
        - so/0009_input_beats.conf
        - so/0010_input_hhbeats.conf
        - so/9999_output_redis.conf.jinja
        - custom/9999_output_custom.jinja
```

If you are modifying or adding a new search pipeline, then add the following to `global.sls`:
8.3.4 Logstash Parsing

If you want to add a legacy Logstash parser (not recommended) then you can copy the file to local. Once the file is in local you can add the proper value to the global.sls as in the example above with - custom/9701_output_custom.jinja.

8.3.5 Forwarding Events to an External Destination

To forward events to an external destination, create a new custom configuration file. Clone the event and match on the output. We recommend using either the http or tcp output plugin. At this time we only support the default bundled Logstash output plugins.

8.3.6 Queue
Memory-backed

From https://www.elastic.co/guide/en/logstash/current/persistent-queues.html:

By default, Logstash uses in-memory bounded queues between pipeline stages (inputs → pipeline workers) to buffer events. The size of these in-memory queues is fixed and not configurable.

Persistent

If you experience adverse effects using the default memory-backed queue, you might consider a disk-based persistent queue. From https://www.elastic.co/guide/en/logstash/current/persistent-queues.html:

In order to protect against data loss during abnormal termination, Logstash has a persistent queue feature which will store the message queue on disk. Persistent queues provide durability of data within Logstash.

Queue Max Bytes

The total capacity of the queue in number of bytes. Make sure the capacity of your disk drive is greater than the value you specify here. If both queue.max_events and queue.max_bytes are specified, Logstash uses whichever criteria is reached first.

Dead Letter Queue

If you want to check for dropped events, you can enable the dead letter queue. This will write all records that are not able to make it into Elasticsearch into a sequentially-numbered file (for each start/restart of Logstash).

This can be achieved by adding the following to the Logstash configuration:

```
dead_letter_queue.enable: true
```

and restarting Logstash:

```
sudo so-logstash-restart
```

The dead letter queue files are located in /nsm/logstash/dead_letter_queue/main/.

More information:

Redis

When using search nodes, Logstash on the manager node outputs to Redis (which also runs on the manager node). Redis queues events from the Logstash output (on the manager node) and the Logstash input on the search node(s) pull(s) from Redis. If you notice new events aren’t making it into Kibana, you may want to first check Logstash on the manager node and then the redis queue.

8.3.7 Log

The Logstash log file is located at /opt/so/log/logstash/logstash.log. Log file settings can be adjusted in /opt/so/conf/logstash/etc/log4j2.properties. Currently, logs are set to rollover daily, and configured to be deleted after 7 days.
8.3.8 Errors

Read-Only

```plaintext
[INFO ][logstash.outputs.elasticsearch] retrying failed action with response code: 403 ("type"="cluster_block_exception", "reason"="blocked by: [FORBIDDEN/12/index.read-only / allow delete (api)];")
```

This error is usually caused by the `cluster.routing.allocation.disk.watermark` (low, high) being exceeded.

You may want to check `/opt/so/log/elasticsearch/<hostname>.log` to see specifically which indices have been marked as read-only.

Additionally, you can run the following command to allow writing to the affected indices:

```bash
curl -XPUT -H 'Content-Type: application/json' localhost:9200/<your_index>/_settings -d '{ "index.blocks.read_only": false }'
```

8.4 Redis

From [https://redis.io/](https://redis.io/):

Redis is an open source (BSD licensed), in-memory data structure store, used as a database, cache and message broker. It supports data structures such as strings, hashes, lists, sets, sorted sets with range queries, bitmaps, hyperloglogs and geospatial indexes with radius queries.

During setup, you can choose to extend your manager node storage using separate search nodes. When you choose this option, Logstash on the manager node outputs to Redis. Search nodes then consume from Redis.

8.4.1 Queue

To see how many logs are in the Redis queue:

```bash
sudo so-redis-count
```

If the queue is backed up and doesn’t seem to be draining, try stopping Logstash on the manager node:

```bash
sudo so-logstash-stop
```

Then monitor the queue to see if it drains:

```bash
watch 'sudo so-redis-count'
```

If the Redis queue looks okay, but you are still having issues with logs getting indexed into Elasticsearch, you will want to check the Logstash statistics on the search node(s).

8.4.2 Tuning

We configure Redis to use 812MB of your total system memory. If you have sufficient RAM available, you may want to increase the `redis_maxmemory` setting in `/opt/so/saltstack/local/pillar/global.sls`. This value is in Megabytes so to set it to use 8 gigs of ram you would set the value to 8192.
Logstash on the manager node is configured to send to Redis. For best performance, you may want to ensure that `batch` is set to `true` and then tune the `batch_events` variable to find the sweet spot for your deployment.

See also:
For more information about logstash’s output plugin for Redis, please see https://www.elastic.co/guide/en/logstash/current/plugins-outputs-redis.html.

Logstash on search nodes pulls from Redis. For best performance, you may want to tune `batch_count` and `threads` to find the sweet spot for your deployment.

See also:
For more information about logstash’s input plugin for Redis, please see https://www.elastic.co/guide/en/logstash/current/plugins-inputs-redis.html.

### 8.4.3 Diagnostic Logging

Redis logs can be found at `/opt/so/log/redis/`.

### 8.4.4 More Information

See also:
For more information about Redis, please see https://redis.io/.

### 8.5 Elasticsearch

From https://www.elastic.co/products/elasticsearch:

Elasticsearch is a distributed, RESTful search and analytics engine capable of solving a growing number of use cases. As the heart of the Elastic Stack, it centrally stores your data so you can discover the expected and uncover the unexpected.

#### 8.5.1 Parsing

In Security Onion 2, Elasticsearch receives unparsed logs from Logstash or Filebeat. Elasticsearch then parses and stores those logs. Parsers are stored in `/opt/so/conf/elasticsearch/ingest/`. Custom ingest parsers can be placed in `/opt/so/saltstack/local/salt/elasticsearch/files/ingest/`. To make these changes take effect, restart Elasticsearch using `so-elasticsearch-restart`.

See also:
For more about Elasticsearch ingest parsing, please see https://www.elastic.co/guide/en/elasticsearch/reference/master/ingest.html.

#### 8.5.2 Community ID

For logs that don’t natively support Community ID, we sponsored the development of an Elasticsearch Ingest Processor to automatically generate Community ID values:

8.5.3 Configuration

Pillar Files

All configuration changes take place in pillar files. You should never need to modify a config file directly. There are two places that hold pillar settings for elasticsearch. The pillars are:

```
/opt/so/saltstack/local/pillar/minions/$minion.sls
```

```
elasticsearch:
  mainip: 10.66.166.22
  mainint: eth0
  esheap: 4066m
  esclusternname: {{ grains.host }}
  node_type: search
  es_port: 9200
  log_size_limit: 3198
  node_route_type: hot
```

```
/opt/so/saltstack/local/pillar/global.sls
```

```
elasticsearch:
  replicas: 0
  true_cluster: False
  true_cluster_name: so
  discovery_nodes: 1
  hot_warm_enabled: False
  cluster_routing_allocation_disk.threshold_enabled: true
  cluster_routing_allocation_disk_watermark_low: 95%
  cluster_routing_allocation_disk_watermark_high: 98%
  cluster_routing_allocation_disk_watermark_flood_stage: 98%
  index_settings:
    so-beats:
      shards: 1
      warm: 7
      close: 30
      delete: 365
    so-firewall:
      shards: 1
      warm: 7
      close: 30
      delete: 365
    so-flow:
      shards: 1
      warm: 7
      close: 30
      delete: 365
    so-ids:
      shards: 1
      warm: 7
      close: 30
      delete: 365
    so-import:
      shards: 1
      warm: 7
      close: 73000
      delete: 73001
    so-osquery:
```
(continues on next page)
Shards

Here are a few tips from https://www.elastic.co/blog/how-many-shards-should-i-have-in-my-elasticsearch-cluster:

TIP: Avoid having very large shards as this can negatively affect the cluster’s ability to recover from failure. There is no fixed limit on how large shards can be, but a shard size of 50GB is often quoted as a limit that has been seen to work for a variety of use-cases.

TIP: Small shards result in small segments, which increases overhead. Aim to keep the average shard size between a few GB and a few tens of GB. For use-cases with time-based data, it is common to see shards between 20GB and 40GB in size.

TIP: The number of shards you can hold on a node will be proportional to the amount of heap you have available, but there is no fixed limit enforced by Elasticsearch. A good rule-of-thumb is to ensure you keep the number of shards per node below 20 to 25 per GB heap it has configured. A node with a 30GB heap should therefore have a maximum of 600-750 shards, but the further below this limit you can keep it the better. This will generally help the cluster stay in good health.

To see your existing shards:

```
curl localhost:9200/_cat/indices
```

The number of shards will be shown in the fifth column.

If you want to view the detail for each of those shards:

```
curl localhost:9200/_cat/shards
```

Given the sizing tips above, if any of your indices are averaging more than 50GB per shard, then you should probably increase the shard count until you get below that recommended maximum of 50GB per shard.
The number of shards for an index is defined in /opt/so/saltstack/local/pillar/global.sls. You can adjust shard counts for each index individually to meet your needs. The next time the node checks in it will apply the settings automatically.

Please keep in mind that old indices will retain previous shard settings and the above settings will only be applied to newly created indices.

**Heap Size**

If total available memory is 8GB or greater, Setup configures the heap size to be 25% of available memory, but no greater than 25GB. You may need to adjust the value for heap size depending on your system’s performance. This can be modified in /opt/so/saltstack/local/pillar/minions/$minion.sls.

For more information, please see:

**Field limit**

Security Onion currently utilizes the default field limit for Elasticsearch indices (1000). If you receive error messages from Logstash, or you would simply like to increase this, you can do so with one of the following options.

**Temporary**

If you only need to increase the field limit temporarily, you can do something like:

```
curl -XPUT -H 'Content-Type: application/json' localhost:9200/logstash-syslog-*/_settings -d '{ "index.mapping.total_fields.limit": 2000 }'
```

The above command would increase the field limit for the logstash-syslog-* indice(s) to 2000. Keep in mind, this setting only applies to the current index, so when the index rolls over and a new one is created, your new settings will not apply.

**Persistent**

If you need this change to be persistent, you can modify the settings stanza for the matched indices in the template:

```
"settings" : {
   "number_of_replicas": 0,
   "number_of_shards": 1,
   "index.refresh_interval" : "5s",
   "index.mapping.total_fields.limit": 2000
},
```

Then restart Logstash:

```
sudo so-logstash-restart
```

Please note that the change to the field limit will not occur immediately – only upon index creation. Therefore, it is recommended to run the previously mentioned temporary command and modify the template file.
Additional options

If you need to make additional directories accessible to Elasticsearch, or would like to specify additional options when starting Elasticsearch, you can do so by adding these items to ELASTICSEARCH_OPTIONS in /etc/nsm/securityonion.conf

8.5.4 Diagnostic Logging

- Elasticsearch logs can be found in /opt/so/log/elasticsearch/.
- Logging configuration can be found in /opt/so/conf/elasticsearch/log4j2.properties.

8.5.5 Distributed

8.5.6 Management

The manager node runs its own local copy of Elasticsearch, which manages cross-cluster search configuration for the deployment. This includes configuration for heavy nodes and search nodes (where applicable), but not forward nodes, as they do not run Elastic Stack components.

8.5.7 Forward Nodes

When using a forward node, Elastic Stack components are not enabled. Filebeat forwards all logs to Logstash on the manager node, where they are stored in Elasticsearch on the manager node or a search node (if the manager node has been configured to use search nodes). From there, the data can be queried through the use of cross-cluster search.

8.5.8 Heavy Nodes

When using a heavy node, Security Onion implements distributed deployments using Elasticsearch’s cross cluster search. When you run Setup and choose Heavy Node, it will create a local Elasticsearch instance and then configure the manager node to query that instance. This is done by updating _cluster/settings on the manager node so that it will query the local Elasticsearch instance.

8.5.9 Search Nodes

Search nodes extend the storage and processing capabilities of the manager node, and run Elasticsearch, Logstash, and Curator. Just like heavy nodes, search nodes are added to the manager node’s cluster search configuration, so the data that resides on the nodes can be queried from the manager node.

8.5.10 Removing a node from the manager node

If you need to remove a node (such as a heavy node or a search node) from your cross cluster search configuration, send the following to Elasticsearch on your manager node (replacing node1 with the actual node you’d like to remove):
PUT _cluster/settings
{
  "persistent": {
    "search": {
      "remote": {
        "node1": {
          "seeds": null
        }
      }
    }
  }
}

You can simply copy/paste the above code (modifying as necessary) into the Console, under “Dev Tools” in Kibana, and click the green triangle. Alternatively, you could submit it to Elasticsearch via a cURL command.

### 8.5.11 Storage

All of the data Elasticsearch collects is stored under `/nsm/elasticsearch/`.

### 8.5.12 Snapshots

Snapshots of the current indices can be taken and stored in a designated repository for archival purposes. Currently, you’ll need to add something like the following to `/opt/so/conf/elasticsearch/elasticsearch.yml`:

```
path.repo: <your file path here>
```

keeping in mind that the above file path is relative to the container’s view of the filesystem.

So, if you decided to add a `path.repo` value of `/backups`, Elasticsearch would be looking for the file path `/backups` inside of the container. To achieve parity with what is present on the host’s filesystem and make that directory accessible to the Elasticsearch Docker container, you’ll want to add something like the following to ELASTICSEARCH_OPTIONS in `/etc/nsm/securityonion.conf`:

```
ELASTICSEARCH_OPTIONS="-v /backups:/backups"
```

(where `/backups` exists on the host file system and is writable by the Elasticsearch user – a directory named `/backups` will be created inside the container, and the container will be able to read/write from that location).

To automate the snapshotting process, you can use Curator, in conjunction with a cron job, much like what is done today with the close and delete jobs.

### 8.5.13 Re-indexing

Re-indexing may need to occur if field data types have changed and conflicts arise. This process can be VERY time-consuming, and we only recommend this if keeping data is absolutely critical. For more information on re-indexing, see:


### 8.6 ElastAlert

From http://elastalert.readthedocs.io/en/latest/elastalert.html#overview:

ElastAlert is a simple framework for alerting on anomalies, spikes, or other patterns of interest from data in Elasticsearch.
At Yelp, we use Elasticsearch, Logstash and Kibana for managing our ever increasing amount of data and logs. Kibana is great for visualizing and querying data, but we quickly realized that it needed a companion tool for alerting on inconsistencies in our data. Out of this need, ElastAlert was created. If you have data being when that data matches certain patterns, ElastAlert is the tool for you.

ElastAlert queries ElasticSearch and provides an alerting mechanism with multiple output types, such as Slack, Email, JIRA, OpsGenie, and many more.

### 8.6.1 Configuration

ElastAlert rules are stored in `/opt/so/conf/elastalert/`.

Security Onion’s default ElastAlert rules are configured with an output type of “debug”, which simply outputs all matches queries to a log file found in `/opt/so/log/elastalert/`.

#### Slack

To have ElastAlert send alerts to something like Slack, we can simply change the alert type and details for a rule like so:

```yaml
alert:
  - "slack":
    slack_webhook_url: "https://hooks.slack.com/services/YOUR_WEBHOOK_URI"
```

#### Email - Internal

To have ElastAlert send to email, we could do something like the following:

```yaml
alert:
  - "email"
  email:
    - "youremail@yourcompany.com"
    smtp_host: "your_company_smtp_server"
    smtp_port: 25
    from_addr: "elastalert@yourcompany.com"
```

#### Email - External

If we need to use an external email provider like Gmail, we can add something like the following:

```yaml
alert:
  - "email"
  email:
    - "youremail@gmail.com"
    smtp_host: "smtp.gmail.com"
    smtp_port: 465
    smtp_ssl: true
    from_addr: "youremail@gmail.com"
    smtp_auth_file: '/etc/elastalert/rules/smtp_auth_file.txt'
```

In the `smtp_auth_file.txt`, add:
MISP

See misp

TheHive

See TheHive

so-elastalert-create

so-elastalert-create is a tool created by Bryant Treacle that can be used to help ease the pain of ensuring correct syntax and creating Elastalert rules from scratch. It will walk you through various questions, and eventually output an Elastalert rule file that you can deploy in your environment to start alerting quickly and easily.

so-elastalert-test

so-elastalert-test is a wrapper script originally written by Bryant Treacle for ElastAlert’s elastalert-test-rule tool. The script allows you to test an ElastAlert rule and get results immediately. Simply run so-elastalert-test, and follow the prompt(s).

Note: so-elastalert-test does not yet include all options available to elastalert-test-rule.

Defaults

With Security Onion’s example rules, Elastalert is configured by default to only count the number of hits for a particular match, and will not return the actual log entry for which an alert was generated.

This is governed by the use of use_count_query: true in each rule file.

If you would like to view the data for the match, you can simply remark this line in the rule file(s). Keep in mind, this may impact performance negatively, so testing the change in a single file at a time may be the best approach.

Timeframe

Keep in mind, for queries that span greater than a minute back in time, you may want to add the following fields to your rule to ensure searching occurs as planned (for example, for 10 minutes):

```plaintext
buffer_time:
  minutes: 10

allow_buffer_time_overlap: true
```

https://elastalert.readthedocs.io/en/latest/ruletypes.html#buffer-time
https://github.com/Yelp/elastalert/issues/805
8.6.2 More Information

See also:
For more information about ElastAlert, please see http://elastalert.readthedocs.io/en/latest/.

8.7 Curator

From https://www.elastic.co/guide/en/elasticsearch/client/curator/current/about.html#about:

Elasticsearch Curator helps you curate, or manage, your Elasticsearch indices and snapshots by:
1. Obtaining the full list of indices (or snapshots) from the cluster, as the actionable list
2. Iterate through a list of user-defined filters to progressively remove indices (or snapshots) from this actionable list as needed.
3. Perform various actions on the items which remain in the actionable list.

8.7.1 Configuration

Curator actions are stored in /opt/so/conf/curator/action/. These actions are run by cron jobs managed by Salt.

Curator defaults to closing indices older than 30 days. To modify this, change cur_close_days in /opt/so/saltstack/local/pillar/minions/$SENSORNAME_$ROLE.sls.

As your disk reaches capacity, Curator starts deleting old indices to prevent your disk from filling up. To change the limit, modify log_size_limit in /opt/so/saltstack/local/pillar/minions/$SENSORNAME_$ROLE.sls.

8.7.2 Logs

When Curator completes an action, it logs its activity in a log file found in /opt/so/log/curator/.

8.7.3 More Information

See also:
For more information about Curator, please see https://www.elastic.co/guide/en/elasticsearch/client/curator/current/about.html#about.

8.8 Data Fields

This page references the various types of data fields utilized by the Elastic Stack in Security Onion. The various fields types are described below.
8.8.1 Fields

Alert Data Fields
Elastalert Fields
Zeek Fields

8.8.2 Template files

Fields are mapped to their proper type using template files found in /opt/so/conf/elasticsearch/templates/.

8.9 Alert Data Fields

Elasticsearch receives NIDS alerts from Suricata via Filebeat or Logstash and parses them using:

/opt/so/conf/elasticsearch/ingest/suricata.alert
/opt/so/conf/elasticsearch/ingest/common_nids
/opt/so/conf/elasticsearch/ingest/common

You can find these online at:

You can find parsed NIDS alerts in Alerts, Hunt, and Kibana via their predefined queries and dashboards or by manually searching for:

event.module:"suricata"
event.dataset:"alert"

Those alerts should have the following fields:

source.ip
source.port
destination.ip
destination.port
network.transport
rule.gid
rule.name
rule.rule
rule.rev
rule.severity
rule.uuid
rule.version
8.10 Elastalert Fields

The following lists field names as they are formatted in Elasticsearch. Elastalert provides its own template to use for mapping into Elastalert, so we do not currently utilize a config file to parse data from Elastalert.

```
index::*:elastalert_status
```

- alert_info.type
- alert_sent
- alert_time
- endtime
- hist
- matches
- match_body.@timestamp
- match_body.num_hits
- match_body.num_matches
- rule_name
- starttime
- time_taken

8.11 Zeek Fields

Zeek logs are sent to Elasticsearch where they are parsed using ingest parsing. Most Zeek logs have a few standard fields and they are parsed as follows:

```
ts => @timestamp
uid => log.id.uid
id.orig_h => source.ip
id.orig_p => source.port
id.resp_h => destination.ip
id.resp_p => destination.port
```

The remaining fields in each log are specific to the log type. To see how the fields are mapped for a specific Zeek log, take a look at its ingest parser.

You can find ingest parsers in your local filesystem at `/opt/so/conf/elasticsearch/ingest/` or you can find them online at:


For example, suppose you want to know how the Zeek conn.log is parsed. You could take a look at `/opt/so/conf/elasticsearch/ingest/zeek.conn` or view it online at:


You’ll see that `zeek.conn` then calls the `zeek.common` pipeline (`/opt/so/conf/elasticsearch/ingest/zeek.common`):

which in turn calls the common pipeline (/opt/so/conf/elasticsearch/ingest/common):

### 8.12 Community ID

From https://github.com/corelight/community-id-spec:

> When processing flow data from a variety of monitoring applications (such as Zeek and Suricata), it’s often desirable to pivot quickly from one dataset to another. While the required flow tuple information is usually present in the datasets, the details of such “joins” can be tedious, particular in corner cases. This spec describes “Community ID” flow hashing, standardizing the production of a string identifier representing a given network flow, to reduce the pivot to a simple string comparison.

Security Onion enables the native Community ID support in both Zeek and Suricata.

We sponsored the development of Community ID support in osquery:
https://dactiv.llc/blog/correlate-osquery-network-connections/

For tools that don’t natively support Community ID, we sponsored the development of an Elasticsearch Ingest Processor to automatically generate Community ID values:

#### 8.12.1 More Information

See also:

For more information about Community ID, please see https://github.com/corelight/community-id-spec.

### 8.13 Re-Indexing

When changing mappings or index settings, we may need to re-index the existing indices to ensure there are no mapping conflicts.

One way to do this by using the following experimental example script:
https://raw.githubusercontent.com/weslambert/securityonion-elastic-misc/master/so-elastic-reindex

Pull down the script to your Security Onion box:

```
wget https://raw.githubusercontent.com/weslambert/securityonion-elastic-misc/master/so-elastic-reindex
```

Make the script executable:

```
sudo chmod +x so-elastic-reindex
```

Re-index all indices matching logstash-*, pulling the appropriate refresh_interval from the template named logstash in Elasticsearch:

```
sudo ./so-elastic-reindex -i "logstash-*" -t "logstash"
```

The script should then progress to re-index the matching indices, and inform you when it has completed.
Warning: Abnormal execution of this script may result in data loss—there are NO GUARANTEES this process will work perfectly for you.
In this section, we’ll review how to keep Security Onion up-to-date.

### 9.1 soup

As we make updates to Security Onion, we publish blog posts to [https://blog.securityonion.net](https://blog.securityonion.net). You’ll want to review these blog posts for any relevant information about the individual updates.

Once you’re ready to install an update, use the `soup` command:

```
sudo soup
```

If necessary, `soup` will update itself and then ask you to run `soup` again. Once `soup` is fully updated, it will then update `Salt` and the `Docker` images.

**Warning:** Please note that `soup` only updates Security Onion components and does NOT update the underlying operating system (OS). There is an option during `Configuration` to automatically update the OS packages.

#### 9.1.1 Agents

If you’ve previously added any external agents (*Wazuh, Beats*, etc.), be sure to upgrade them to match the version of your upgraded components.

#### 9.1.2 Distributed deployments

If you have a distributed deployment with a manager node and separate sensor nodes and/or search nodes, you **only** need to run `soup` on the manager. Once `soup` has completed, other nodes should update themselves at the next `Salt` highstate (typically within 15 minutes).
9.1.3 Airgap

If you have an airgap deployment, please see the Airgap section for further information.

9.2 End Of Life

This page lists End Of Life (EOL) dates for older versions of Security Onion and older components.


Security Onion 14.04 reached EOL on November 30, 2018:
https://blog.securityonion.net/2018/06/6-month-eol-notice-for-security-onion.html

ELSA reached EOL on October 9, 2018:
https://blog.securityonion.net/2018/04/6-month-eol-notice-for-elsa.html

Xplico reached EOL on June 5, 2018:
https://blog.securityonion.net/2017/12/6-month-eol-notice-for-security-onion.html
In Security Onion, there are two main types of accounts:

- operating system (OS) accounts
- application accounts used when authenticating to Security Onion Console (SOC)

## 10.1 Passwords

### 10.1.1 OS user account

When you first install Security Onion, you create a standard OS user account for yourself. If you need to change your OS user password, you can use the `passwd` command:

```bash
passwd
```

### 10.1.2 OS root account

Your default user account should have sudo permissions. Command-line utilities that require administrative access can be prefixed with `sudo`. For example, the `so-status` command requires administrative access so you can run it with `sudo` as follows:

```bash
sudo so-status
```

### 10.1.3 Security Onion Console (SOC)

Log into Security Onion Console (SOC) using the username and password you created in the Setup wizard.

You can change your password in Security Onion Console (SOC) by clicking the user icon in the upper right corner and then clicking Settings:
If you’ve forgotten your SOC password, you can reset it using the `so-user` command:

```
so-user update username@example.com
```

## 10.1.4 TheHive

Log into *TheHive* using the username and password you created in the Setup wizard.

You can change your password in *TheHive* by clicking the user icon in the upper right corner, clicking *Settings*. Then click *Update password* and follow the prompts.

## 10.2 Adding Accounts

### 10.2.1 OS

If you need to add a new OS user account, you can use the `adduser` command. For example, to add a new account called `tom`:
sudo adduser tom

For more information, please see man adduser.

10.2.2 SOC & TheHive & Fleet - CLI

If you need to add a new account to Security Onion Console (SOC), TheHive, and Fleet you can use the so-user-add command and specify the user's email address. For example, to add a new account for tom@example.com:

sudo so-user-add tom@example.com

10.2.3 TheHive - UI

If you need to add a new TheHive account, log into TheHive with your existing account and then click Admin and Users to access the User management screen. Then click the Add user button and follow the prompts. Once the user has been created, you can then set their password.

10.3 Listing Accounts

10.3.1 OS

Operating System (OS) user accounts are stored in /etc/passwd. You can get a list of all OS accounts using the following command:

cut -d: -f1 /etc/passwd

If you want a list of user accounts (not service accounts), then you can filter /etc/passwd for accounts with a UID greater than 1000 like this:

cat /etc/passwd | awk -F: '$3 > 1000 {print ;}' | cut -d: -f1

10.3.2 SOC

To list all Security Onion Console (SOC) accounts, you can use the so-user command with the list option:

sudo so-user list

Alternatively, you can get a list of users in Security Onion Console (SOC) by clicking Administration and then Users:
10.3.3 TheHive

To see all TheHive accounts, log into TheHive and then click Admin and Users to access the User management screen.

10.4 Disabling Accounts

10.4.1 OS

If you need to disable an OS user account, you can expire the account using `usermod --expiredate 1`. For example, to disable the account for user `tom`:

```
sudo usermod --expiredate 1 tom
```

For more information, please see `man passwd` and `man usermod`.

10.4.2 SOC & TheHive & Fleet - CLI

If you need to disable an account in Security Onion Console (SOC), TheHive, and Fleet, you can use the `so-user-disable` command and specify the user’s email address. For example, to disable the account for `tom@example.com`:

```
sudo so-user-disable tom@example.com
```
10.4.3 TheHive - UI

Log into TheHive and then click Admin and Users to access the User management screen. Then click the Lock button for the user account you want to disable.
You can control individual services with the so-<noun>-<verb> scripts. You can see a list of all of these scripts with the following command:

```
ls /usr/sbin/so-***
```

The following examples are for Zeek, but you could substitute whatever service you’re trying to control (Logstash, Elasticsearch, etc.).

Start Zeek:
```
sudo so-zeek-start
```

Stop Zeek:
```
sudo so-zeek-stop
```

Restart Zeek:
```
sudo so-zeek-restart
```
This section covers how to customize Security Onion for your environment.

12.1 Cortex

From https://github.com/TheHive-Project/Cortex:

Cortex tries to solve a common problem frequently encountered by SOCs, CSIRTs and security researchers in the course of threat intelligence, digital forensics and incident response: how to analyze observables they have collected, at scale, by querying a single tool instead of several?

Cortex, an open source and free software, has been created by TheHive Project for this very purpose. Observables, such as IP and email addresses, URLs, domain names, files or hashes, can be analyzed one by one or in bulk mode using a Web interface. Analysts can also automate these operations thanks to the Cortex REST API.

12.1.1 Usage

Log into the Cortex web interface at /cortex (at the IP address or hostname of your Security Onion installation) using the same credentials that you use to log into TheHive.
In Security Onion, Cortex is set up with two default organizations:

- **cortex** - This is a default organization that is created by Cortex for overall management.
- **SecurityOnion** - This is an organization that we create to enable analyzers by default and provide integration with *TheHive*.

Users initially authenticate to Cortex via the username and password supplied during setup. Once authenticated to the Cortex organization, users will possess *superadmin* privileges, capable of managing all organizations, users, etc.

From here, users should create an additional user for the *SecurityOnion* organization, or create their own organization/users if they wish to log into Cortex and manage analyzers and responders.

It is always recommended that you create your own organization, but the provided organizations should work for testing.

### 12.1.2 More Information

**See also:**
For more information about Cortex, please see [https://github.com/TheHive-Project/Cortex](https://github.com/TheHive-Project/Cortex).

### 12.2 Proxy Configuration

If you need to force your Internet traffic through a proxy server, you can put your proxy server settings in `/etc/environment` like this:

```bash
export http_proxy=http://server:port
export https_proxy=https://server:port
export ftp_proxy=https://server:port
export no_proxy="localhost,127.0.0.1"
```

You may also need to include the actual IP address of the Security Onion box itself in the `no_proxy` variable above.

#### 12.2.1 Docker

To configure Docker proxy settings, please see [https://docs.docker.com/network/proxy/](https://docs.docker.com/network/proxy/).

#### 12.2.2 sudo

If you’re going to run something using `sudo`, remember to use the `-i` option to force it to process the environment variables. For example:

```bash
sudo -i rule-update
```

### 12.3 Firewall

#### 12.3.1 Node Communication

When configuring network-based firewalls, you’ll want to take into consideration the communication/ports required for nodes to talk to one another.
All nodes to manager:

- 3142 (Apt-cacher-ng) (if manager proxy enabled)
- 5000 (Docker registry)
- 8086 (influxdb)
- 4505 (Salt)
- 4506 (Salt)
- 5644 (Filebeat)
- 443 (Sensoroni)
- 8080 (Osquery, if enabled)

Search nodes from/to manager:

- 9300 (Node-to-node for Elasticsearch)
- 9696 (Redis)

### 12.3.2 Function

The firewall state is designed to function with the idea of creating port groups and host groups, each with their own alias or name, and associating the two in order to create an allow rule. A node that has a port group and host group association assigned to it will allow those hosts to connect to those ports on that node. The default allow rules for each node are defined by its role (manager, searchnode, sensor, heavynode, etc) in the grid. Host groups and port groups can be created or modified from the manager node using either `so-allow`, `so-firewall` or manually editing the yaml files. When setup is run on a new node, it will SSH to the manager, using the soremote account, and add itself to the appropriate host groups. All node types are added to the minion host group to allow Salt communication. If you were to add a search node, you would see its IP appear in both the `minion` and the `search_node` host groups.

**Port Groups** Port groups are a way of grouping together ports similar to a firewall port/service alias. For example if you had a web server you could include 80 and 443 tcp into an alias or in this case a port group.

**Host Groups** Host groups is similar to port groups but for storing lists of hosts that will be allowed to connect to the associated port groups.

There are two directories that contain the yaml files for the firewall configuration.

```
/opt/so/saltstack/default/firewall
```

This is where the default firewall rules are located. The files in this directory should not be modified as they could possibly be overwritten during a soup update in the event we update those files.

```
/opt/so/saltstack/default/salt/firewall/portgroups.yaml
```

This is where the default port groups are defined.
This is where the default hostgroups are defined. There isn’t much in here other than anywhere, dockernet, localhost and self.

This is where the default allow rules come together and pair hostgroups and portgroups and assign that pairing to a node based on its role in the grid. In the image below, we can see how we define some rules for an eval node.
This is the directory where the firewall rules specific to your grid are located.

This is where custom port groups are defined.

This is where many default named hostgroups get populated with IPs that are specific to your environment. When you run `so-allow` or `so-firewall`, it modifies this file to include the IP provided in the proper hostgroup. Some node types get their IP assigned to multiple host groups

This is where host group and port group associations would be made to create custom host group and port group assignments that would apply to all nodes of a certain role type in the grid.

### 12.3.3 Managing

Managing firewall rules, for all devices, should be done from the manager node using either `so-allow`, `so-firewall` or, for advanced cases, manually editing the yaml files.
12.3.4 Examples

Allow hosts to send syslog to a sensor node.

By default, if you use `so-allow` to add a host to the syslog hostgroup, that host will only be allowed to connect to the manager node. If we want to allow a host or group of hosts to send syslog to a sensor, then we can do the following:

1. Create a new host group that will contain the IPs of the hosts that you want to allow to connect to the sensor. This will add the host group to `/opt/so/saltstack/local/salt/firewall/hostgroups.local.yaml`. If the host group already exists, you can skip to step 2. From the manager run:

   ```
sudo so-firewall addhostgroup <GROUP_NAME>
   
```

2. Add the desired IPs to the host group. This will add the IPs to the host group in `/opt/so/saltstack/local/salt/firewall/hostgroups.local.yaml`.

   ```
sudo so-firewall includehost <GROUP_NAME> <IP>
   
```

3. Since we reused the syslog port group that is already defined, we don’t need to create a new port group. Now we have to build the association between the host group and the syslog port group and assign that to our sensor node. Add the following to the sensor minion pillar file located at `/opt/so/saltstack/local/pillar/minions/<HOSTNAME>_<ROLE>.sls`

   ```
firewall:
  assigned_hostgroups:
    chain:
      DOCKER-USER:
        hostgroups:
          syslogtosensor1:
            portgroups:
              - portgroups.syslog
   
```

4. Now that the configuration is in place, you can either wait for the sensor to sync with Salt running on the manager, or you can force it to update its firewall by running the following from the manager:

   ```
salt <HOSTNAME>_<ROLE> state.apply firewall
   ```

**Warning:** Please review the *Salt* section to understand pillars and templates. Modifying these values outside of `so-allow` or `so-firewall` could lead to problems accessing your existing hosts. This is an advanced case and you most likely won’t never need to modify these files.

12.4 Email Configuration

Some applications rely on having a mail server in the OS itself and other applications (like Wazuh) have their own mail configuration and so they don’t rely on a mail server in the OS itself.

12.4.1 Operating System

You can install and configure your favorite mail server. Depending on your needs, this could be something simple like `nullmailer` or something more complex like `exim4`. 
12.4.2 Wazuh

If you want Wazuh to send email, you can modify `/opt/so/conf/wazuh/ossec.conf` as follows:

```xml
<global>
  <email_notification>yes</email_notification>
  <email_to>YourUsername@YourDomain.com</email_to>
  <smtp_server>YourMailRelay.YourDomain.com</smtp_server>
  <email_from>ossec@YourDomain.com</email_from>
  <email_maxperhour>100</email_maxperhour>
</global>
```

Then restart Wazuh:

```
sudo so-wazuh-restart
```

You can specify the severity of an event for which Wazuh will send email alerts by specifying an appropriate value for `email_alert_level` in `/opt/so/conf/wazuh/ossec.conf`. If you notice `email_alert_level` is not being respected for a certain rule, it may be that the option is overridden by `<options>alert_by_email</options>` being set for a rule. You can modify this behavior in `/opt/so/conf/wazuh/rules/local_rules.xml`.


12.4.3 Zeek

Edit `/opt/so/conf/zeek/zeekctl.cfg` and set the following:

```plaintext
MailTo = YourUsername@YourDomain.com
sendmail = /usr/sbin/sendmail
```

Then update and restart Zeek:

```
sudo so-zeek-restart
```

You should then start receiving hourly connection summary emails. If you don’t want the connection summary emails, you can add the following to `zeekctl.cfg` and update and restart Zeek as shown above:

```plaintext
tracesummary=
```

You may want to receive emails for Zeek notices. To do that, add the following to `/opt/so/conf/zeek/local.zeek` and then update/restart Zeek as shown above:

```plaintext
hook Notice::policy(n: Notice::Info) {
  add n$actions[Notice::ACTION\_ALARM];
}
```

Also see [http://mailman.icsi.berkeley.edu/pipermail/bro/2013-December/006418.html](http://mailman.icsi.berkeley.edu/pipermail/bro/2013-December/006418.html).

12.4.4 Elastalert

Follow the steps on the [Elastalert page](https://www.elastic.co/elastalert).
12.4.5 Lack of network traffic

If you configured Zeek as shown above, they should automatically email you if your network sensors stop seeing traffic.

12.5 NTP

Depending on how you installed, the underlying operating system may be configured to pull time updates from the NTP Pool Project and perhaps others as a fallback. You may want to change this default NTP config to use your preferred NTP provider. If you’re using our Security Onion ISO image, this can be set in `/etc/chrony.conf`.

12.5.1 IDS Alerts

Anybody can join the NTP Pool Project and provide NTP service. Occasionally, somebody provides NTP service from a residential DHCP address that at some point in time may have also been used for Tor. This results in IDS alerts for Tor nodes where the port is 123 (NTP). This is another good reason to modify the NTP configuration to pull time updates from your preferred NTP provider.

12.6 SSH

Security Onion uses the latest SSH packages. It does not modify the default SSH configuration in `/etc/ssh/sshd_config` or manage it in any way with Salt. This allows you to add any PAM modules or enable two factor authentication (2FA) of your choosing.

12.6.1 Distributed Deployments

For distributed deployments, nodes only connect to the manager via SSH when they initially join the grid. That initial connection is done using the `soremote` account. If you enable 2FA for SSH, you will need to disable 2FA for the `soremote` account. The `soremote` account can be disabled when you are not adding any nodes to the grid.

12.6.2 Hardening

Some organizations require the removal of certain ciphers and algorithms from sshd. Starting in Security Onion 2.3.10, we have a tool that will do this for you. Simply run `so-ssh-harden` and then reload the sshd config and verify that you can connect via ssh properly. Alternatively, you can manually modify your `sshd_config` as follows:

```
sshd -T | grep "^ciphers" | sed -e "s/\(3des-cbc\|aes128-cbc\|aes192-cbc\|aes256-cbc\|arcfour\|arcfour128\|arcfour256\|blowfish-cbc\|cast128-cbc\|rijndael-cbc\|rijndael-lysator.liu.se\)/\n\)/g" >> /etc/ssh/sshd_config
sshd -T | grep "^kexalgorithms" | sed -e "s/\(diffie-hellman-group14-shal\|ecdh-sha2-nistp256\|diffie-hellman-group-exchange-sha256\)/\n\)/g" >> /etc/ssh/sshd_config
sshd -T | grep "^macs" | sed -e "s/\(hmac-sha2-512\|umac-64@openssh.com\|hmac-sha2-256\|umac-64@openssh.com\)/\n\)/g" >> /etc/ssh/sshd_config
sshd -T | grep "^hostkeyalgorithms" | sed "s|ecdsa-sha2-nistp256,||g" | sed "s|ssh-rsa,||g" >> /etc/ssh/sshd_config
```
To get the best performance out of Security Onion, you’ll want to tune it for your environment. Start by creating Berkeley Packet Filters (BPFs) to ignore any traffic that you don’t want your network sensors to process. Then tune your IDS rulesets. There may be entire categories of rules that you want to disable first and then look at the remaining enabled rules to see if there are individual rules that can be disabled. Once your rules and alerts are under control, then check to see if you have packet loss. If so, then tune the number of AF-PACKET workers for sniffing processes. If you are on a large network, you may need to do additional tuning like pinning processes to CPU cores. More information on each of these topics can be found in this section.

13.1 Salt

From https://docs.saltstack.com/en/latest:

Salt is a new approach to infrastructure management built on a dynamic communication bus. Salt can be used for data-driven orchestration, remote execution for any infrastructure, configuration management for any app stack, and much more.

Note: Salt is a core component of Security Onion 2 as it manages all processes on all nodes. In a distributed deployment, the manager node controls all other nodes via salt. These non-manager nodes are referred to as salt minions.

13.1.1 Firewall Requirements

Minions must be able to connect to the manager node on ports 4505/tcp and 4506/tcp:

http://docs.saltstack.com/topics/tutorials/firewall.html
13.1.2 Checking Status

You can use salt’s `test.ping` to verify that all your nodes are up:

```
sudo salt \* test.ping
```

13.1.3 Remote Execution

Similarly, you can use salt’s `cmd.run` to execute a command on all your nodes at once. For example, to check disk space on all nodes:

```
sudo salt \* cmd.run 'df'
```

13.1.4 Configuration

Many of the options that are configurable in Security Onion 2 are done via pillar assignments in either the global or minion pillar files. Pillars are a Saltstack concept, formatted typically in YAML, that can be used to parameterize states via templating. Saltstack states are used to ensure the state of objects on a minion. In many of the use cases below, we are providing the ability to modify a configuration file by editing either the global or minion pillar file.

Global pillar file: This is the pillar file that can be used to make global pillar assignments to the nodes. It is located at `/opt/so/saltstack/local/pillar/global.sls`.

Minion pillar file: This is the minion specific pillar file that contains pillar definitions for that node. Any definitions made here will override anything defined in other pillar files, including global. This is located at `/opt/so/saltstack/local/pillar/minions/<minionid>.sls`.

Default pillar file: This is the pillar file located under `/opt/so/saltstack/default/pillar/`. Files here should not be modified as changes would be lost during a code update.

Local pillar file: This is the pillar file under `/opt/so/saltstack/local/pillar/`. These are the files that will need to be changed in order to customize nodes.

Here are some of the items that can be customized with pillar settings:

- Filebeat
- Firewall
- Managing Alerts
- Suricata
- Zeek

13.1.5 More Information

See also:

For more information about Salt, please see https://docs.saltstack.com/en/latest/.

13.2 Homenet

Currently homenet is only used for Suricata, but could be used for other tools in the future.
13.2.1 Configuration

A node can be assigned either the global homenet or its own homenet.

By default, a node will use the global homenet pillar value if it is defined in the global pillar file (/opt/so/saltstack/local/pillar/global.sls) under `global:hnmanager`.

```
global:
    soversion: '2.3.0'
    hnmanager: '10.0.0.0/8,192.168.0.0/16,172.16.0.0/12'
```

In order to define a per node homenet, it can be defined in the minion pillar file (/opt/so/saltstack/local/pillar/minions/$SENSORNAME_$ROLE.sls) under `sensor:hnsensor`.

```
sensor:
    interface: 'bond0'
    mainip: '172.16.106.112'
    mainint: 'eth0'
    zeek_lbps: 5
    suriprocs: 2
    manager: 'somanager1'
    mtu: 1500
    uniqueid: 1602623674
    hnsensor: 10.0.0.0/8
```

In order to sync the configuration change with the node, we can either wait for the node to automatically highstate on the predefined interval, or we can force for it. Since this homenet only applies to Suricata, we can apply the suricata state to the node.

- From the manager:
  ```
salt $SENSORNAME_$ROLE state.apply suricata
  ```
  or

- From the node:
  ```
salt-call state.apply suricata
  ```

13.2.2 More Information

See also:

For more information about Suricata, such as defining other address groups or ports groups please see [Suricata](#).

13.3 BPF


> BPF supports filtering packets, allowing a userspace process to supply a filter program that specifies which packets it wants to receive. For example, a tcpdump process may want to receive only packets that initiate a TCP connection. BPF returns only packets that pass the filter that the process supplies. This avoids copying unwanted packets from the operating system kernel to the process, greatly improving performance.
13.3.1 Configuration

Global BPF

You can specify your BPF in the global pillar on your manager node (/opt/so/saltstack/local/pillar/global.sls), and by default, it will apply to all interfaces in your entire deployment. If there is no BPF configuration already in the file, you can append it to the bottom of the file.

If you have separate sensors reporting to that manager node, they will pull down the relevant BPF as part of the Salt update that runs every 15 minutes and then restart Suricata/Steno/Zeek so that the BPF change will take effect.

Node-Specific BPF

If you don’t want your sensors to inherit BPF from the manager node, you can edit the minion sls file (/opt/so/saltstack/local/pillar/minions/$Hostname.sls), which will override any global BPF settings set from the global pillar.

Simple Example

Suppose you want Stenographer to not record full packet capture for port 443:

```
steno:
  bpf:
    - not port 443
```

Quoting

YAML rules apply and so if you want to use a reserved YAML character such as `[]` `{}>` `*` `!` `%` `#` `@` `,` `,` , then you may need to enclose the entire line in double quotes. For example:

```
steno:
  bpf:
    - "!(port 443)"
```

Multiple Conditions

If your BPF contains multiple conditions you can put them on multiple lines and join them with `&&` but make sure the final condition has no `&&` at the end. For example:

```
nids:
  bpf:
    - not host 192.168.1.2 &&
    - not host 192.168.1.3 &&
    - not host 192.168.1.4
```

VLAN

If you have traffic that has VLAN tags, you can craft a BPF as follows:

```
<your filter> or (vlan and <your filter>)
```
Notice that you must include your filter on both sides of the vlan tag.

For example:

\[
\text{(not (host 192.168.1.2 or host 192.168.1.3 or host 192.168.1.4)) or (vlan and (not,)}
\]

\[
\text{(host 192.168.1.2 or host 192.168.1.3 or host 192.168.1.4)))}
\]

Warning: Please note that Zeek and Stenographer currently do not see VLAN tags due to the way that AF-PACKET works:

https://github.com/google/stenographer/issues/211

### Troubleshooting BPF using tcpdump

If you need to troubleshoot BPF, you can use tcpdump as shown in the following articles.

http://taosecurity.blogspot.com/2004/09/understanding-tcpdumps-d-option-have.html
http://taosecurity.blogspot.com/2008/12/bpf-for-ip-or-vlan-traffic.html

#### 13.3.2 More Information

See also:

For more information about BPF, please see:

http://biot.com/capstats/bpf.html

### 13.4 Managing Rules

#### 13.4.1 Updating Rules

To update your rules, run `so-rule-update` on your manager node:

```
sudo so-rule-update
```

If you have a distributed deployment and you update the rules on your manager node, then those rules will automatically replicate from the manager node to your sensors within 15 minutes. If you don’t want to wait 15 minutes, you can force the sensors to update immediately by running the following command on your manager node:

```
sudo salt \\* state.highstate
```

#### 13.4.2 Rulesets

Security Onion offers the following choices for rulesets to be used by Suricata.
13.4.3 ET Open

- optimized for Suricata, but available for Snort as well
- free

For more information, see:
https://rules.emergingthreats.net/open/

13.4.4 ET Pro (Proofpoint)

- optimized for Suricata, but available for Snort as well
- rules retrievable as released
- license fee per sensor (users are responsible for purchasing enough licenses for their entire deployment)

To enable ET Pro in an already installed grid modify the /opt/so/saltstack/local/pillar/minions/<manager.sls>

```yaml
idstools:
  config:
    ruleset: 'ETPRO'
    oinkcode: 'MYOINKCODE'
```

For more information, see:

13.4.5 Snort Community

- optimized for Snort
- community-contributed rules
- free

For more information, see:
https://www.snort.org/downloads/#rule-downloads
https://www.snort.org/faq/what-are-community-rules

13.4.6 Snort Registered

- optimized for Snort
- Snort SO (Shared Object) rules will only work with Snort
- same rules as Snort Subscriber ruleset, except rules only retrievable after 30 days past release
- free

For more information, see:
13.4.7 Snort Subscriber (Talos)

- optimized for Snort
- Snort SO (Shared Object) rules will only work with Snort
- rules retrievable as released
- license fee per sensor (users are responsible for purchasing enough licenses for their entire deployment)

For more information, see:
https://www.snort.org/downloads/#rule-downloads
https://snort.org/documents/registered-vs-subscriber

13.5 Adding Local Rules

13.5.1 NIDS

You can add NIDS rules in /opt/so/saltstack/local/salt/idstools/local.rules on your manager. Within 15 minutes, Salt should then copy those rules into /opt/so/rules/nids/local.rules. The next run of idstools should then merge /opt/so/rules/nids/local.rules into /opt/so/rules/nids/all.rules which is what Suricata reads from.

If you don’t want to wait for these automatic processes, you can run them manually from the manager (replacing $SENSORNAME_$ROLE as necessary):

```
sudo salt-call state.highstate
sudo so-rule-update
sudo salt $SENSORNAME_$ROLE state.apply suricata
```

For example:

- Let’s add a simple rule to /opt/so/saltstack/local/salt/idstools/local.rules that’s really just a copy of the traditional id check returned root rule:

```
alert ip any any -> any any (msg:"GPL ATTACK_RESPONSE id check returned root 2"; content:"uid=0|28|root|29|"; classtype:bad-unknown; sid:7000000; rev:1;)
```

- From the manager, tell Salt to update:

```
sudo salt-call state.highstate
```

- Update rules:

```
sudo so-rule-update
```

- Restart Suricata (replacing $SENSORNAME_$ROLE as necessary):

```
sudo salt $SENSORNAME_$ROLE state.apply suricata
```
• If you built the rule correctly, then Suricata should be back up and running.
• You can then run `curl http://testmynids.org/uid/index.html` on the node to generate traffic which should cause this rule to alert (and the original rule that it was copied from, if it is enabled).

### 13.5.2 YARA

**Local Rules:**

To add local YARA rules, create a directory in `/opt/so/saltstack/local/salt/strelka/rules`, for example `localrules`. Inside of `/opt/so/saltstack/local/salt/strelka/rules/localrules`, add your YARA rules.

After adding your rules, update the configuration by running `so-strelka-restart`.

**Remotely Managed Rules:**

To have `so-yara-update` pull YARA rules from a Github repo, copy `/opt/so/saltstack/local/salt/strelka/rules/`, and modify `repos.txt` to include the repo URL (one per line).

Next, run `so-yara-update` to pull down the rules. Finally, run `so-strelka-restart` to allow Strelka to pull in the new rules.

### 13.6 Managing Alerts

Security Onion generates a lot of valuable information for you the second you plug it into a TAP or SPAN port. Between Zeek logs, alert data from Suricata, and full packet capture from Stenographer, you have enough information to begin identifying areas of interest and making positive changes to your security stance.

**Note:** Network Security Monitoring, as a practice, is not a solution you can plug into your network, make sure you see blinking lights and tell people you are “secure.” It requires active intervention from an analyst to qualify the quantity of information presented. One of those regular interventions is to ensure that you are tuning properly and proactively attempting to reach an acceptable level of signal to noise.

### 13.6.1 Alerting Engines & Severity

There are three alerting engines within Security Onion: *Suricata, Wazuh* and *Playbook* (Sigma). Though each engine uses its own severity level system, Security Onion converts that to a standardized alert severity:

- `event.severity: 4` ➞ `event.severity_label: critical`
- `event.severity: 3` ➞ `event.severity_label: high`
- `event.severity: 2` ➞ `event.severity_label: medium`
- `event.severity: 1` ➞ `event.severity_label: low`

All alerts are viewable in *Alerts*, *Hunt*, and *Kibana*. 
13.6.2 NIDS Testing

The easiest way to test that our NIDS is working as expected might be to simply access http://testmynids.org/uid/index.html from a machine that is being monitored by Security Onion. You can do so via the command line using curl:

```
curl testmynids.org/uid/index.html
```

Alternatively, you could also test for additional hits with a utility called tmNIDS, running the tool in interactive mode:

```
curl -sSL https://raw.githubusercontent.com/0xtf/testmynids.org/master/tmNIDS -o /tmp/tmNIDS && chmod +x /tmp/tmNIDS && /tmp/tmNIDS
```

If everything is working correctly, you should see a corresponding alert (GPL ATTACK_RESPONSE id check returned root) in Alerts, Kibana, or Hunt. If you do not see this alert, try checking to see if the rule is enabled in /opt/so/rules/nids/all.rules:

```
grep 2100498 /opt/so/rules/nids/all.rules
```

13.6.3 Identifying rule categories

Both the Snort Subscriber (Talos) and the Emerging Threats rulesets come with a large number of rules enabled (over 20,000 by default). You should only run the rules necessary for your environment. So you may want to disable entire categories of rules that don’t apply to you. Run the following command to get a listing of categories and the number of rules in each:

```
cut -d" -f2 /opt/so/rules/nids/all.rules | grep -v "^$" | grep -v "^#" | awk '{print $1, $2}' | sort | uniq -c | sort -nr
```

Also see:

13.6.4 So what’s next?

Firstly, in tuning your sensor, you must understand whether or not taking corrective actions on this signature will lower your overall security stance. For some alerts, your understanding of your own network and the business being transacted across it will be the deciding factor. For example, if you don’t care that users are accessing Facebook, then you can silence the policy-based signatures for Facebook access.

Another consideration is whether or not the traffic is being generated by a misconfigured piece of equipment. If it is, then the most expedient measure may be to resolve the misconfiguration and then reinvestigate tuning.

There are multiple ways to handle overly productive signatures and we’ll try to cover as many as we can without producing a full novel on the subject.

13.6.5 Disable the sid

Security Onion uses idstools to download new signatures every night and process them against a set list of user generated configurations.

To enable or disable SIDs for Suricata, the Salt idstools pillar can be used in the minion pillar file (/opt/so/saltstack/local/pillar/minions/<minionid>.sls). In a distributed Security Onion environment,
you only need to change the configuration in the manager pillar and then all other nodes will get the updated rules automatically.

If SID 4321 is noisy, you can disable it as follows:

```yaml
idstools:
  sids:
    disabled:
    - 4321
```

Then, from the manager run `sudo salt $SENSORNAME_$ROLE state.apply idstools` to update the config.

If you want to disable multiple rules at one time, you can use a regular expression, but make sure you enclose the full entry in single quotes like this:

```yaml
idstools:
  sids:
    disabled:
    - 're:heartbleed'
```

### 13.6.6 Modify the SID

 `/opt/so/saltstack/local/pillar/minions/<minionid>.sls` contains a modify sub-section under the `idstools` section. You can list modifications here and then update the config:

- From the manager, run:
  ```bash
  salt $SENSORNAME_$ROLE state.apply idstools
  ```

### 13.6.7 Rewrite the signature

In some cases, you may not want to use the modify option above, but instead create a copy of the rule and disable the original. In Security Onion, locally created rules are stored in `/opt/so/rules/nids/local.rules`.

- Edit the `/opt/so/rules/nids/local.rules` file using `vi` or your favorite text editor:
  ```bash
  sudo vi /opt/so/rules/nids/local.rules
  ```
- Paste the rule. You may want to bump the SID into the 90,000,000 range and set the revision to 1.
- Now that we have a signature that will generate alerts a little more selectively, we need to disable the original signature. As shown above, we edit the minion pillar and add the SID to the `idstools - sids - disabled` section.
- Finally, from the manager, update the config on the remote node:
  ```bash
  salt $SENSORNAME_$ROLE state.highstate
  ```

### 13.6.8 Threshold

You can manage `threshold.conf` for `Suricata` using `Salt` pillars. The format of the pillar file can be seen below, as well as in `/opt/so/saltstack/local/pillar/thresholding/pillar.usage` and `/opt/so/saltstack/local/pillar/thresholding/pillar.example`

Usage:
thresholding:
  sids:
    <signature id>:
      - threshold:
          gen_id: <generator id>
          type: <threshold | limit | both>
          track: <by_src | by_dst>
          count: <count>
          seconds: <seconds>
      - rate_filter:
          gen_id: <generator id>
          track: <by_src | by_dst | by_rule | by_both>
          count: <count>
          seconds: <seconds>
          new_action: <alert | pass>
          timeout: <seconds>
      - suppress:
          gen_id: <generator id>
          track: <by_src | by_dst | by_either>
          ip: <ip | subnet>

Example:

thresholding:
  sids:
    8675309:
      - threshold:
          gen_id: 1
          type: threshold
          track: by_src
          count: 10
          seconds: 10
      - threshold:
          gen_id: 1
          type: limit
          track: by_dst
          count: 100
          seconds: 30
      - rate_filter:
          gen_id: 1
          track: by_rule
          count: 50
          seconds: 30
          new_action: alert
          timeout: 30
      - suppress:
          gen_id: 1
          track: by_either
          ip: 10.10.3.7
    11223344:
      - threshold:
          gen_id: 1
          type: limit
          track: by_dst
          count: 10
          seconds: 10
      - rate_filter:

(continues on next page)
gen_id: 1
track: by_src
count: 50
seconds: 20
new_action: pass
timeout: 60
- suppress:
  gen_id: 1
  track: by_src
  ip: 10.10.3.0/24

In order to apply the threshold to all nodes, place the pillar in `/opt/so/saltstack/local/pillar/global.sls`. If you want to apply the threshold to a single node, place the pillar in `/opt/so/saltstack/local/pillar/minions/<MINION_ID>.sls`

13.6.9 Suppressions

A suppression rule allows you to make some finer grained decisions about certain rules without the onus of rewriting them. With this functionality we can suppress rules based on their signature, the source or destination address and even the IP or full CIDR network block. This way, you still have the basic ruleset, but the situations in which they fire are altered. It’s important to note that with this functionality, care should be given to the suppressions being written to make sure they do not suppress legitimate alerts. See above for suppress examples.

13.6.10 Why is idstools ignoring disabled rules

If your syntax is correct, you are likely trying to disable a rule that has flowbits set. For a quick primer on flowbits see [http://blog.snort.org/2011/05/resolving-flowbit-dependancies.html](http://blog.snort.org/2011/05/resolving-flowbit-dependancies.html) and section 3.6.10 of the Snort Manual (http://www.snort.org/docs).

Let’s look at the following rules using:

```
alert tcp $HOME_NET any -> $EXTERNAL_NET !1433 (msg:"ET POLICY Outbound MSSQL \n-\nConnection to Non-Standard Port - Likely Malware"; flow:to_server,established;
  content:"|12 01 00|"; depth:3; content:"|00 00 00 00 00 00 15 00 06 01 00 1b 00 01 02 00 1c 00|
  distance:1; within:18; content:"|03 00|"; distance:1; within:2;
  content:"|00 04 ff 08 00 01 55 00 00 00|"; distance:1; within:10; flowbits:set,ET.MSSQL;
  classstype:bad-unknown; sid:2013409; rev:3;)
alert tcp $HOME_NET any -> $EXTERNAL_NET 1433 (msg:"ET POLICY Outbound MSSQL \n-\nConnection to Standard port (1433)"; flow:to_server,established; content:"|12 01 00|
  "; depth:3; content:"|00 00 00 00 00 00 15 00 06 01 00 1b 00 01 02 00 1c 00|
  distance:1; within:18; content:"|03 00|"; distance:1; within:2;
  content:"|00 04 ff 08 00 01 55 00 00 00|"; distance:1; within:10; flowbits:set,ET.MSSQL;
  classstype:bad-unknown; sid:2013410; rev:4;)
alert tcp $HOME_NET any -> $EXTERNAL_NET !1433 (msg:"ET TROJAN Bancos.DV MSSQL CnC \n-\nConnection Outbound"; flow:to_server,established; flowbits:isset,ET.MSSQL; content:
  "|49 00 B4 00 4D 00 20 00 54 00 48 00 45 00 20 00 4D 00 41 00 53 00 54 00 45 00 52|
  00|"; classstype:trojan-activity; sid:2013411; rev:1;)
```

If you try to disable the first two rules without disabling the third rule (which has “flowbits:isset...”) the third rule could never fire due to one of the first two rules needing to fire first. Pulled Pork (helpfully) resolves all of your flowbit dependencies, and in this case, is “re-enabling” that rule for you on the fly. Disabling all three of those rules by adding the following to disablesid.conf has the obvious negative effect of disabling all three of the rules:
When you run `sudo so-rule-update`, watch the “Setting Flowbit State…” section and you can see that if you disable all three (or however many rules share that flowbit) that the “Enabled XX flowbits” line is decremented and all three rules should then be disabled in your `all.rules`.

### 13.7 High Performance Tuning

#### 13.7.1 CPU Affinity/Pinning

For best performance, CPU intensive processes like Zeek and Suricata should be pinned to specific CPUs. In most cases, you’ll want to pin sniffing processes to the same CPU that your sniffing NIC is bound to.

*Suricata Performance*

*Zeek Performance*

#### 13.7.2 Misc

Consider adopting some of the suggestions from here:

- https://github.com/pevma/SEPTun
- https://github.com/pevma/SEPTun-Mark-II

#### 13.7.3 RSS

Check your sniffing interfaces to see if they have Receive Side Scaling (RSS) queues. If so, you may need to reduce to 1:


#### 13.7.4 Disk/Memory

If you have plenty of RAM, disable swap altogether.

Use `hdparm` to gather drive statistics and alter settings, as described here:


`vm.dirty_ratio` is the maximum amount of system memory that can be filled with dirty pages before everything must get committed to disk.

`vm.dirty_background_ratio` is the percentage of system memory that can be filled with “dirty” pages, or memory pages that still need to be written to disk – before the pdflush/flush/kdmflush background processes kick in to write it to disk.
More information:
https://lonesysadmin.net/2013/12/22/better-linux-disk-caching-performance-vm-dirty_ratio/

13.7.5 Elastic

You will want to make sure that each part of the pipeline is operating at maximum efficiency. Depending on your configuration, this may include Filebeat, Logstash, Redis, and Elasticsearch.
This section is a collection of miscellaneous tricks and tips for Security Onion.

14.1 Airgap

Security Onion is committed to allowing users to run a full install on networks that do not have Internet access. Starting in 2.x RC3, users will see an option to specify that the install is airgapped. The installer will then make the appropriate modifications to make this work properly.

14.1.1 Key Differences

By selecting Airgap as an install option, a couple of things happen that are different than a normal install with Internet access. First, all CentOS repos are removed and replaced with a new repo that runs on the manager. During the install, all of the necessary RPMs are copied from the ISO to a new repo located in /nsm/repo/. All devices in the grid will now use this repo for updates to packages. Another difference is the latest ET Open rules from Emerging Threats are copied to /nsm/repo/rules/ so that the manager can access them. This allows users to use the standard SO process for managing SIDS etc. Finally, yara rules for Strelka are copied to /nsm/repo/rules/strelka/ so that Strelka has the latest and greatest rules for static file analysis.

14.1.2 Updating

Starting in version 2.3, soup will automatically detect that you are upgrading an airgap install and will ask for the location of the upgrade disk. This can be done by burning the disk and putting it in the DVD drive, burning a USB stick like any standard ISO, or simply copying the ISO file to the airgapped manager.

14.1.3 Updating from RC3
Note: If upgrading from RC3 there is an extra step that needs to take place to copy over the proper version of soup in order to complete the update. To accomplish this you need to run the following commands.

- Create a temp directory:
  ```bash
  mkdir -p /tmp/sotemp
  ```

- If using a DVD with the image burned to it:
  ```bash
  sudo mount /dev/cdrom /tmp/sotemp
  ```

  Otherwise, if using an ISO file:
  ```bash
  sudo mount -t iso9660 -o loop /home/user/securityonion-2.3.0.iso /tmp/sotemp
  ```

- Copy the new version of `soup`:
  ```bash
  sudo cp /tmp/sotemp/SecurityOnion/salt/common/tools/sbin/soup /opt/so/saltstack/default/salt/common/tools/sbin/
  ```

- Update `Salt`:
  ```bash
  sudo salt-call state.apply common
  ```

- Unmount the temp directory:
  ```bash
  sudo umount /tmp/sotemp
  ```

- Run the new version of `soup`
  ```bash
  sudo soup
  ```

## 14.2 Cloud AMI

Security Onion VPC Traffic Mirroring Configuration

**Warning:** THE CLOUD AMI IS NOT YET AVAILABLE BUT IS COMING SOON!

This section covers configuring a Security Onion cloud image hosted in Amazon Web Services (AWS) to receive mirrored/spanned traffic from other instances hosted within an Amazon Virtual Private Cloud (VPC).

**Tip:** You can only mirror traffic from an EC2 instance that is powered by the AWS Nitro system. For a list of supported Nitro systems, please see [https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/instance-types.html#ec2-nitro-instances](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/instance-types.html#ec2-nitro-instances).

**Note:** This section does not cover network connectivity to the Security Onion node. This can be achieved through configuring an external IP for the node’s management interface, or through the use of a VPN connection.
via OpenVPN, pfSense etc. For more details about VPN connections, please see https://medium.com/@svfusion/setup-site-to-site-vpn-to-aws-with-pfsense-1cac16623bd6.

This section does not cover how to set up a VPC in AWS. For more details about setting up a VPC, please see https://docs.aws.amazon.com/directoryservice/latest/admin-guide/gsg_create_vpc.html.

### 14.2.1 Getting Started

To setup the Security Onion AMI and VPC mirror configuration, use the steps below.

**Create a Security Group for Sniffing Interface**

Security Groups act like a firewall for your Amazon EC2 instances controlling both inbound and outbound traffic. We will need to create a security group specifically for the interface we will be using to sniff the traffic. This security group will need to be as open as possible to ensure all traffic destined to the sniffing interface will be allowed through.

To create a security group, follow these steps:

- From the EC2 Dashboard Select: Security Groups under the Network & Security sections in the left window pane.
- Select: Create Security Group
- Provide a Security Group Name and Description.
- Select the appropriate VPC for the security group.
- With the inbound tab selected, select: Add Rule
- Add the appropriate inbound rules to ensure all desired traffic destined for the sniffing interface is allowed.
- Select: Create

**Create Sniffing Interface**

Prior to launching the Security Onion AMI you will need to create the interface that will be used to monitor your VPC. This interface will be attached to the Security Onion AMI as a secondary interface. To create a sniffing interface, follow these steps:

- From the EC2 Dashboard Select: Network Interfaces under the Network & Security section in the left window pane.
- Select: Create Network Interface
- Provide a description and choose the appropriate subnet you want to monitor.
- Select the security Group that you created for the sniffing interface.
- Select: Create

**Create a Security Onion EC2 instance in Amazon Web Services (AWS)**

To configure a Security Onion instance, follow these steps:

- From the EC2 dashboard select: Launch Instance
- Search the AWS Marketplace for Security Onion and make sure you get the latest version of our official AMI.

---

14.2. Cloud AMI
Choose the appropriate instance type based on the desired hardware requirements and select Next: Configure Instance Details. For assistance on determining resource requirements please visit our Hardware Requirements section.

From the subnet drop-down menu select the same subnet as the sniffing interface.

Under the Network interfaces section configure the eth0 (management) interface.

Under the Network interfaces section select: Add Device to attach the previously created sniffing interface to the instance.

From the Network Interface drop-down menu for eth1 choose the sniffing interface you created for this instance. Please note if you have multiple interfaces listed you can verify the correct interface by navigating to the Network Interfaces section in the EC2 Dashboard.

Select: Next: Add Storage and configure the volume settings.

Select: Next: Add Tags and add any additional tags for the instance.

Select: Next: Configure Security Group and add the appropriate inbound rules.

Select: Review and Launch

If prompted, select the appropriate SSH keypair that will be used to ssh into the Security Onion instance for administration

The default username for the Security-Onion-2.3 AMI is: onion

Once you have logged in, please do the following steps.

Change the hostname (Optional):

```
sudo vim /etc/hostname
```

Update packages:

```
sudo soup
```

Run through both phases of setup:

```
sudo sosetup
```

**Traffic Mirroring**

Traffic mirroring allows you to copy the traffic to/from an instance and send it to the sniffing interface of a network security monitoring sensor or a group of interfaces using a network load balancer. For more details about AWS Traffic Mirroring please see: https://docs.aws.amazon.com/vpc/latest/mirroring/what-is-traffic-mirroring.html

**Create Mirror Target**

A mirror target in AWS refers to the destination for the mirrored traffic. This can be a single interface or a group of interfaces using a network load balancer. To configure a mirror target, follow these steps:

- From the VPC dashboard select: Mirror Targets under the Traffic Mirroring section in the left window pane.

- Select: Create traffic mirror target

- Under the Choose target section select the appropriate target type and choose the sniffing interface connected to the Security Onion instance. For more details about traffic mirror targets please see: https://docs.aws.amazon.com/vpc/latest/mirroring/traffic-mirroring-targets.html
Create Mirror Filter

A mirror filter allows you to define the traffic that is copied to in the mirrored session and is useful for tuning out noisy or unwanted traffic. To configure a mirror filter, follow these steps:

- From the VPC dashboard select: Mirror Filters under the Traffic Mirroring section in the left window pane.
- Select: Create traffic mirror filter
- Add the appropriate inbound and outbound rules. For more details about traffic mirror filters please see: https://docs.aws.amazon.com/vpc/latest/mirroring/traffic-mirroring-filters.html
- Select: Create

Create Mirror Session

A traffic mirror session defines the source of the traffic to be mirrored based on the selected traffic mirror filters and sends that traffic to the desired traffic mirror target. For more details about traffic mirror sessions please see: https://docs.aws.amazon.com/vpc/latest/mirroring/traffic-mirroring-session.html

- From the VPC dashboard select: Mirror Sessions under the Traffic Mirroring section in the left window pane.
- Select: Create traffic mirror session
- Under the Mirror source section, choose the interface that you want to be mirrored.
- Under the Mirror target section, choose the interface or load balancer you want to send the mirrored traffic to.
- Assign a session number under the Additional settings section for the mirror session.
- In the filters section under Additional settings choose the mirror filter you want to apply to the mirrored traffic.
- Select: Create

Verify Traffic Mirroring

To verify the mirror session is sending the correct data to the sniffing interface run the following command on the Security Onion instance:

```
sudo tcpdump -nni <interface>
```

You should see VXLAN tagged traffic being mirrored from the interface you selected as the Mirror Source.

To verify Zeek is properly decapsulating and parsing the VXLAN traffic you can verify logs are being generated in the `/nsm/zeek/logs/current` directory:

```
lsof -la /nsm/zeek/logs/current/
```

14.3 Docker

From https://www.docker.com/what-docker:
Docker is the world’s leading software container platform. Developers use Docker to eliminate “works on my machine” problems when collaborating on code with co-workers. Operators use Docker to run and manage apps side-by-side in isolated containers to get better compute density. Enterprises use Docker to build agile software delivery pipelines to ship new features faster, more securely and with confidence for both Linux, Windows Server, and Linux-on-mainframe apps.

14.3.1 Download

Our Docker images are stored on Docker Hub:
https://hub.docker.com/u/securityonion/

If you download our Security Onion ISO image, the Docker engine and these Docker images are baked right into the ISO image.

If you instead use another ISO image, our installer will download Docker images as necessary.

14.3.2 Security

To prevent tampering, our Docker images are signed using Docker Notary:
https://docs.docker.com/notary/getting_started/

Any time we push an image to Docker Hub, we explicitly set --disable-content-trust=false to sign the image using Docker Notary.

Any time we download an image from Docker Hub, we also explicitly set --disable-content-trust=false to verify that signature using Docker Notary.

14.3.3 Elastic

To maintain a high level of stability, reliability, and support, our Elastic Docker images are based on the Docker images provided by Elastic.co. Their Docker images are built on CentOS 7: https://www.elastic.co/blog/docker-base-centos7

14.3.4 Registry

The manager node runs a Docker registry. From https://docs.docker.com/registry/recipes/mirror/:

If you have multiple instances of Docker running in your environment (e.g., multiple physical or virtual machines, all running the Docker daemon), each time one of them requires an image that it doesn’t have it will go out to the internet and fetch it from the public Docker registry. By running a local registry mirror, you can keep most of the redundant image fetch traffic on your local network.

14.3.5 Networking and Bridging

By default, Docker configures its bridge with an IP of 172.17.0.1.
https://docs.docker.com/engine/userguide/networking/#default-networks

For many folks this is fine, but what if we actually use the the 172.17.0.0/16 range within our internal network(s)? This results in a conflict when trying to assign IP addresses to interfaces and trying to route outside of the host.
It is currently possible to change this at install time. Once you change this default docker network you **MUST** configure all nodes in the grid to use this range:

- During setup choose change docker network range.
- Enter your desired address range. You do not need the /24 at the end.

### 14.3.6 Containers

Our Docker containers all belong to a common Docker bridge network, called `so-elastic-net`. Each container is also aliased, so that communication can occur between the different docker containers using said alias. For example, communication to the `so-elasticsearch` container would occur through an alias of `elasticsearch`.

You may come across interfaces in `ifconfig` with the format `veth*`. These are the external interfaces for each of the Docker containers. These interfaces correspond to internal Docker container interfaces (within the Docker container itself).

To identify which external interface belongs to which container, we can do something like the following:

From the host, type:

```
sudo docker exec so-elasticsearch cat /sys/class/net/eth0/iflink
```

This should provide you with a value with which you can grep the host `net class ifindex(es)`:

**Example:**

```
grep 25 /sys/class/net/veth*/ifindex | cut -d'/ ' -f5
```

You should then receive some output similar to the following:

```
vethc5ff027
```

where `vethc5ff027` is the **external interface** of `eth0` **within the so-elasticsearch container**.

### 14.3.7 VMware Tools

If you have VMware Tools installed and you suspend and then resume, the Docker interfaces will no longer have IP addresses and the Elastic stack will no longer be able to communicate. One workaround is to remove `/etc/vmware-tools/scripts/vmware/network` to prevent VMware suspend/resume from modifying your network configuration.

### 14.3.8 Dependencies

**TheHive / Cortex**

`so-thehive` - **REQ** - TheHive Web App  
`so-thehive-cortex` - **OPT** - Cortex Web App  
`so-thehive-es` - **REQ** - TheHive & Cortex state data
Fleet

so-fleet - REQ - Fleet Web App
so-mysql - REQ - Fleet state data
so-redis - REQ - Required for live querying

Playbook

so-playbook - REQ - Playbook Web App
so-navigator - OPT - Navigator Web App
so-soctopus - REQ - Automation

SOCtopus

so-soctopus - REQ - SOCtopus App
so-elasticsearch - OPT - Automation

Suricata

so-suricata - REQ - Suricata app

Kibana

so-kibana - REQ - Kibana Web App
so-elasticsearch - REQ -

Zeek

so-bro - REQ - Zeek app

14.4 DNS Anomaly Detection

Dr. Johannes Ullrich of the SANS Internet Storm Center posted a great DNS Anomaly Detection script based on the query logs coming from his DNS server. We can do the same thing with Zeek’s dns.log (where Zeek captures all the DNS queries it sees on the network).

Note: Please note that the following script is only intended for standalone machines and will not work properly on distributed deployments. Another option which might work better ElastAlert and its new_term rule.

Thanks to senatorhotchkiss on our mailing list for updating the original script to replace bro-cut with jq:

#!/bin/bash

ZEEK_LOGS="/nsm/zeek/logs"
TODAY=`date +%Y-%m-%d`

(continues on next page)
14.5 ICMP Anomaly Detection

At Security Onion Conference 2016, Eric Conrad shared some IDS rules for detecting unusual ICMP echo requests/replies and identifying C2 channels that may utilize ICMP tunneling for covert communication.

14.5.1 Usage

We can add the rules to `/opt/so/rules/nids/local.rules` and the variables to `suricata.yaml` so that we can gain better insight into ICMP echoes or replies over a certain size, containing particularly suspicious content, etc.

14.5.2 Presentation

You can find Eric’s presentation here:

14.5.3 Download

You can download the rules here:
https://drive.google.com/file/d/0ByeHgv6rpa3gUDNuMUdobFBCNk

14.6 Adding a new disk

Before doing this in production, make sure you practice this on a non-production system!

There are at least 3 different ways to do this:
14.6.1 Method 1: LVM (Logical Volume Management)

The easiest method of adding disk space is using LVM (assuming you installed using LVM).

14.6.2 Method 2: Mount a separate drive to /nsm

If you aren’t using LVM, you can mount a drive directly to /nsm. If doing this after installation, you will need to stop services, move data, and then restart services.

14.6.3 Method 3: Make /nsm a symlink to the new logging location

Another option is to make /nsm a symbolic link to the new logging location. If doing this after installation, you will need to stop services, move data, and then restart services. Certain services like AppArmor may need special configuration to handle the symlink.

14.7 PCAPs for Testing

14.7.1 Links

- https://www.malware-traffic-analysis.net/
- https://www.netresec.com/?page=PcapFiles
- https://www.netresec.com/?page=MACCDC
- https://github.com/zeek/zeek/tree/master/testing/btest/Traces
- https://wiki.wireshark.org/SampleCaptures
- https://www.stratosphereips.org/datasets-overview
- https://ee.lbl.gov/anonymized-traces.html
- https://redmine.openinfosecfoundation.org/projects/suricata/wiki/Public_Data_Sets
- https://forensicscontest.com/puzzles
- https://archive.wrccdc.org/
- https://github.com/chrissanders/packets

14.7.2 Downloading

You can download pcaps from the link above using a standard web browser or from the command line using a tool like wget or curl. Here are some examples.

To download the pcap from https://www.malware-traffic-analysis.net/2020/09/16/index.html using wget:
wget https://www.malware-traffic-analysis.net/2020/09/16/2020-09-16-Qakbot-infection-traffic.pcap.zip

To download a pcap from https://www.netresec.com/?page=MACCDC:

wget https://download.netresec.com/pcap/maccdc-2012/maccdc2012_00000.pcap.gz

14.7.3 tcpreplay

You can use `tcpreplay` to replay any standard pcap to the sniffing interface of your Security Onion sensor.

14.7.4 so-import-pcap

A drawback to using tcpreplay is that it’s replaying the pcap as new traffic and thus the timestamps that you see in Kibana and other interfaces do not reflect the original timestamps from the pcap. To avoid this, a new tool was developed called `so-import-pcap`.

14.8 Removing a Node

There may come a time when you need to remove a node from your distributed deployment. To do this, log into your manager and list all Salt keys:

```
sudo salt-key
```

Then remove the node by deleting its key from Salt (replacing nodename with the actual node name):

```
sudo salt-key -d nodename
```

Remove the node from Grafana, by removing the reference in `/opt/so/saltstack/local/pillar/data/sensorstab.sls`, and the file in `/opt/so/conf/grafana/grafana_dashboards/sensor_nodes/<hostname>-Node.json` on the manager, then restart Grafana with:

```
sudo so-grafana-restart
```

14.8.1 Remove search node

If you are removing a search node, you will want to remove it from cross cluster search. In Kibana, navigate to Dev Tools and paste the following text into the window (replacing nodename with the actual node name):

```
PUT _cluster/settings
{
   "persistent": {
      "search": {
         "remote": {
            "nodename": {
               "skip_unavailable": null,
               "seeds":null
            }
         }
      }
   }
}
```

(continues on next page)
Click the play button to send the request to Elasticsearch.

### 14.9 Syslog Output

If you want to send logs to an external system, you can configure Logstash to output to syslog.

**See also:**

For more information about Logstash’s syslog output plugin, please see https://www.elastic.co/guide/en/logstash/current/plugins-outputs-syslog.html.

Please keep in mind that we don’t provide free support for third party systems.

### 14.10 UTC and Time Zones

When you run Security Onion Setup, it sets the timezone to UTC/GMT. Logging in UTC is considered a best practice across the cybersecurity industry because it makes it that much easier to correlate events across different systems, organizations, or time zones. Additionally, it avoids issues with time zones that have daylight savings time which would result in a one-hour time warp twice a year.

Web interfaces like Kibana should render those UTC timestamps in the timezone of your local browser.
This section covers some of the main utilities in Security Onion.

15.1 jq

From https://stedolan.github.io/jq/:

jq is like sed for JSON data - you can use it to slice and filter and map and transform structured data with the same ease that sed, awk, grep and friends let you play with text.

15.1.1 Usage

We configure Zeek to write logs to /nsm/zeek/logs/ in JSON format. If you want to parse those logs from the command line, then you can use jq. Here’s a basic example:

```bash
jq "." /nsm/zeek/logs/current/conn.log
```

This command will parse all of the records in /nsm/zeek/logs/current/conn.log. For each of the records, it will then output every field and its value.

15.1.2 More Information

See also:

For more information about jq, please see https://stedolan.github.io/jq/.

15.2 so-allow

Security Onion locks down the Firewall by default. Depending on what kind of installation you do, Setup may walk you through allowing your analyst IP address(es). If you need to add other analyst IP addresses or open firewall ports
for agents or syslog devices, you can run `sudo so-allow` and it will walk you through this process.

This program allows you to add a firewall rule to allow connections from a new IP address.

Choose the role for the IP or Range you would like to add

[a] - Analyst - ports 80/tcp and 443/tcp
[b] - Logstash Beat - port 5044/tcp
[o] - Osquery endpoint - port 8090/tcp
[s] - Syslog device - 514/tcp/udp
[w] - Wazuh agent - port 1514/tcp/udp
[p] - Wazuh API - port 55000/tcp
[r] - Wazuh registration service - 1515/tcp

Please enter your selection (a - analyst, b - beats, o - osquery, w - wazuh):

15.2.1 Wazuh

If you choose the analyst option, so-allow will also add the analyst IP address to the Wazuh safe list. This will prevent Wazuh Active Response from blocking the analyst IP address.

15.3 so-import-pcap

so-import-pcap will import one or more pcaps into Security Onion and preserve original timestamps.

It will do the following:

- generate IDS alerts using *Suricata*
- generate network metadata using *Zeek*
- store IDS alerts and network metadata in *Elasticsearch* with original timestamps
- store pcaps where Security Onion Console (SOC) can find them

15.3.1 Usage

**Warning:** so-import-pcap works differently on Security Onion 2 than it did in previous versions!

This new version of so-import-pcap requires you to run through Setup and choose a configuration that supports so-import-pcap. This includes Import Node and other nodes that include sensor services like Eval and Standalone. The quickest and easiest option is to choose Import Node which gives you the minimal services necessary to import a pcap. so-import-pcap then provides a hyperlink for you to view all alerts and logs in *Hunt*. You can also find NIDS alerts in *Alerts* and all logs in *Kibana*.

Once Setup completes, you can then run `sudo so-import-pcap` and supply the full path to at least one pcap file. For example, to import a single pcap named `import.pcap`:

```
sudo so-import-pcap /full/path/to/import.pcap
```

To import multiple pcaps:
If you don’t already have some pcap files to import, see PCAPs for Testing for a list of sites where you can download sample pcaps.

15.4 so-zeek-logs

If you want to specify what Zeek logs are ingested, you can use so-zeek-logs. It will show you a list of all Zeek logs and you can specify which of those logs are ingested. Once you’ve made your selection, it will modify the Filebeat configuration for you.
CHAPTER 16

Help

Having problems? Try the suggestions below.

• Have you run soup to ensure that you’re on the latest version?
• Check the FAQ.
• Search the Community Support forum.
• Search the documentation and support forums of the tools contained within Security Onion: Tools
• Check log files in /opt/so/log/ or other locations for any errors or possible clues:
  – Setup /root/sosetup.log
  – Suricata /opt/so/log/suricata/suricata.log
  – Zeek /nsm/zeek/logs/current/
  – Elasticsearch /opt/so/log/elasticsearch/<hostname>.log
  – Kibana /opt/so/log/kibana/kibana.log
  – Logstash /opt/so/log/logstash/logstash.log
  – Elastalert /opt/so/log/elastalert/elastalert_stderr.log
• Are you able to duplicate the problem on a fresh Security Onion installation?
• Check the Known Issues to see if this is a known issue that we are working on.
• If all else fails, please feel free to reach out for Support.

16.1 FAQ

Install / Update / Upgrade
Users / Passwords
16.1.1 Install / Update / Upgrade

Why won’t the ISO image boot on my machine?
Please see the TroubleBooting section.

What’s the recommended procedure for installing Security Onion?
Please see the Installation section.

What languages are supported?
We only support the English language at this time.

How do I install Security Onion updates?
Please see the soup section.

What do I need to do if I’m behind a proxy?
Please see the Proxy Configuration section.

Can I run Security Onion on Raspberry Pi or some other non-x86 box?
No, we only support 64-bit Intel/AMD architectures. Please see the Hardware Requirements section.

16.1.2 Users / Passwords

What is the password?
Please see the Passwords section.
How do I add a new user account?

Please see the Adding Accounts section.

16.1.3 Support / Help

Where do I send questions/problems/suggestions?

Please see the Community Support section.

Is commercial support available for Security Onion?

Yes, we offer commercial support at https://securityonionsolutions.com.

16.1.4 IDS engines

Can Security Onion run in IPS mode?

We do not support IPS.

16.1.5 Security Onion internals

Where can I read more about the tools contained within Security Onion?

Please see the Tools section.

What’s the directory structure of /nsm?

Please see the Directory Structure section.

Why does Security Onion use UTC?

Please see the UTC and Time Zones section.
Why are the timestamps in Kibana not in UTC?
Please see the UTC and Time Zones section.

Why is my disk filling up?
Security Onion records full packet capture to disk.

16.1.6 Tuning

How do I configure email for alerting and reporting?
Please see the Email Configuration section.

How do I configure a BPF?
Please see the BPF section.

How do I filter traffic?
Please see the BPF section.

How do I exclude traffic?
Please see the BPF section.

What are the default firewall settings and how do I change them?
Please see the Firewall section.

What do I need to modify in order to have the log files stored on a different mount point?
Please see the Adding a new disk section.

16.1.7 Miscellaneous

Where can I find interesting pcaps to replay?
Please see the PCAPs for Testing section.

Why is Security Onion connecting to an IP address on the Internet over port 123?
Please see the NTP section.
Should I backup my Security Onion box?

Network Security Monitoring as a whole is considered “best effort”. It is not a “mission critical” resource like a file server or web server. Since we’re dealing with “big data” (potentially terabytes of full packet capture), backups would be prohibitively expensive. Most organizations don’t do any backups and instead just rebuild boxes when necessary.

How can I add and test local rules?

Please see theAdding local rules and testing them with scapy section.

Can I connect Security Onion to Active Directory or LDAP?

We understand the appeal of integrating with directory services like Active Directory and LDAP, but we typically recommend against joining any security infrastructure (including Security Onion) to directory services. The reason is that when you get an adversary inside your network, one of their first goals is going to be gaining access to that directory. If they get access to the directory, then they get access to everything connected to the directory. For that reason, we recommend that all security infrastructure (including Security Onion) be totally separate from directory services.

back to top

16.2 Directory Structure

16.2.1 /opt/so/conf

Applications read their configuration from /opt/so/conf/. However, please keep in mind that most config files are managed withSalt, so if you manually modify those config files, your changes may be overwritten at the next Salt update.

16.2.2 /opt/so/log

Debug logs are stored in /opt/so/log/.

16.2.3 /opt/so/rules

ElastAlert and Suricata rules are stored in /opt/so/rules/.

16.2.4 /opt/so/saltstack/local

Custom Salt settings can be added to /opt/so/saltstack/local/.

16.2.5 /nsm

The vast majority of data is stored in /nsm/.
16.2.6 /nsm/zeek

Zeek writes its protocol logs to /nsm/zeek/.

16.2.7 /nsm/elasticsearch

Elasticsearch stores its data in /nsm/elasticsearch/.

16.2.8 /nsm/pcap

Stenographer stores full packet capture in /nsm/pcap/.

16.2.9 /nsm/wazuh

All Wazuh files are stored in /nsm/wazuh/. For convenience, we have placed symlinks for Wazuh config at /opt/so/conf/wazuh/ (linked to /nsm/wazuh/etc) and Wazuh rules at /opt/so/rules/hids/ (local_rules.xml links to /nsm/wazuh/etc/rules/local_rules.xml and ruleset links to /nsm/wazuh/ruleset).

16.3 Tools

Security Onion would like to thank the following open-source projects for their contribution to our community!

(listed alphabetically)

ATT&CK Navigator
Cortex
Curator
CyberChef
Docker
domainstats
ElastAlert
Elasticsearch
Filebeat
Fleet
freqserver
Grafana
TheHive
Kibana
Logstash
osquery
Redis
16.4 Support

16.4.1 Paid Support

If you need private or priority support, please consider purchasing hardware appliances or support from Security Onion Solutions:

https://securityonionsolutions.com/support

**Tip:** Purchasing from Security Onion Solutions helps to support development of Security Onion as a free and open source platform!

16.4.2 Community Support

If you need free support, you can reach out to our *Community Support*.

16.5 Community Support

16.5.1 Check Documentation First

Before posting for help, check to see if your question has already been answered in the *Help* or *FAQ* sections.

16.5.2 Moderation

If at first you don’t see your post appear, it may have been queued for approval by the forum moderators. There is no need to re-submit your post. It will be reviewed and then approved if appropriate.

16.5.3 Etiquette

Please be courteous and respectful. Disrespectful messages can result in being banned from the forum.
16.5.4 Questions/Problems

Start a new discussion instead of replying to an old one

Please search the forum to see if you can find similar discussions that may help you. However, please do not reply to old discussions with your new issue. Instead, please start a new discussion and provide a hyperlink to the related discussion.

Avoid generic OS questions

Security Onion is based on a standard Linux distribution. In order to keep the signal-to-noise ratio as high as possible, the Security Onion forums should only be used for questions directly relating to Security Onion itself. If you have generic questions about Linux, you should search Google for other forums or relevant information.

Provide sufficient technical info

In order to be as effective and efficient as possible, please consider the following when posing your question/problem to the group:
http://www.chiark.greenend.org.uk/~sgtatham/bugs.html

Please include the following details where you can:

- Exact version. ex. 2.3.0
- Install source. Did you install from our Security Onion ISO image or did you perform a network installation?
- If network install, did you install from CentOS 7 or Ubuntu 18.04?
- Install type. ex. eval, standalone, forward node, search node, etc
- Does so-status show all services running?
- Do you get any failures when you run `salt-call state.highstate` from the node?
- Explain your issue. For example: Installation fails when I select this series of options...

16.5.5 Forum

Once you’ve read and understand all of the above, you can post your question to the community support forum:
https://github.com/Security-Onion-Solutions/securityonion/discussions

16.6 Help Wanted

Folks frequently ask how they can give back to the Security Onion community. Here are a few of our community teams that you can help with.

16.6.1 Marketing Team

We need more folks to help spread the word about Security Onion by blogging, tweeting, and other social media.
16.6.2 Support Team

If you’d like help out other Security Onion users, please join the forum and start answering questions!
https://github.com/Security-Onion-Solutions/securityonion/discussions

16.6.3 Documentation Team

If you find that some information in our Documentation is incorrect or lacking, please feel free to submit Pull Requests via GitHub!
https://github.com/Security-Onion-Solutions/securityonion-docs

16.6.4 Core Development

Most of our code is on GitHub. Please feel free to submit pull requests!
https://github.com/Security-Onion-Solutions

16.6.5 Thanks

The following folks have made significant contributions to Security Onion over the years. Thanks!

- Wes Lambert
- Mike Reeves
- Jason Ertel
- Josh Brower
- Josh Patterson
- Phil Plantamura
- William Wernert
- Bryant Treacle
- Dustin Lee
- Kevin Branch
- Scott Runnels
- Brad Shoop
- Paul Halliday
- Seth Hall
- Liam Randall
- Eric Ooi
- Lawrence Abrams
- Mark Hillick
- Joe Hargis
- Dennis Distler
• Jon Schipp
• Josh More
• Jack Blanchard
If you have any security concerns regarding Security Onion or believe you have uncovered a vulnerability, please follow these steps:

- send an email to security@securityonion.net
- include a description of the issue and steps to reproduce
- please use plain text format (no Word documents or PDF files)
- please do not disclose publicly until we have had sufficient time to resolve the issue

**Note:** This security address should be used only for undisclosed vulnerabilities. Dealing with fixed issues or general questions on how to use Security Onion should be handled via the normal Support channels.
This appendix covers the process of upgrading from the old Security Onion 16.04 to the new Security Onion 2.

**Warning:** Security Onion 2 is a MAJOR architectural change, so please note the following:

- Security Onion 2 has higher hardware requirements, so you should check that your hardware meets those requirements.
- Once you’ve upgraded from Ubuntu 16.04 to Ubuntu 18.04, you will essentially do a new installation of Security Onion 2 on top of Ubuntu 18.04. Very little data will be retained during the upgrade!
- There will be no way to migrate application accounts from Security Onion 16.04 to Security Onion 2.
- There will be no way to migrate sguild data from Security Onion 16.04 to Security Onion 2.
- You may need to purge pcap to make free space for the upgrade process. Any pcap remaining after the upgrade can only be accessed via tcpdump.
- We do not provide any guarantees that the upgrade process will work! If the upgrade fails, be prepared to perform a fresh installation of Security Onion 2.

For the reasons listed above, we recommend that most users procure new hardware and perform a fresh installation of Security Onion 2.

**Tip:** If you’re planning to purchase new hardware, please consider official Security Onion appliances from Security Onion Solutions (https://securityonionsolutions.com). Our custom appliances have already been designed for certain roles and traffic levels and have Security Onion pre-installed. Purchasing from Security Onion Solutions will save you time and effort and help to support development of Security Onion as a free and open source platform!

If you have reviewed all of the warnings above and still want to attempt an in-place upgrade, you should be able to do the following:
Warning: Please ensure you have local access to the machine being upgraded (console/DRAC/IPMI). Failure to do so could result in an unsuccessful upgrade, requiring a clean installation of Security Onion 2.

```
sudo soup
sudo reboot
```

Copy and paste the following:

```
sudo rm /etc/apt/sources.list.d/securityonion-ubuntu-stable-xenial.list && 
sudo so-stop && 
sudo service syslog-ng stop && 
sudo service mysql stop && 
sudo service salt-minion stop && 
sudo docker system prune -a -f && 
sudo sed -i 's|PREV="pre-.*$|PREV="pre-upgrade-to-18.04"|g' && 
       /var/lib/dpkg/info/securityonion-bro.preinst && 
 sudo /var/lib/dpkg/info/securityonion-bro.preinst install && 
 sudo apt install update-manager-core -y && 
 sudo sed -i 's|Prompt=never|Prompt=lts|g' /etc/update-manager/release-upgrades && 
 sudo pkill xscreensaver && 
 sudo do-release-upgrade -y
```

You may be interactively prompted to provide an answer to the following questions or similar during the upgrade:

```
Non-superusers capture PCAP -> No
login.defs -> Install package maintainer's version
grub -> Choose to keep local version
sshd_config -> Choose to keep local version
syslog-ng.conf -> Choose to keep local version
```

```
sudo reboot
```

AFTER UPGRADING TO 18.04:

```
sudo service apache2 stop && 
sudo systemctl disable apache2.service && 
sudo service mysql stop && 
sudo systemctl disable mysql.service && 
sudo ntpdate -u time.nist.gov && 
sudo apt autoremove -f --purge; done && 
for i in $(dpkg -l | grep securityonion | awk '{print $2}'); do sudo apt remove $i -y; 
   --purge; done && 
sudo mv /etc/salt/ /etc/salt_pre_upgrade && 
sudo mv /var/ossec /var/ossec_pre_upgrade && 
sudo apt purge salt-* -y && 
 sudo sed -i '/^$' /etc/cron.d/salt-update && 
 sudo apt install netplan.io -y && 
 sudo apt purge -y ifupdown && 
sudo rm /etc/network/interfaces* && 
sudo mv /nsm/zeek/spool/ /nsm/zeek/old_spool && 
sudo mv /nsm/zeek/logs/stats/ /nsm/zeek/logs/old_stats
```

On Distributed Manager - also do the following for Redis:
Apply netplan for the management interface in /etc/netplan/netplan.yaml (create the file and ensure that the extension is .yaml):

### If using DHCP (not recommended)

```yaml
network:
  version: 2
  renderer: networkd
  ethernets:
    ens18:
      dhcp4: true
```

### If using static IP:

```yaml
network:
  version: 2
  renderer: networkd
  ethernets:
    ens18:
      addresses:
        - 10.10.10.2/24
      gateway4: 10.10.10.1
      nameservers:
        search: [mydomain]
        addresses: [10.10.10.1, 1.1.1.1]
```

Other examples: https://netplan.io/examples/
sudo netplan apply (may disconnect after this command, so ensure local access is available)
sudo reboot
sudo nmcli con delete "Wired connection 1" (delete for later use as bond interface)

**Warning:** Don’t reboot yet!

Install Security Onion 2:
Skip to step 7, detailed here:
https://docs.securityonion.net/en/2.3/installation.html#installation-on-ubuntu-or-centos

Post-Installation:
While the files will still reside disk, config files/settings will NOT be migrated to the appropriate format/locations for Security Onion 2.

Example configuration may include:
- IDS Rule Oinkcode/Thresholds/Disablements (/etc/nsm/rules/threshold.conf, /etc/nsm/pulledpork)
- Custom Logstash config (/etc/logstash/custom)
- Custom Zeek scripts or BPFs (/opt/zeek/share/zeek/policy, /etc/nsm/rules/bpf.conf)
If you are viewing the online version of this documentation, you can click here for our Security Onion Cheat Sheet.

This was based on a cheat sheet originally created by Chris Sanders which can be found here: http://chrissanders.org/2017/06/security-onion-cheat-sheet/
Index

H
Host Groups, 137

P
Port Groups, 137