



Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"ICS/SCADA Security Essentials (Industrial Control Systems 410)"
at <http://www.giac.org/registration/gicsp>

Passive Analysis of Process Control Networks

GIAC (GICSP) Gold Certification

Author: Jennifer Ann Janesko, jennifer.janesko@secunet.com

Advisor: Adam Kliarsky

Accepted: May 28th, 2018

Abstract

In recent years there has been an increased push to secure critical ICS infrastructures by introducing information security management systems. One of the first steps in the ISMS lifecycle is to identify which assets are present in the infrastructure and to determine which ones are critical for operations. This is a challenge because, for various reasons, the documentation of the current state of ICS networks is often not up-to-date. Classic inventorying techniques such as active network scanning cannot be used to remedy this because ICS devices tend to be sensitive to unexpected network traffic. Active scanning of these systems can lead to physical damage and even injury. This paper introduces a passive network analysis approach to starting, verifying and/or supplementing an ICS asset inventory. Additionally, this type of analysis can also provide some insight into the ICS network's current security posture.

1. Introduction: ICS Security Challenge

In recent years there has been an increased push to secure critical infrastructures driven by process control networks. In August of 2015, for example, Germany enacted an amendment to its law, the *Energiewirtschaftsgesetz*. This amendment makes it a legal requirement for every electricity provider (generation, transmission, and distribution) to implement an information security management system (ISMS) (“Gesetz...”, 2015). New regulations such as these plus high-profile cases such as the 2015 Ukraine power grid attack (Lee, et. al., 2016) have raised awareness of the importance of protecting control systems and their associated process control networks.

In some ways, process control networks are similar to common IT networks. Components such as servers, gateways, and switches are common. In other ways, process control networks are different. They contain devices that are not commonly found in IT networks such as: programmable logic controllers (PLCs), human machine interfaces (HMIs), specialized devices such as remote terminal units (RTUs) and safety instrumented systems (SISs). These devices tend to be more sensitive than common IT components and require stable networks free of latency, jitter and interruption (Stoufer, et. al., 2015).

The question for critical infrastructure providers is how to start and maintain an ISMS. The first phase in the ISMS life cycle is “plan”. During this phase, an organization must perform a risk assessment. This includes the identification of assets and determination of whether or not these assets are critical (Allen, 2013). In the “do” part of the ISMS lifecycle, it is also required to keep an inventory of devices and services running on the networks along with their network configurations (“The 60 Minute Network Security Guide”, 2006; Allen, 2013). This becomes a challenge within process control networks because it is not always clearly documented which assets are connected to the network and how they are connected.

There are two major factors that lead to a lack of up-to-date documentation. ICS installations are often performed by third-party vendors who deliver documentation after-the-fact. If a larger installation is currently underway, then the documentation will not reflect the current state. In addition to this, ICS operators may make their own additions and modifications

to the network to solve day-to-day problems and not update the documentation to reflect these changes. This knowledge is instead often informally shared between colleagues.

In a standard IT network, this problem would be relatively straightforward to solve. An analyst can perform an active scan of the available devices, services, and network segments to get a snapshot of the network. In a process control network where an ISMS is just being introduced, this type of scan could result in injury or damage if performed. ICS networks and devices can be extremely sensitive to unexpected network traffic. Therefore, a different approach to ICS asset identification in an organization with a nascent security posture is required.¹

This paper will introduce and outline a practical, passive data collection and analysis approach for performing an asset inventory of process control networks. This approach will utilize a collection of open-source and freely-available tools. This technique also provides a means by which an analyst can check for rudimentary security misconfigurations and potential malicious activity in the process control network under investigation.

This passive data collection and analysis approach has two major phases. In the first phase, network traffic data must be safely captured. In the second phase, the captured traffic is loaded into analysis tools to identify: assets in the network, security and configuration weaknesses and potential active threats. Chapter 2 addresses the safe capture of network traffic data. Chapter 3 addresses the analysis of the captured data.

2. Capturing Network Data

The first phase in passive analysis is to collect data from the network. There are four possible opportunities for network traffic capture.

The first possibility is to get a network capture in the form of PCAP or PCAPNG files from the customer. It is often the case that either the customer or the customer's vendor has captured network traffic for debugging or monitoring purposes, and these files already exist. The advantage, in this case, is that the network captures have already been done. The disadvantage is

¹It has been argued that active scanning and testing of components inside of process control networks is a viable testing technique. This technique is used only in organizations with an established security posture and with appropriate precautionary measures beforehand (Peterson, 2016; Chason, et. al., 2014).

that the network captures may have focused on resolving a network problem rather than having been optimized for inventorying purposes.

If the customer does not have suitable network captures already available, then it is up to the analyst to gather the necessary data. In modern networks, there are usually two options for capturing data over a modern network: via a monitoring port on a switch (also known as a mirroring or SPAN port) or via a network tap. Monitoring ports are ports on switches that have been configured to listen to the traffic from one or more other active ports on the switch. This approach is illustrated in Figure 1.

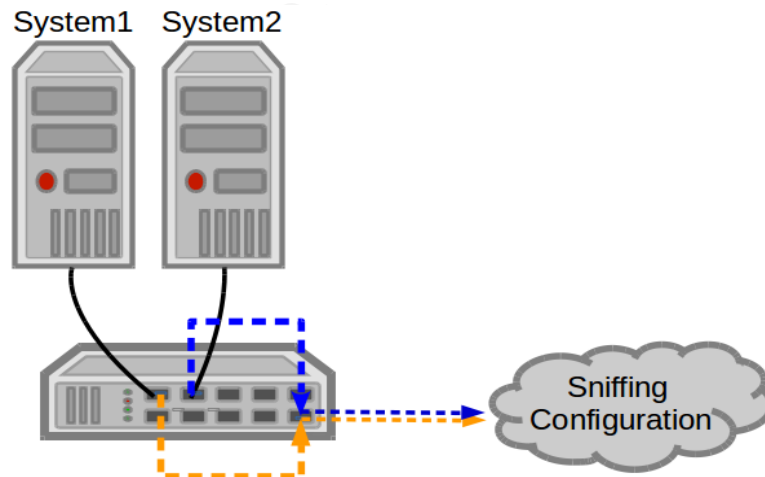


Figure 1: Illustration of a Monitoring Port

In Figure 1, the network traffic to and from System 1 and 2 on the switch is also sent to a monitoring port. This traffic can then be captured by a sniffer over the monitoring port without physical modification of the network.

Another further possibility for capturing network data is via a LAN tap device. LAN taps are passive devices that must be physically inserted between network devices and can capture all of the network traffic between those devices. Figure 2 provides an illustration of a LAN tap.

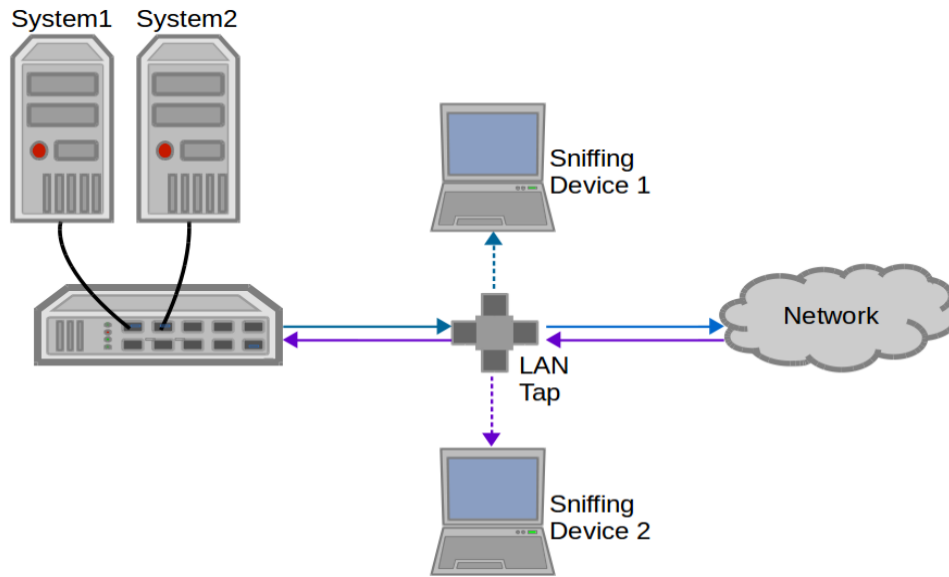


Figure 2: Physical Sniffing Configuration with Network Tap

LAN taps have four total ports. Two ports allow regular network traffic to pass from one part of a network segment to another. The other two ports are special. Each port forwards traffic from *one direction of normal traffic flow* to a sniffing device. For example, the sniffing port with the blue dotted line in Figure 2 forwards only the traffic that goes from switch to the network. The port with the purple dotted line forwards only the traffic from the network to the switch.

Whether or not to use monitoring ports or LAN taps is dependent on the network being analyzed. The advantages and disadvantages of each are captured in Table 1 below.

	Advantages	Disadvantages
Monitoring Port	<ul style="list-style-type: none"> • No network disconnection is required. • Multiple ports can be configured to be monitored. • A monitoring port captures network traffic in both directions on one interface. 	<ul style="list-style-type: none"> • If the switch becomes too busy, it may stop mirroring network traffic. • If monitoring port becomes too busy, it may drop packets. • Packets with network errors are often dropped by monitoring ports. • Misconfiguration of monitoring ports can lead to network problems. • A monitoring port can allow traffic from the sniffing device to enter the network.

	Advantages	Disadvantages
LAN Tap	<ul style="list-style-type: none"> • A tap device captures all network traffic (including packets with network errors). • The tap device prevents the sniffing device from transmitting any packets to the network. This reduces the risk of network disruption. 	<ul style="list-style-type: none"> • To capture network traffic in two directions, the network sniffer must have two network cards (or two computers); one for each direction of traffic. • To apply a LAN tap, the network must be temporarily disconnected to insert the tap device. • A LAN tap can only capture traffic from one network pathway.

Table 1: Monitoring Port vs. LAN Tap: Advantages and Disadvantages

The final network traffic capture possibility is only applicable in older infrastructures. In older process control networks, a network may use hubs rather than switches, routers and gateways. Since network hubs rebroadcast network traffic on all ports, a mirroring port would not need to be configured. In this final option of collecting data, the analyst would need to set up a sniffing interface on an open port of the hub. Because all of traffic is broadcast to every port, the sniffing device would be able to capture all of the traffic for all of the devices connected to the hub.

2.1. Selecting the Target for Network Capture

To perform a passive asset inventory, a strategic point or set of points should be selected in the network. Good areas to target include switches and network segments that contain traffic to and from centralized control systems. If the network is distributed, it may be necessary to travel to more than one location and set up sniffers in remote locations. In those cases, the analyst must consider whether those locations are practically reachable, whether or not specialized certification is required to enter the premises and whether or not personal protective equipment is needed.

2.2 Connecting to the Network

Once the strategy of gathering network traffic has been selected, care should be given to connect to the process control network safely. The analyst should avoid introducing traffic to the

network. To avoid introducing any new traffic to the network that could have damaging consequences, the analyst should work with a LAN tap. As mentioned earlier in this chapter, a tap is normally inserted between two network devices or two segments of a network. LAN taps may also be effectively used with a monitoring port or a hub.

When connecting to a monitoring port or hub with a LAN tap, four network devices are necessary: the switch with the monitoring port or the hub with a free port, a maintenance laptop from the customer, a LAN tap and the analyst's sniffing device. The devices would be connected as shown in Figure 3.

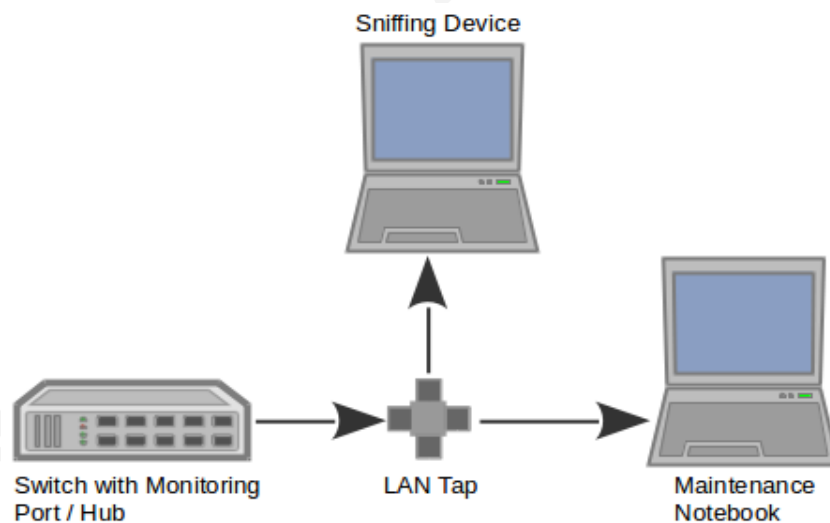


Figure 3: Physical Setup for Network Capture

As mentioned earlier, a hub will broadcast all of the traffic for the systems that are connected to it. For this type of device, no further configuration is needed. For a monitoring port on a switch, it can be more involved. First and foremost, the customer must have the means to configure the monitoring port. This means the following is necessary:

- the software/hardware required to access the switch's management interface
- the credentials to access the switch's management interface
- the expertise to configure the monitoring.

If any of these preconditions are not present in-house, then the vendor who supports the network will need to be contacted and be involved in the analysis.

Maintenance notebooks are laptops that are regularly used in the network for diagnostic and programming purposes. Most process control networks have at least one of these notebooks available. These notebooks have been connected to the network before and therefore should not anything new to the network. The maintenance notebook will be connected via the LAN tap to communicate with the switch/hub. When setting up this connection, it is important to test that there is a link established between the switch/hub and maintenance notebook before connecting the sniffing device.

The sniffing device can be a dedicated appliance or a laptop with enough hard disk space to capture data. It should be connected to the LAN tap port that forwards the data is traveling from the hub/switch to the maintenance laptop. This can be tested by connecting to a sniffing port, running either Wireshark or tcpdump and monitoring the traffic for a short period of time. If the IP address of the maintenance laptop is present in the network captures, then the sniffing device should be connected to the other sniffing port on the LAN tap.

2.3 Capturing the Network Traffic

Once the sniffing infrastructure has been set up, network traffic capture can begin. First, the network interface on the sniffing device must be configured. For the purposes of demonstration in this section and for the sake of space, the Ubuntu Linux operating system will be used for the sniffing device. This does not mean that other systems cannot be used, but they will not be addressed in this paper.

As a first step, the sniffing network interface should be set to promiscuous mode. If the system has a network manager running, it may mean that it needs to be deactivated. This can be done with the following command:

```
$ sudo service network-manager stop
```

To set up promiscuous mode, use the following command:

```
$ sudo ifconfig <interface> promisc
```

To capture network traffic, the utility tcpdump is used. This utility saves captured data in a PCAP format. Files in this format can be analyzed by a variety of network utilities and also replayed for further analysis.

Network sniffing can result in large amounts of data being captured. Within a 24 hour period, for example, 80 GB data can easily be collected. tcpdump provides command line options to break up its resulting PCAP files into fixed sized files. Here is an example of a tcpdump command:

```
$ tcpdump -i <interface> -w <base name of PCAP> -C <size of PCAP  
in MB> -s 0
```

The -i switch specifies the network interface that will sniff network traffic. This -w switch specifies the root of the PCAP files names. The extensions of the files will be numbered, such as .001, .002. -C specifies the maximum size of the PCAP files. The -s 0 switch tells tcpdump to capture whole packets.

The question then becomes, how big should the PCAP files be? This is partially dependent on the analyst's workflow. There is theoretically no limit (aside from hard disk space) on the size of a PCAP file. But, the tools that need to process the PCAP files are limited. Two useful utilities, Wireshark and Network Miner, are power tools for analysis of network data, but they tend to lag or not function at all with files that are larger than 100 MB. It is therefore suggested that the PCAPs be broken up in 100 MB increments. The command would look like:

```
$ tcpdump -i <interface> -w <base name of PCAP> -C 100 -s 0
```

Given the fact that tcpdump can generate a large amount of data, an analyst may want to attach an external hard drive and save the data there. In some cases, the error "Permission denied." is displayed. This could indicate a restriction by AppArmor ("tcpdump permission denied", 2014). To check this, run the following command:

```
$ grep tcpdump /sys/kernel/security/apparmor/profiles
```

If any result comes back, then run the following command and try tcpdump again.

```
$ sudo aa-complain /usr/sbin/tcpdump
```

Once capturing is up and running, allow it to run for at least 24 hours during a time when the process control network is at its most productive. This will provide a realistic capture of the devices on the network and hopefully evidence of the most important devices in the network. Once the data has been captured, it ready for analysis.

3. Analyzing Network Data

This section will provide a practical overview of how to analyze network traffic once it has been captured. To this end, real process control network captures which are available online will be used for demonstration. The reader has the option to download these network captures and perform the activities while reading the paper. The exercise will require a laptop or desktop computer that can support a Virtual Box virtual machine with 2 processor cores and 8 GB of RAM. During the exercise, the reader should plan to download roughly 2,5 GB of software and spend roughly 1-1.5 hours installing and configuring the software. The downloaded software will include:

- Oracle Virtual Box
- Security Onion (a virtual machine)
- Grassmarlin (along with the Oracle Java JDK)

Instructions for the installation and configuration of the software can be found in the Appendix.

Security Onion is an Ubuntu-based system that has been adapted to operate as a network intrusion detection system. It has an entire suite of network monitoring and packet analysis tools pre-installed. There are two pieces of software available in Security Onion that will be reviewed in this paper: Snort and SGUIL. Snort monitors network traffic, compares it against attack signature rules and logs events if it finds a match. SGUIL is a graphical user interface that allows an analyst to analyze the events that Snort has flagged as potential attacks or vulnerabilities.

Grassmarlin is a tool that was originally developed by the National Security Agency in the United States of America. The goal of the software is to provide a starting point for analysis of process control networks. It has the capability of digesting large amounts of network traffic. It attempts to categorize and visualize the data so that an analyst can quickly get an overview of what is in the network.

3.1 Analyzing Data: What to look for

The following types of analyses are possible with the captured data.

- Identification of assets:

The network captures will contain communication exchanges and network broadcasts from nodes in the network. From this, characteristics such as IP addresses, MAC addresses, VLANs, subnets, manufacturers and system types can be detected.

- Identification of communication protocols:

The network captures contain information about which ports are involved in communication, whether TCP or UDP is being used and the messages that are exchanged. This data can be used to derive which communication protocols are being used over the network.

- Identification of exchanged data:

At the packet level, it is possible to identify authentication credentials, exchanged files and which commands and messages are being sent between devices.

- Identification of anomalous traffic:

Based on the information gathered from the first three analysis points, an analyst has the potential to identify anomalous network activity that requires attention. For example, if a process network uses only fixed IP addresses and DHCP requests are being detected in the network captures, this suggests that further investigation is in order. In addition to this, the network packets can be analyzed for potential, known attacks.

The remainder of this chapter will provide examples of how these analysis activities can be performed.

3.2 Analyzing Data: Practical Privacy Considerations

Before getting started, the issue of data privacy must be addressed. The tools that are being introduced in this chapter allow packet-level access to information exchanged over the network. This means that the analyst will have access to the files, messages and login information that is exchanged, and this could be seen as a threat to the privacy of employees. This eventuality should be discussed with the project leader before starting the analysis. In some cases, the analysis may have to be cleared with an oversight board inside of the organization. In other cases, there may be limitations placed on the kinds of analysis that can be performed.

3.3 Getting Example Data

There are very few examples of process control network captures that are publicly available. For obvious reasons, process control network administrators do not want the secrets of their networks made available for any attacker to access and analyze. A set of process control network captures from the CS3 conference (formerly called 4SICS) is available online. CS3 is a yearly security conference in which an ICS/IoT network lab is set up for security research (“The Security Lab”, 2018). Network captures from the 2015 lab are available at: <https://www.netresec.com/?page=PCAP4SICS>. For the purposes of this paper, all three network capture files will be analyzed.

3.4 Identifying a Grounding Point for Analysis

Before starting an analysis, it is necessary to try to learn as much about the network as possible. What sort of devices should be expected in the network? What IP address ranges and IP addresses are already known? How is the network segmented, if at all? Are VLANs used? Are there any network plans that show the relationships between devices? This information can be used to help in the identification and classification of assets that belong in the network and ones that require further investigation. It can also help with a more in-depth analysis as to whether the identified devices are behaving as expected.

For the example network captures, some of this information is provided on the download page (<https://www.netresec.com/?page=PCAP4SICS>). This information includes the types of devices present and their expected IP addresses. Valid IP addresses for the network are summarized in the following list (where “x” represents either an unknown IP address or a list of IP addresses).

- 192.168.87.x
- 192.168.88.x
- 192.168.89.1
- 10.10.10.x

Once the information about the network has been gathered, analysis of the captured network traffic can effectively begin.

3.4 Analyzing with Grassmarlin

The first tool to use in analysis is Grassmarlin. Grassmarlin will provide a visual overview of the network as well as a list of all of the devices detected in the network traffic.

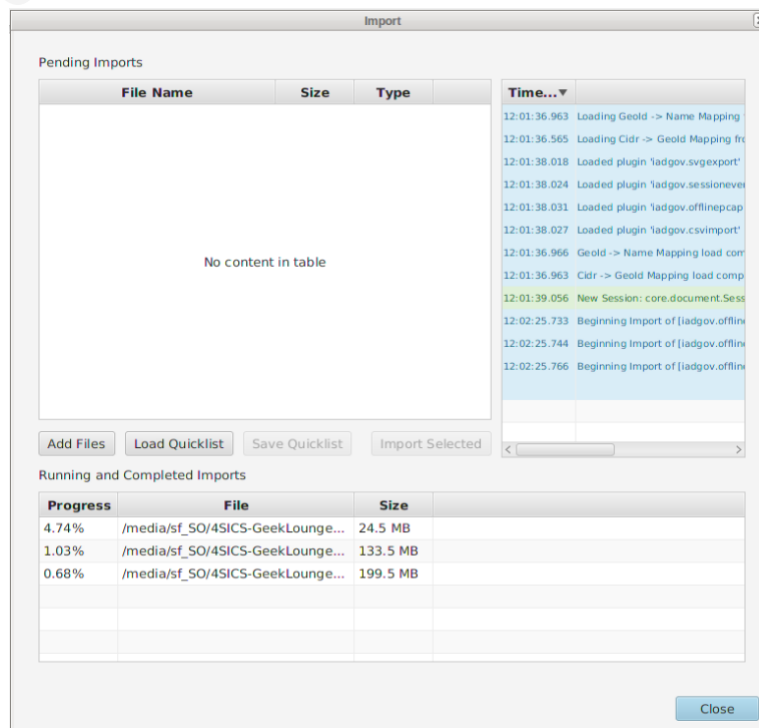


Figure 4: Grassmarlin File Import Window

To open Grassmarlin, open a terminal and issue the following command:

```
$ grassmarlin
```

This will open an empty Grassmarlin window. To import the network captures, go to File→Import Files. This will open a window where PCAP files can be imported (see Figure 4). To import network captures, click on the “Add Files” button, locate and select all of the desired PCAP files, and then click on the “Import Selected” button. Grassmarlin will then import the data and show the progress for each file in the section at the bottom left of the screen.

Grassmarlin has the capacity to import a large amount of data at one time. It buffers the data in such a way so that the user can specify gigabytes of data for import. For example, 24 hour captures of network traffic with roughly 40 GB of traffic could be loaded in about 20 minutes.

Once the traffic has been loaded into Grassmarlin, it will look something like that in Figure 5.

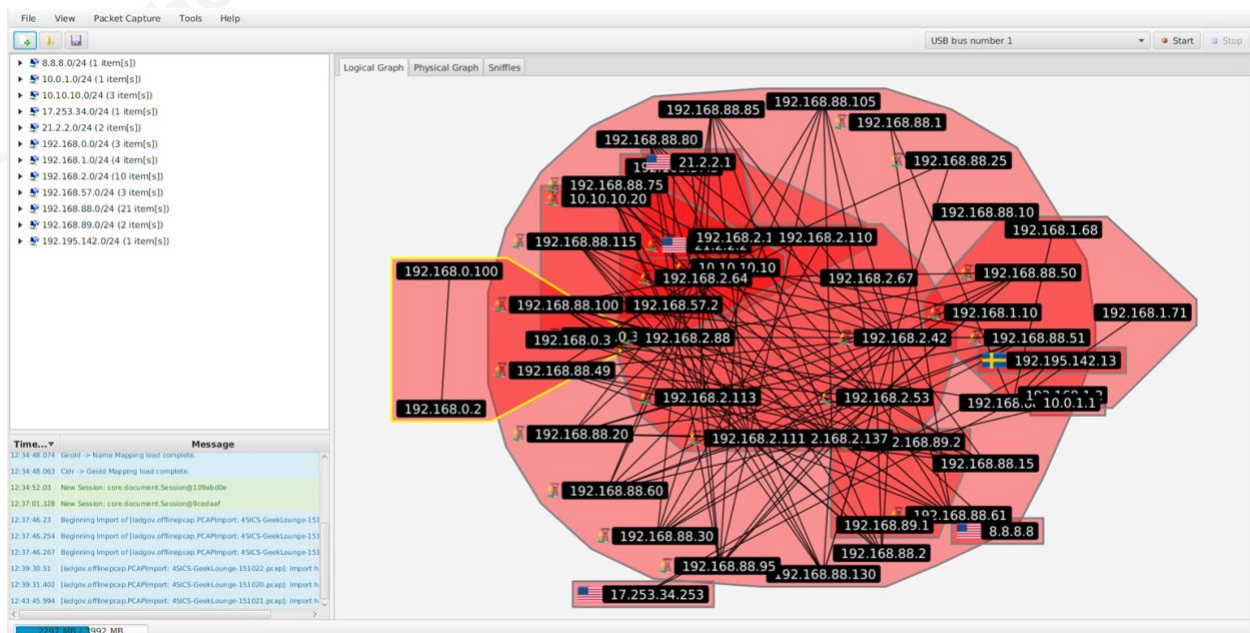


Figure 5: Initial View of Loaded Data in Grassmarlin

The left-hand panel shows the subnets where IP addresses were detected in the network traffic.

The right-hand panel shows a “logical graph” of the suspected devices in the network. The logical graph attempts to group devices by network subnet and show which devices communicate with other devices in the network.

Already the view in the left-panel provides useful information (see Figure 6).



Figure 6: Subnets Detected in Network Captures

When comparing the IP address ranges in this network to the expected IP address ranges, the following IP address ranges are anomalous.

- 8.8.8.0/24
- 10.0.1.0/24
- 17.253.34.0/24
- 192.168.0.0/24
- 192.168.1.0/24
- 192.168.2.0/24
- 192.168.57.0/24
- 192.195.142.0/24

The first anomaly to note is that there are IP addresses that fall outside of the IANA-defined private IP address ranges. These private IP address ranges are:

- 10.0.0.0 – 10.255.255.255
- 172.16.0.0 – 172.31.255.255

- 192.168.0.0 – 192.168.255.255

Process control networks are often “air-gapped” which means that they do not communicate with external networks. The presence of these IP addresses in the list indicates that, at the minimum, some device or devices in the network were attempting to communicate with external IP addresses. This could indicate a misconfiguration of a valid device or the presence of a rogue device that does not belong on the network.

Take, for example, the first anomaly in the list, 8.8.8.0/24. It is possible to expand the listing in Grassmarlin and see the details of the IP range communications as in Figure 7.

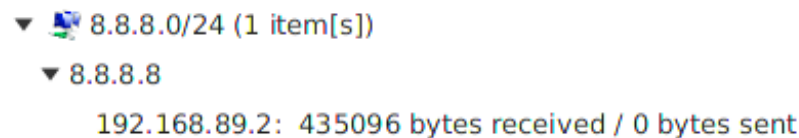
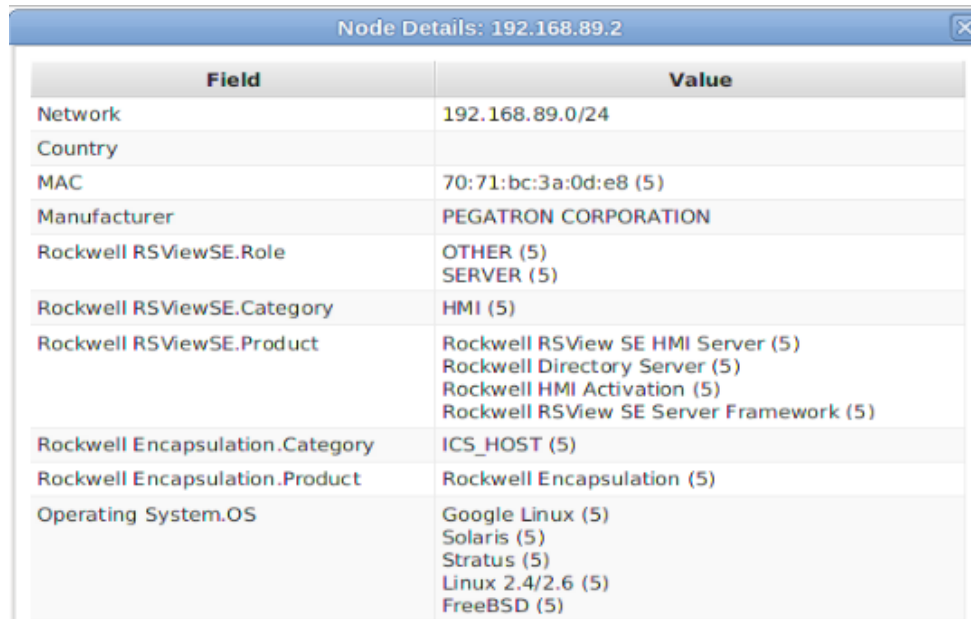


Figure 7: Detail of 8.8.8.0/24

When drilling down into the network, it can be seen that a device with the IP address of 192.168.89.2 attempted to contact the IP address 8.8.8.8 and that 8.8.8.8 did not respond. 8.8.8.8 is a well-known IP address and points to one of Google’s public DNS servers. Since there was no data returned from 8.8.8.8, it is likely that the network did not allow the DNS request to go through to the external network. This is a good sign that the network, indeed, isolates the devices from the external, public network.

So, what then, is the 192.168.89.2 device? This information can be quickly identified. By drilling down into the 192.168.89.0/24 subnet, it is possible to right click on the IP address and select “View Details for 192.168.89.2”. This will open a window that contains the following kinds of information:



Field	Value
Network	192.168.89.0/24
Country	
MAC	70:71:bc:3a:0d:e8 (5)
Manufacturer	PEGATRON CORPORATION
Rockwell RSViewSE.Role	OTHER (5) SERVER (5)
Rockwell RSViewSE.Category	HMI (5)
Rockwell RSViewSE.Product	Rockwell RSView SE HMI Server (5) Rockwell Directory Server (5) Rockwell HMI Activation (5) Rockwell RSView SE Server Framework (5)
Rockwell Encapsulation.Category	ICS_HOST (5)
Rockwell Encapsulation.Product	Rockwell Encapsulation (5)
Operating System.OS	Google Linux (5) Solaris (5) Stratus (5) Linux 2.4/2.6 (5) FreeBSD (5)

Figure 8: Detail of 192.168.89.2

Here, Grassmarlin shows either information that was directly taken from packet captures or derived from the data in the packet captures. The only reasonably reliable²³ information in the table of the current version of Grassmarlin is the MAC address and the manufacturer. This information can be brought to the network administrator for further analysis and identification of the device.

The second anomaly to note is that there are private IP address ranges discovered that were not identified in our expected IP address ranges. As with the external IP address ranges, it is possible to drill down and to see a summary of the network traffic for these ranges. Take for example, activity in the unexpected IP range 192.168.2.0/24 as seen in Figure 9.

². The manufacturer is derived from the first 3 octets of the MAC address. It is possible for an attacker to spoof a MAC address and masquerade as a device from a ICS manufacturer.

³ Grassmarlin attempts to derive information from network captures rather than from active scans of a device. This can result in the software making false conclusions about the types of devices in the network.

▼ 192.168.2.0/24 (10 item[s])

▼ 192.168.2.42

- 192.168.2.67: 42 bytes received / 0 bytes sent
- 192.168.88.2: 0 bytes received / 0 bytes sent
- 192.168.88.15: 60 bytes received / 0 bytes sent
- 192.168.88.20: 238434 bytes received / 35306 bytes sent
- 192.168.88.25: 369 bytes received / 4686 bytes sent
- 192.168.88.30: 1091 bytes received / 3908 bytes sent
- 192.168.88.49: 60 bytes received / 0 bytes sent
- 192.168.88.50: 78 bytes received / 22 bytes sent
- 192.168.88.51: 2992 bytes received / 786 bytes sent
- 192.168.88.60: 216206 bytes received / 38961 bytes sent
- 192.168.88.61: 272820 bytes received / 50375 bytes sent
- 192.168.88.75: 60 bytes received / 0 bytes sent
- 192.168.88.80: 0 bytes received / 0 bytes sent
- 192.168.88.85: 0 bytes received / 0 bytes sent
- 192.168.88.95: 60 bytes received / 11 bytes sent
- 192.168.88.100: 7590370 bytes received / 1933755 bytes sent
- 192.168.88.105: 0 bytes received / 512 bytes sent
- 192.168.88.115: 330487 bytes received / 57350 bytes sent
- 192.168.88.130: 0 bytes received / 0 bytes sent

Figure 9: Activity from an Unidentified Devices in the Network

This activity should be flagged as suspicious because of the number of devices with which the node attempts to communicate. Usually, only centralized control systems attempt to directly communicate with so many devices. And, it is usually the case that the network administrators know the IP addresses of their control systems by heart. The fact that this node is not in the original list of valid nodes and that it is so communicative is alarming. This device is either misconfigured, or it is being used to perform reconnaissance in the network. Drilling down to look at the manufacturer provides the information in Figure 10.

Node Details: 192.168.2.42	
Field	Value
Network	192.168.2.0/24
Country	
MAC	68:f7:28:3f:67:5c (5) 00:07:7c:1a:61:83 (5)
Manufacturer	LCFC(HeFei) Electronics Technology co., ltd Westermo Teleindustri AB

Figure 10: Manufacturer Info of 192.168.2.42

Interestingly, this IP address had two MAC addresses during the time of the network captures. This indicates that either two devices in the network had the same IP address, or there was one device where the MAC address was modified. This should be a red flag to an analyst and would require further investigation.

Another anomalous category of interesting IP addresses to look out for are IP addresses that begin with 255.x.x.x. These are broadcast IP addresses. Often, they indicate that a device on the network is attempting to be assigned an IP address via DHCP. In process control networks, devices usually have fixed IP addresses, so this should be a red flag for an analyst.

The final category of anomalous IP addresses that should raise red flags are addresses that start with 169.254.x.x. These are default IP addresses assigned in Windows systems when there is no fixed IP assigned and no DHCP server available to assign an IP address. If there are addresses like this present in network captures, this means one of two things. This could be a valid, but misconfigured device, or it could be the device of an attacker. In both cases, follow up is necessary.

The left-panel is useful for drilling down into subnets and individual devices, but it fails to provide a high-level overview of the communication between devices. The right-hand panel with the logical graph can help to resolve this. As can be seen in Figure 5, the right-panel does not initially provide a helpful overview of the network communication. To make this a useful visual aid, the analyst has the ability to clean up the picture by zooming in and out and dragging the nodes and groups of nodes around until a more sensible representation can be achieved as in Figure 11.

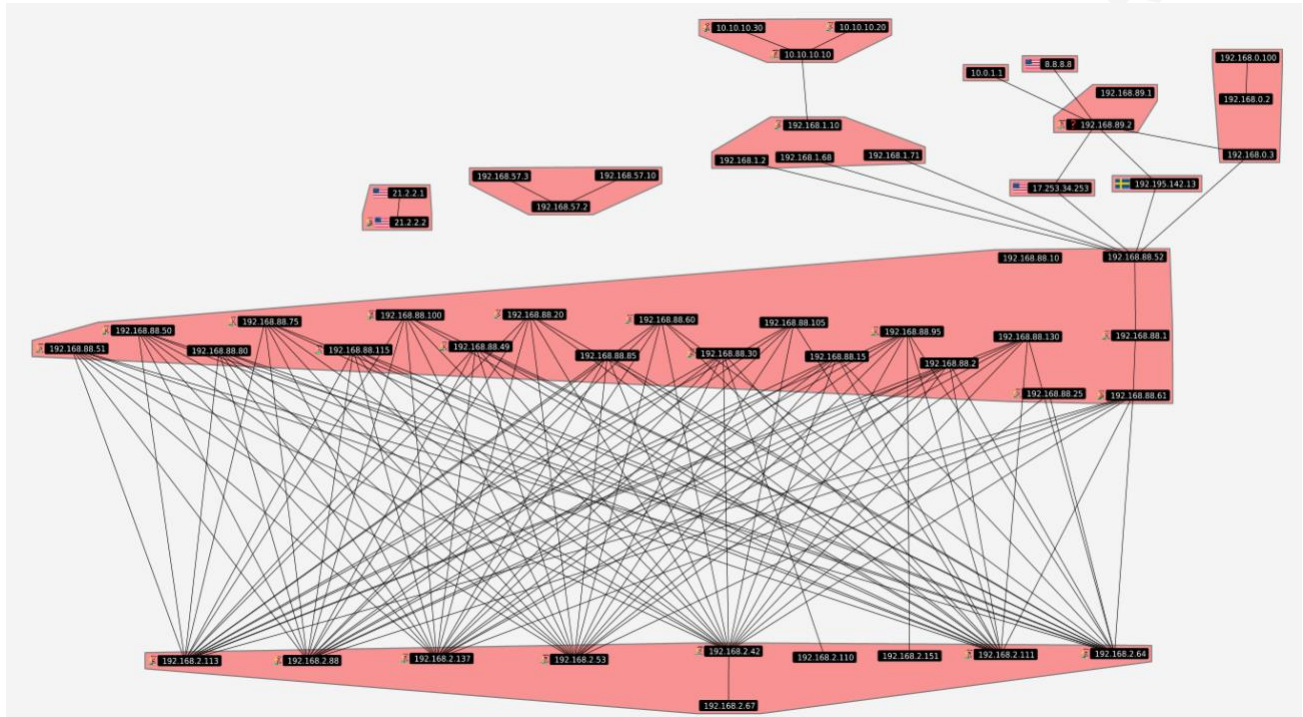
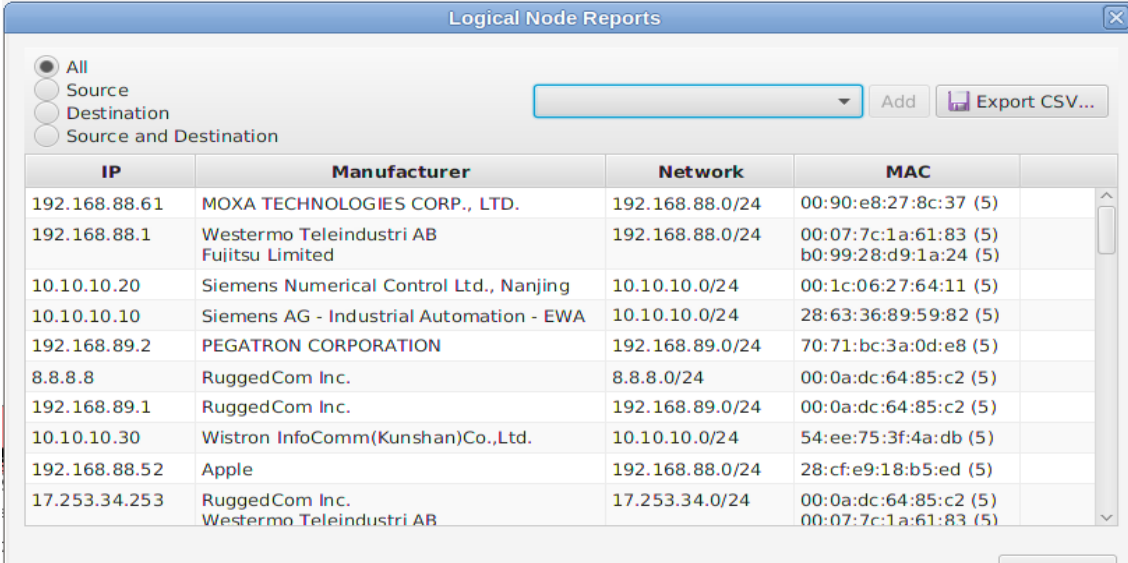


Figure 11: Rearranged Logical Graph

This can help the analyst get a better feel of how the network is organized by illustrating which nodes in subnets communicate with others. This can also be used as a visual talking point with network administrators to identify communication between nodes that requires further analysis.

For inventorying purposes, it is possible to generate a list with the IP addresses, networks, MAC addresses and manufacturers of the devices identified in the network captures. This can be done by going to View -> Logical Nodes Report. This will bring up a window like in Figure 12.



The screenshot shows a window titled "Logical Node Reports". At the top, there are radio buttons for "All", "Source", "Destination", and "Source and Destination". To the right of these is a dropdown menu and buttons for "Add" and "Export CSV...". Below this is a table with the following data:

IP	Manufacturer	Network	MAC
192.168.88.61	MOXA TECHNOLOGIES CORP., LTD.	192.168.88.0/24	00:90:e8:27:8c:37 (5)
192.168.88.1	Westermo Teleindustri AB	192.168.88.0/24	00:07:7c:1a:61:83 (5)
	Fujitsu Limited		b0:99:28:d9:1a:24 (5)
10.10.10.20	Siemens Numerical Control Ltd., Nanjing	10.10.10.0/24	00:1c:06:27:64:11 (5)
10.10.10.10	Siemens AG - Industrial Automation - EWA	10.10.10.0/24	28:63:36:89:59:82 (5)
192.168.89.2	PEGATRON CORPORATION	192.168.89.0/24	70:71:bc:3a:0d:e8 (5)
8.8.8.8	RuggedCom Inc.	8.8.8.0/24	00:0a:dc:64:85:c2 (5)
192.168.89.1	RuggedCom Inc.	192.168.89.0/24	00:0a:dc:64:85:c2 (5)
10.10.10.30	Wistron InfoComm(Kunshan)Co.,Ltd.	10.10.10.0/24	54:ee:75:3f:4a:db (5)
192.168.88.52	Apple	192.168.88.0/24	28:cf:e9:18:b5:ed (5)
17.253.34.253	RuggedCom Inc.	17.253.34.0/24	00:0a:dc:64:85:c2 (5)
	Westermo Teleindustri AB		00:07:7c:1a:61:83 (5)

Figure 12: Logical Node Report Configuration Window

This window starts with only a column that lists the IP addresses of the devices. Additional columns can be added, and then they can be exported as a comma-delimited list which can be opened in programs such as Excel. This list can be used for comparison with an existing inventory list. Or, it can be used as the basis for a new inventory list.

3.5 Analyzing with Snort and SGUIL

As mentioned earlier, Snort is a network monitoring software that analyzes the content of network packets for anomalous activity. SGUIL is a user interface that enables the analysis of the events that Snort reports. In order for Snort to be able to analyze the traffic, the traffic will need to be replayed in the Security Onion environment. After this, SGUIL can be used to analyze the data.

3.4.1 Replaying Network Captures for Snort

The first step to replaying network captures so that Snort can analyze them is to make sure that the required Security Onion services are running. To check the services, open a terminal and issue the following command:

```
$ sudo sostat | less
```

This will provide a screen with the status of all of the services like in Figure 13.

```
=====
Service Status
=====
Status: securityonion
* sgul server[ OK ]
Status: HIDS
* ossec_agent (sguil)[ FAIL ]
Status: Bro
Name      Type      Host      Status  Pid  Started
manager   manager  localhost stopped
proxy     proxy    localhost stopped
so-eth1-1 worker   localhost stopped
Status: so-eth1
* netsniff-ng (full packet data)[ FAIL ]
* pcap_agent (sguil)[ FAIL ]
* snort_agent-1 (sguil)[ FAIL ]
* snort-1 (alert data)[ FAIL ]
* barnyard2-1 (spooler, unified2 format)[ FAIL ]
=====
Interface Status
=====
docker0   Link encap:Ethernet  HWaddr 02:42:0c:b0:7d:6e
: 
```

Figure 13: sostat

If anything in this screen indicates that it is stopped or has failed to start, then quit from sostat (q) and then run the following command:

```
$ sudo service nsm restart
```

This will restart the Security Onion services, and the environment will be ready to accept data.

If the Snort rules have not been updated in a while, the following command will ensure that the rules are up-to-date.

```
$ sudo rule-update
```

To replay the network captures, use the terminal and change directory to where the PCAP files are located. They can be loaded with the following command.

```
$ tcpreplay -i eth1 -t *.pcap
```

In this command, the `-i` switch indicates which network interface to target (`eth1`), `-t` says to replay at the highest speed possible, and `*.pcap` says to replay all of the files in the directory with the extension `.pcap`.

When replaying the files, errors like the ones in Figure 14 may arise.

```
souser@so-new:/media/sf_S0$ sudo tcpreplay -i eth1 -t *.pcap
sending out eth1
processing file: 4SICS-GeekLounge-151020.pcap
processing file: 4SICS-GeekLounge-151021.pcap
Warning in send_packets.c:send_packets() line 178:
Unable to send packet: Error with PF_PACKET send() [727743]: Message too long (errno = 90)
Warning in send_packets.c:send_packets() line 178:
Unable to send packet: Error with PF_PACKET send() [926009]: Message too long (errno = 90)
Warning in send_packets.c:send_packets() line 178:
Unable to send packet: Error with PF_PACKET send() [935892]: Message too long (errno = 90)
```

Figure 14: Error Messages with Large Packets

If this error occurs, it means that the MTU on the interface is too small to handle the size of the packets that are being replayed. To modify the size of the MTU, use the following command:

```
$ sudo ifconfig eth1 mtu 3000 up
```

Then rerun the `tcpreplay` command again. If the errors still occur, try changing the MTU to 9000. This is the largest MTU possible.

It should take a relatively short period of time to replay the example PCAP files in this example. This is due to the relatively small file sizes (~375 MB). In a real-world analysis, there will most likely be much larger amounts of data. For estimating purposes, around 5 GB of data can take 10-15 minutes to replay.

3.4.2 Analyzing with SGUIL

To open SGUIL, double-click on the icon on the desktop and enter the login and password specified when the system was installed and configured. After this, a dialogue window will be presented inquiring about which interface should be analyzed.

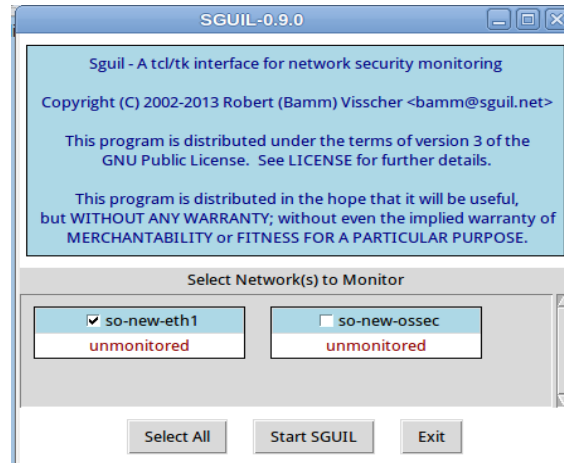


Figure 15: SGUIL: Select Network for Analysis

After selecting the eth1 network, click on the Start SGUIL button, and the main analysis screen will be displayed.

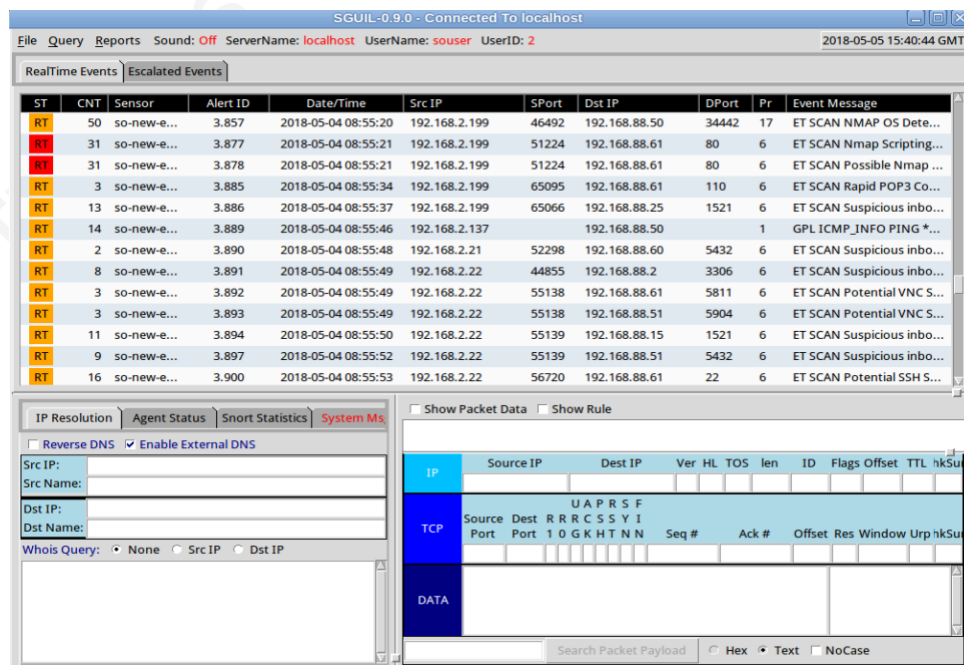
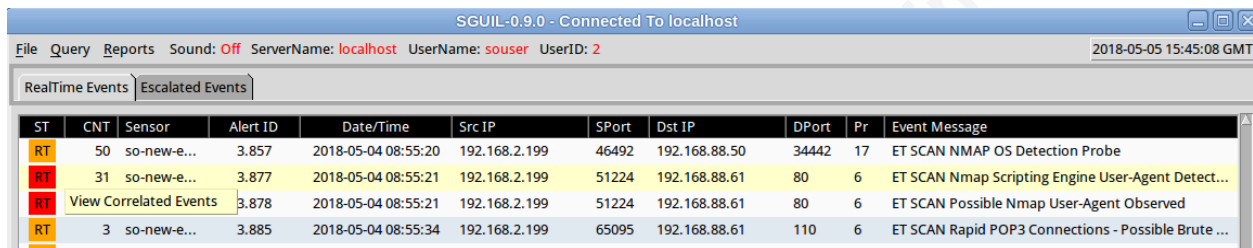


Figure 16: SGUIL Main Analysis Window

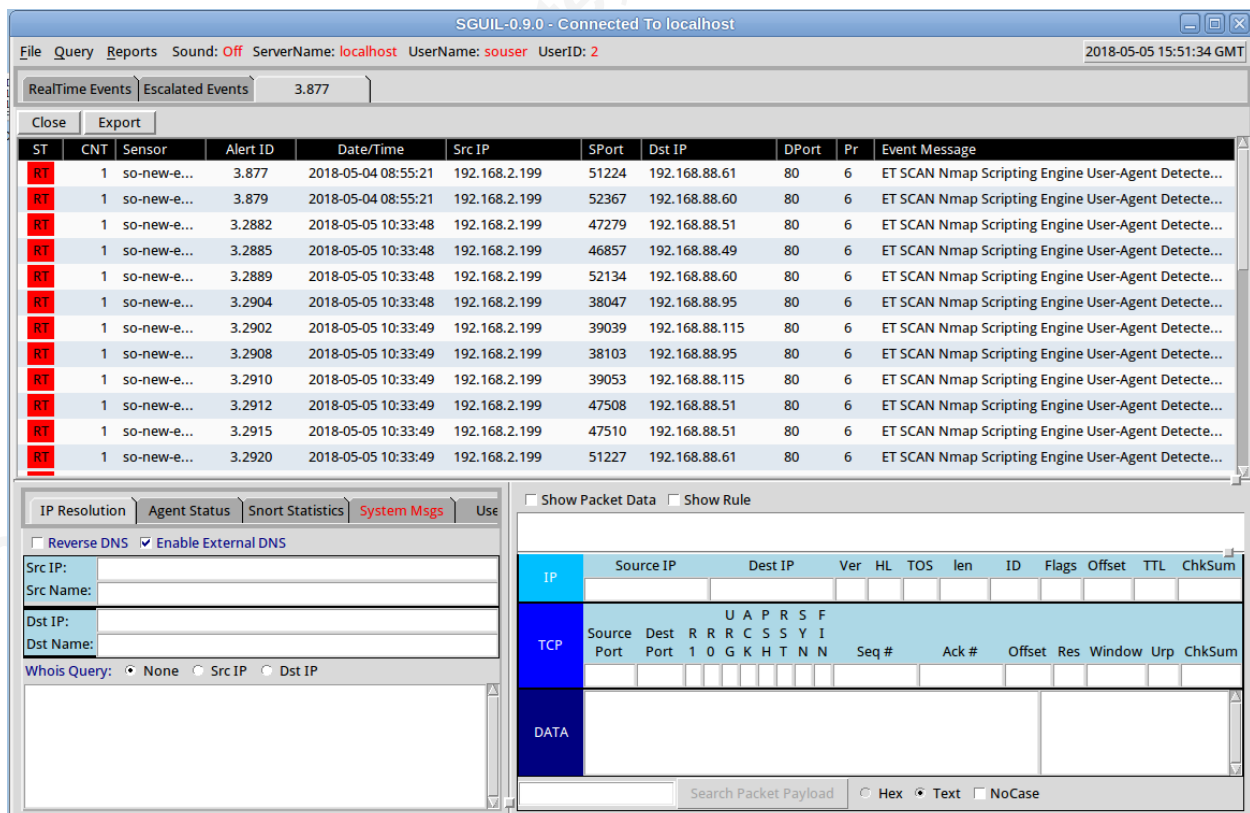
This screen is an aggregate screen that shows all of the events flagged by Snort along with the number of times that the event occurred. The number of times is specified in the CNT column. The analyst should review the events for any that require further investigation. To drill

down into an interesting event, right click on the CNT value of the record and click “View Correlated Events” (see Figure 17).



ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	50	so-new-e...	3.857	2018-05-04 08:55:20	192.168.2.199	46492	192.168.88.50	34442	17	ET SCAN Nmap OS Detection Probe
RT	31	so-new-e...	3.877	2018-05-04 08:55:21	192.168.2.199	51224	192.168.88.61	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	3	so-new-e...	3.878	2018-05-04 08:55:21	192.168.2.199	51224	192.168.88.61	80	6	ET SCAN Possible Nmap User-Agent Observed
RT	3	so-new-e...	3.885	2018-05-04 08:55:34	192.168.2.199	65095	192.168.88.61	110	6	ET SCAN Rapid POP3 Connections - Possible Brute ...

Figure 17: Drill Down to Individual Events



ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	1	so-new-e...	3.877	2018-05-04 08:55:21	192.168.2.199	51224	192.168.88.61	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.879	2018-05-04 08:55:21	192.168.2.199	52367	192.168.88.60	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2882	2018-05-05 10:33:48	192.168.2.199	47279	192.168.88.51	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2885	2018-05-05 10:33:48	192.168.2.199	46857	192.168.88.49	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2889	2018-05-05 10:33:48	192.168.2.199	52134	192.168.88.60	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2904	2018-05-05 10:33:48	192.168.2.199	38047	192.168.88.95	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2902	2018-05-05 10:33:49	192.168.2.199	39039	192.168.88.115	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2908	2018-05-05 10:33:49	192.168.2.199	38103	192.168.88.95	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2910	2018-05-05 10:33:49	192.168.2.199	39053	192.168.88.115	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2912	2018-05-05 10:33:49	192.168.2.199	47508	192.168.88.51	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2915	2018-05-05 10:33:49	192.168.2.199	47510	192.168.88.51	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2920	2018-05-05 10:33:49	192.168.2.199	51227	192.168.88.61	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...

Figure 18: View of Individual Events

In this screen it is possible to see that one IP address, namely 192.168.2.199, is attempting to connect to multiple IP addresses on port 80. The alert gives the indication that this is an Nmap scan.

By highlighting one of the records and activating the “Show Packet Data” checkbox, it is possible to inspect the actual packet that was sent that triggered this alert like if Figure 19.

The screenshot shows the SGUIL-0.9.0 interface. The top bar indicates the application is connected to localhost. Below the menu bar, there are tabs for 'RealTime Events' and 'Escalated Events'. A table of alerts is displayed, with columns for ST, CNT, Sensor, Alert ID, Date/Time, Src IP, SPort, Dst IP, DPort, Pr, and Event Message. One alert is selected, and its details are shown in the bottom pane. The 'Show Packet Data' checkbox is checked, and the packet details are displayed in a structured format.

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	1	so-new-e...	3.877	2018-05-04 08:55:21	192.168.2.199	51224	192.168.88.61	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.879	2018-05-04 08:55:21	192.168.2.199	52367	192.168.88.60	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2882	2018-05-05 10:33:48	192.168.2.199	47279	192.168.88.51	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2885	2018-05-05 10:33:48	192.168.2.199	46857	192.168.88.49	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2889	2018-05-05 10:33:48	192.168.2.199	52134	192.168.88.60	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2904	2018-05-05 10:33:48	192.168.2.199	38047	192.168.88.95	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2902	2018-05-05 10:33:49	192.168.2.199	39039	192.168.88.115	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2908	2018-05-05 10:33:49	192.168.2.199	38103	192.168.88.95	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...
RT	1	so-new-e...	3.2910	2018-05-05 10:33:49	192.168.2.199	39053	192.168.88.115	80	6	ET SCAN Nmap Scripting Engine User-Agent Detecte...

IP	Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags	Offset	TTL	ChkSum
IP	192.168.2.199	192.168.88.61	4	5	0	264	38840	2	0	62	51170

TCP	Source Port	Dest Port	R R R R	C C C C	S S S S	S S S S	Y Y Y Y	I I I I	Seq #	Ack #	Offset	Res	Window	Urp	ChkSum	
TCP	51224	80	X	X	
TCP	4F 50 54 49 4F 4E	53 20 2F 20 48 54 54 50 2F 31	2E 31 0D 0A 48 6F	73 74 3A 20 31 39 32 2E 31 36	38 2E 38 38 2E 36	31 0D 0A 41 63 63 65 73 73 2D	43 6F 6E 74 72 6F	6C 2D 52 65 71 75 65 73 74 2D	4D 65 74 68 6F 64	3A 20 48 45 41 44 0D 0A 43 6F	6E 6E 65 63 74 69	6F 6E 3A 20 63 6C 6F 73 65 0D	0A 55 73 65 72 2D	41 67 65 6E 74 3A 20 4D 6F 7A	69 6C 6C 61 2F 35	2E 30 20 28 63 6F 6D 70 61 74
TCP	69 6C 6C 65 3B 20	4E 6D 61 70 20 53 63 72 69 70	74 69 6E 67 20 45	6E 67 69 6E 65 3B 20 68 74 74	70 3A 2F 2F 6E 6D	61 70 2E 6F 72 67 2F 62 6F 6F	6B 2F 6E 73 65 2E	68 74 6D 6C 29 0D 0A 4F 72 69	67 69 6E 3A 20 65	78 61 6D 70 6C 65 2E 63 6F 6D	0D 0A 0D 0A					

DATA: OPTIONS / HTTP/1.1..Host: 192.168.88.61..Access-Control-Request-Method: HEAD..Connection: close..User-Agent: Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)..Origin: example.com....

Figure 19: Inspecting the Contents of a Suspicious Packet

If desired, the packet can be loaded into other tools such as Wireshark or Network Miner for further inspection. To do this, right-click on the Alert ID of the desired record, and select the desired target analysis tool. As seen in the example above, an actor in the network was performing an Nmap scan. This is highly suspicious for an ICS network, and it should be reported and investigated immediately.

In real process networks, it is unlikely that SGUIL will show any alerts that indicate an attack has taken place. If the network is properly air-gapped, and the operators are careful about which maintenance laptops are connected to the network, the most common event that will be listed is the “SNMP public access UDP” alert. This simply means that SNMP is being used to

monitor devices in the network and the the community name “public” is being used. This is a security weakness, but it is not the signal of an attack. Other likely events may end up correlating to misconfigured devices. Questionable events should be noted and discussed with the network administrator.

3.6 Analyzing the Data: Other Useful Tools

Security Onion offers a wide variety of analysis tools. Two further tools worth highlighting are Network Miner and Kibana.

Network Miner allows the analyst to carve out files from network traffic, identify login credentials that have been sent over the network and get further information about the communication protocols on the network.

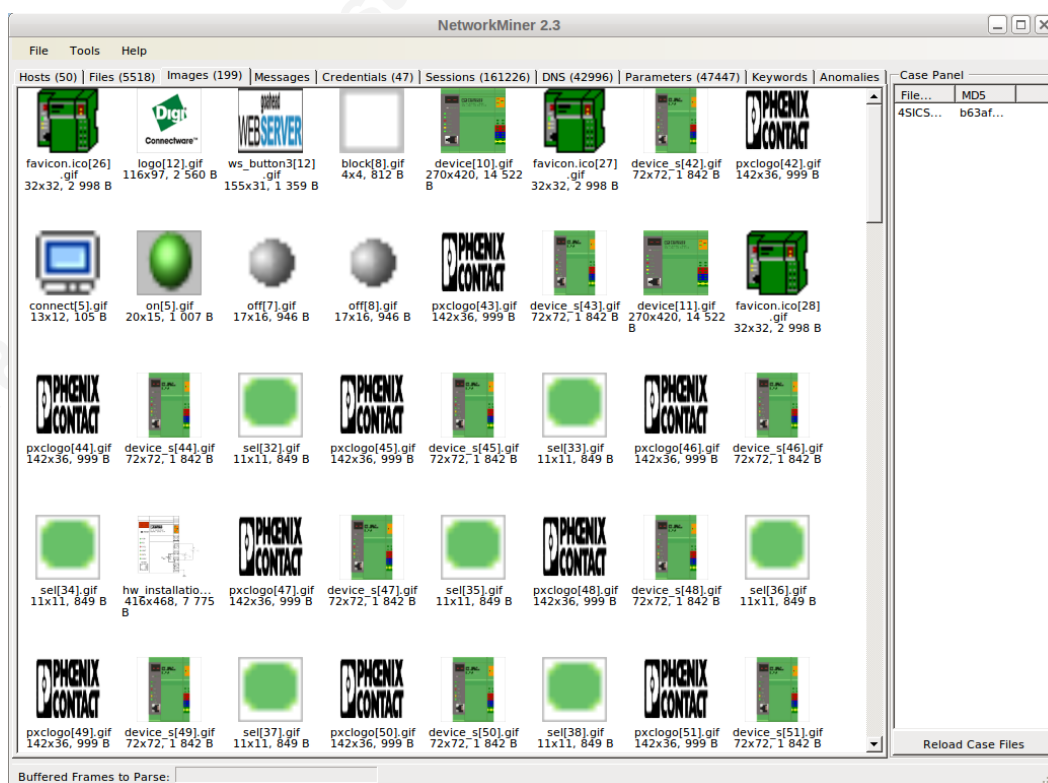


Figure 20: Captured Images Rendered in NetworkMiner

The biggest drawback with Network Miner is that it cannot handle large amounts of data. This means that it can only handle PCAP files of less than 200 MB.

Kibana is relatively new to Security Onion. When replaying traffic in Security Onion, the data is processed, categorized and stored in Bro logs. Kibana takes the data in the Bro logs and provides a web-based, graphical interface to view the data. For example, it provides tables and visual representations of which protocols were detected in the network traffic, which devices sent the most data, which devices received the most data, etc. Kibana replaces a tried-and-true tool called ELSA. Over time as Kibana matures, it will become a powerful tool for analyzing, querying, categorizing and visualizing network traffic.

4. Conclusion: ICS ISMS Next Steps

While the tools and approach presented in this paper can already provide a great deal of information for analyzing process control networks, there is still a great deal of room for improvement and improvement seems to be continuous. New versions of Grassmarlin are released every 6 months to one year. Security Onion has matured rapidly in just the last year. And, new products are being introduced to the market to help make these types of analyses easier. A security analyst needs to remain aware of the changing market and pick the tools that are fit for the job at hand.

As demonstrated in this paper, the use of passive network analysis makes it possible to gather a great deal of information about assets in an ICS network. It even enables the possibility of identifying misconfigurations and potential attackers. This passive approach, though, does have its limits. Subnets where packets were not directly captured may be hidden behind the IP of a router or gateway. Devices that remained dormant on the network during the period when the network traffic was captured will not be registered. When doing this analysis, it is important to keep these factors in mind. A passive network analysis will likely not provide an enumeration of the entire network, but it can serve as a good start.

References

- Allen, J. H. (2013, July 2). Plan, Do, Check, Act. Retrieved April 16, 2018, from <https://www.us-cert.gov/bsi/articles/best-practices/deployment-and-operations/plan-do-check-act#commitment>
- Chason, G., Dinnage, S., Lee, A., Searle, J., Widger, D., & Wright, A. (2014). *National Electric Sector Cybersecurity Organization Resource (NESCOR)* (No. 1163840). <https://doi.org/10.2172/1163840>
- Gesetz über die Elektrizitäts- und Gasversorgung (Energiewirtschaftsgesetz - EnWG) § 11 Betrieb von Energieversorgungsnetzen. (2015). Retrieved April 16, 2018, from http://www.gesetze-im-internet.de/enwg_2005/__11.html
- GRASSMARLIN: Provides situational awareness of Industrial Control Systems (ICS) and Supervisory Control and Data Acquisition (SCADA) networks in support of network security assessments. iadgov. (2018). (Version 3.2.1). NSA Cybersecurity. Retrieved from <https://github.com/iadgov/GRASSMARLIN> (Original work published 2015)
- ICS Lab. (2018). Retrieved April 30, 2018, from <https://cs3sthlm.se/ics-lab/>
- Lee, R. M., Assante, M. J., & Conway, T. (2016, March 18). Analysis of the Cyber Attack on the Ukrainian Power Grid. E-ISAC. Retrieved from https://ics.sans.org/media/E-ISAC_SANS_Ukraine_DUC_5.pdf
- march. (2018, May 5). Java 8 › Oracle Java › Installation › Java. Retrieved May 5, 2018, from https://wiki.ubuntuusers.de/Java/Installation/Oracle_Java/Java_8/
- Peterson, D. (2016, July 18). ICS Security Assessment Methodology, Tools & Tips. Retrieved April 17, 2018, from <https://www.youtube.com/watch?v=0WoA9SYLDoM&t=2525s>
- SCADA/ ICS PCAP files from 4SICS. (2018, January 23). Retrieved May 5, 2018, from <https://www.netresec.com/?page=PCAP4SICS>
- security-onion: Linux distro for IDS, NSM, and Log Management. (2018). (Version 14.04.5.13). Security Onion Solutions, LLC. Retrieved from <https://github.com/Security-Onion-Solutions/security-onion> (Original work published 2015)
- Stouffer, K., Pillitteri, V., Lightman, S., Abrams, M., & Hahn, A. (2015). *Guide to Industrial Control Systems (ICS) Security* (No. NIST SP 800-82r2). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-82r2>

tcpdump permission denied. (2010, April 6). Retrieved May 5, 2018, from

<https://ubuntuforums.org/showthread.php?t=1501339>

The 60 Minute Network Security Guide (First Steps Towards a Secure Network Environment).

(2006, May 16). National Security Agency of the United States of America. Retrieved from <http://www.cs.unibo.it/babaoglu/courses/security/resources/documents/NSA-SNAC.pdf>

Appendix

Setting Up the Analysis Software

To analyze network traffic, a combination of tools will be used: Security Onion, Grassmarlin, and SGUIL. Security Onion is an Ubuntu installation that has been specifically configured to be an intrusion detection system. Grassmarlin, originally developed by the NSA, focuses specifically on ICS network analysis. SGUIL will provide a starting point for the analysis of suspicious network activity.

Installing Security Onion

Security Onion can be downloaded as an ISO. Instructions for the download and verification of the software can be found under: https://github.com/Security-Onion-Solutions/security-onion/blob/master/Verify_ISO.md. The version of Security Onion used in this tutorial is the 14.04.5.13 ISO image built on 2018/04/25.

The first step to set up Security Onion after download and verification is to configure a host network in Virtual Box if there is not one present. This will be later used to set up the network interface for the replay of network traffic in Security Onion. To do this, go to **File → Host Network Manager**. If a host network does not exist, add a new one.

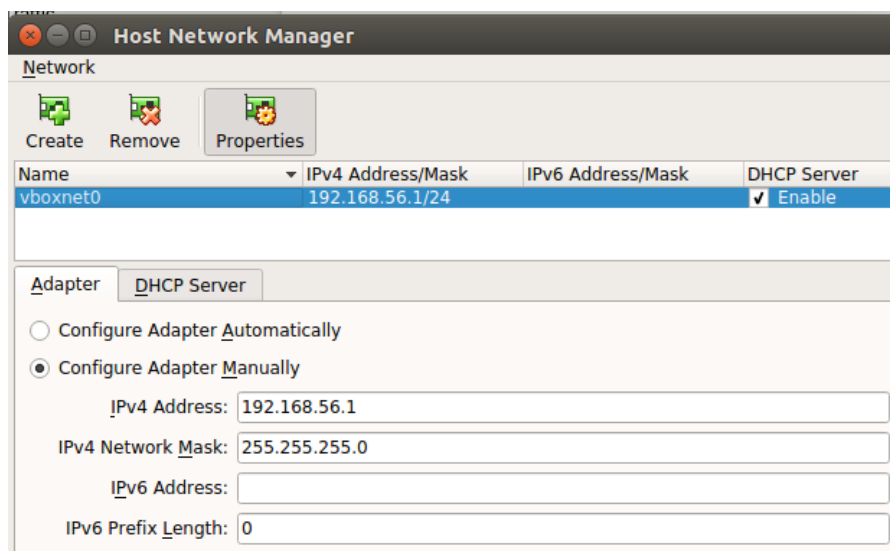


Figure 21: Virtual Box Host Network Manager

After this, set up a new 64-bit Ubuntu, Linux virtual machine with a 30 GB dynamically allocated VDI disk. After this is set up, modify the following settings for the VM⁴:

- Processor: minimum 2
- RAM: minimum 8 GB
- Network:
 - Adapter 1: NAT (This interface will be used to download updates to software.)
 - Adapter 2: Host Only Network, Advanced -> Promiscuous Mode → Allow (See Figure 22.)

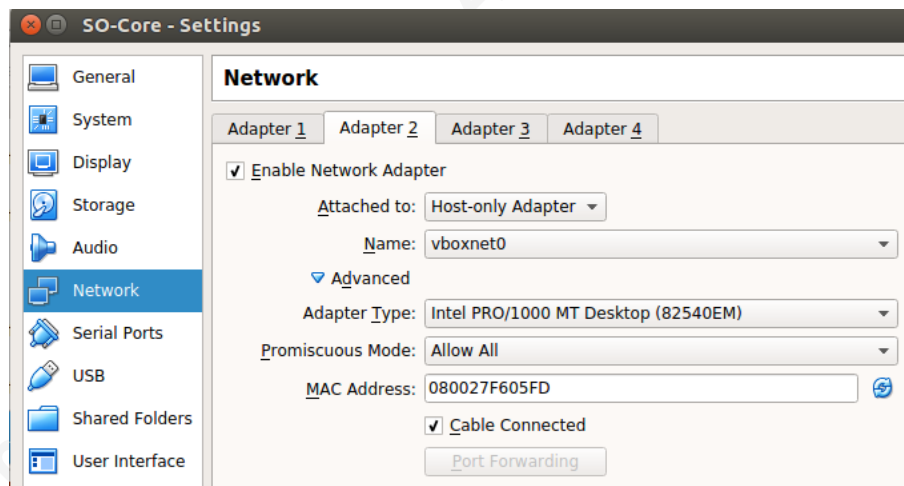


Figure 22: Host-only Adapter configuration

Once these configurations are complete, connect the virtual machine to the ISO and begin installation. Follow the prompts to install the operating system. There are three points that require some clarification.

The first point of clarification can be found at the “Preparing to install Security Onion” screen (see Figure 23).

⁴The minimum processor and RAM requirements must be met or parts of Security Onion will not function properly due to lack of sufficient system resources.

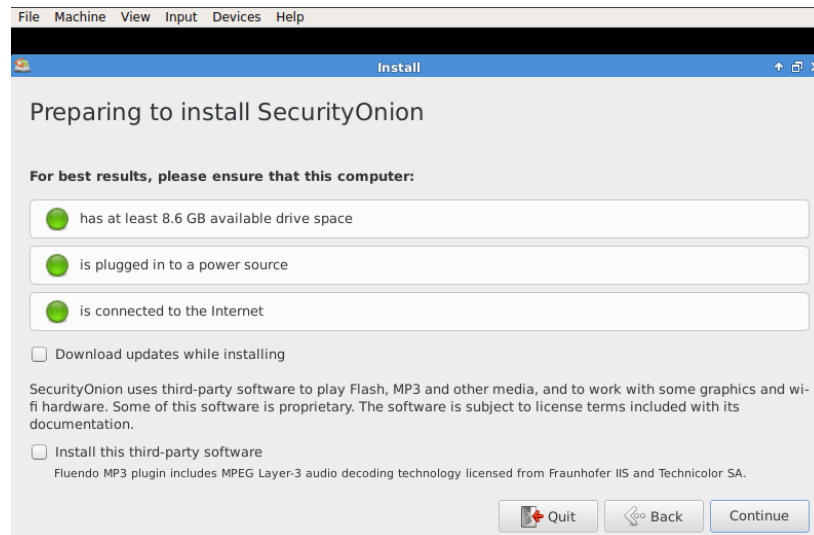


Figure 23: Preparing to install Security Onion

The regular security analyst would be tempted to activate the “Download updates while installing” checkbox. **Do not activate this option.** Security Onion provides its own process for updating the operating system and its software. This process will be run after Security Onion has been installed.

The next point for clarification can be found on the next “Installation Type” screen (see Figure 24).

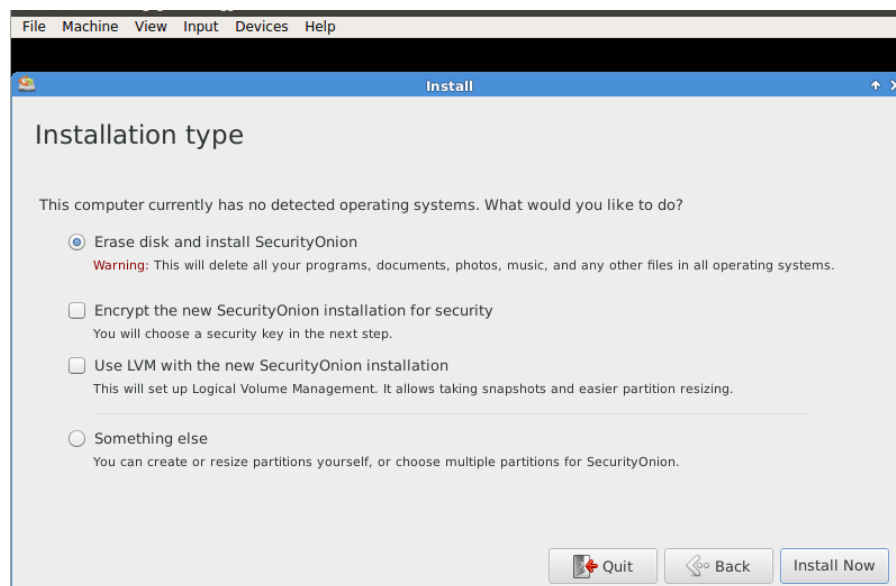


Figure 24: Installation type screen

A security analyst may be tempted to activate the “Encrypt the Security Installation for security” option. **Do not activate this option.** Accept the defaults for this screen and proceed.

Another challenge during installation is that the size of the installation screen is too small to contain some of the screens (see Figure 25).

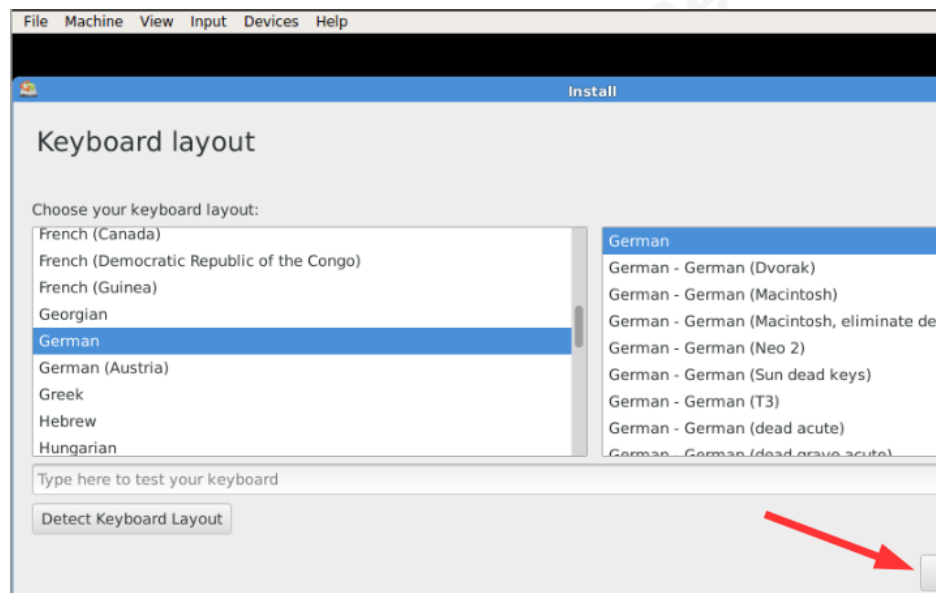


Figure 25: Installation screen where buttons are cut off

In some cases it is possible to use the mouse to drag the installation box to a place where the buttons are displayed and clickable. If this does not work, then it is possible to use the tab button to reach the desired button. This might take some trial-and-error to find the right number of tabs to get to the desired button.

Once the operating system installation is complete, the virtual machine will restart. After logging in, the screen will be noticeably small. To remedy this, install Virtual Box Guest Additions. Virtual Box Guest Additions can be installed in Security Onion with the following steps:

- Go to Devices → Insert Guest Additions.
- Select “Run” when prompted.
- Reboot.

After rebooting, eject the CD from the desktop.

Once Guest Additions has been added, it is possible to set up a shared folder with the host operating system. This is the most convenient way to move large PCAP files between the host operating system and Security Onion. To do this, perform the following steps:

- Go to Devices → Shared Folders → Shared Folder Settings.
- Add a shared folder and set it to “Automount” and “Make Permanent”.
- Restart the Virtual Machine.

The shared folder should be found under: `/media/sf_<folderOnHost>`.⁵ The shared folder will not be accessible due to a permissions issue. To solve this, issue the following command in the terminal:

```
$ sudo usermod -aG vboxsf <userName>
```

After this, restart the system. This should resolve the display resolution problem, allow access to the host drive where PCAP files are stored, and provide the opportunity to modify the network settings inside of the virtual machine. The next step is to install the Security Onion software.

On the desktop there will be an icon called “Setup”. Double click on the icon, and the installation process will start. Follow the prompts to configure the network with the following settings:

- **Management interface:** eth0, DHCP
- **Sniffing interface:** eth1

Once this installation sequence is completed, restart the operating system and double click on the Setup icon again. Skip configuring the network and install the remainder of the software. The following list provides specific guidance for setup.

- **Evaluation Mode or Production Mode?** Evaluation
- **Which network interface(s) should be monitored?** eth1
- **What would you like to configure HOME_NET as?** Accept the default values.
- **By default, the master server stores logs...** Yes, store logs locally.

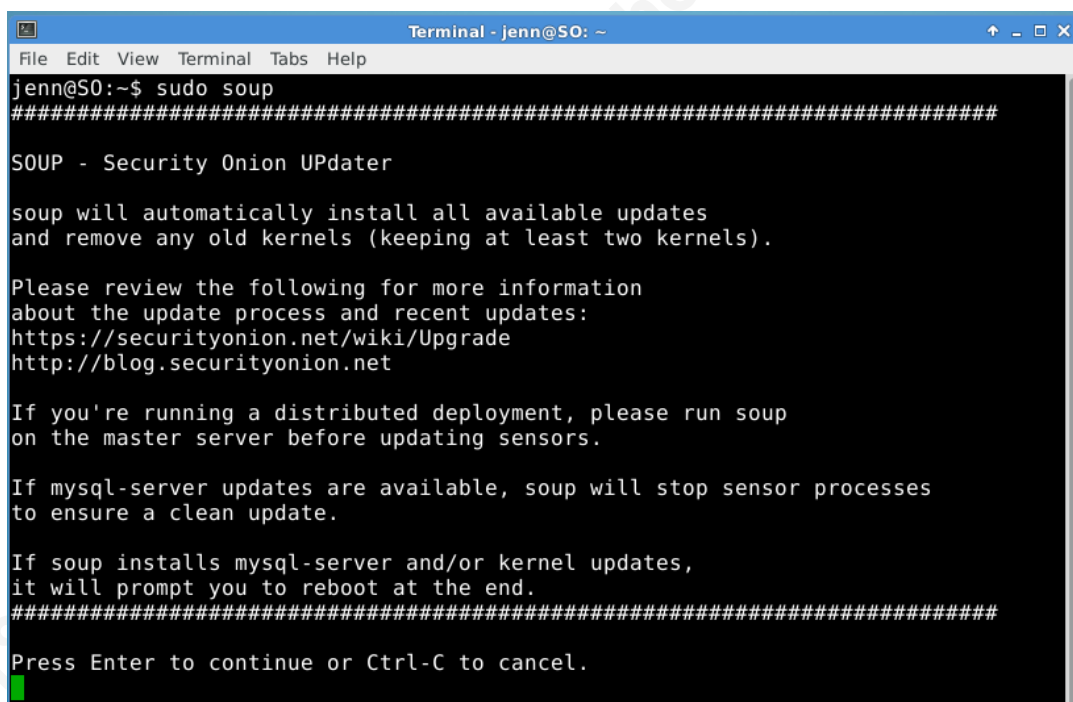
⁵It may be the case that the drive did not mount. If this is the case, perform the following command in the terminal:

```
$ sudo mount -t vboxsf <nameOfFolderOnHostSystem> /media/sf_<folderOnHost>
```

- **How much space should be allocated... to store logs? 20**

After this, accept the changes. This will complete the installation. The installation will have set up and configured SGUIL, Snort, the Elastic Stack, Bro and Kibana.

After this, it is time to update the software and operating system. To do this, open the terminal and type, `sudo soup`. This will load both approved updates for the operating system and for the Security Onion software and configurations.



```

Terminal - jenn@SO: ~
File Edit View Terminal Tabs Help
jenn@SO:~$ sudo soup
#####
SOUP - Security Onion UPdater

soup will automatically install all available updates
and remove any old kernels (keeping at least two kernels).

Please review the following for more information
about the update process and recent updates:
https://securityonion.net/wiki/Upgrade
http://blog.securityonion.net

If you're running a distributed deployment, please run soup
on the master server before updating sensors.

If mysql-server updates are available, soup will stop sensor processes
to ensure a clean update.

If soup installs mysql-server and/or kernel updates,
it will prompt you to reboot at the end.
#####
Press Enter to continue or Ctrl-C to cancel.
  
```

Figure X: Updating Security Onion environment

At the end of the update process, Security Onion will reboot.

Installing Grassmarlin

The next step in preparing the environment is to install Grassmarlin. There are two steps to installation: install Oracle Java and then install the Grassmarlin software package.

The latest, at the time of this writing, non-beta version of Grassmarlin can be found in the Grassmarlin git repository:

<https://github.com/iadgov/GRASSMARLIN/releases/tag/v3.2.1>. The

grassmarlin_3.2.1.ubuntu1404-1_amd64.deb software package should be selected.

The (Java 8) JDK can be downloaded from the Oracle site:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>. Select the version for Linux that is packaged as a .tar.gz file. Create a folder in /opt called Oracle_Java. Copy the downloaded file to the Oracle_Java folder and extract it.

```
$ sudo tar xvf jdk-XXX.tar.gz)
```

This will create a folder structure that looks like the following:

```
/opt/Oracle_Java/jdk-1.8.0_VERSION/
```

After this, run the following commands:

```
$ sudo update-alternatives --install "/usr/bin/java" "java"
"/opt/Oracle_Java/jdk1.8.0_VERSION/bin/java" 1
$ sudo update-alternatives --install "/usr/bin/javac" "javac"
"/opt/Oracle_Java/jdk1.8.0_VERSION/bin/javac" 1
$ sudo update-alternatives --install "/usr/bin/javaws" "javaws"
"/opt/Oracle_Java/jdk1.8.0_VERSION/bin/javaws" 1
$ sudo update-alternatives --install "/usr/bin/jar" "jar"
"/opt/Oracle_Java/jdk1.8.0_VERSION/bin/jar" 1
$ sudo update-alternatives --set "java"
"/opt/Oracle_Java/jdk1.8.0_VERSION/bin/java"
$ sudo update-alternatives --set "javac"
"/opt/Oracle_Java/jdk1.8.0_VERSION/bin/javac"
$ sudo update-alternatives --set "javaws"
"/opt/Oracle_Java/jdk1.8.0_VERSION/bin/javaws"
$ sudo update-alternatives --set "jar"
"/opt/Oracle_Java/jdk1.8.0_VERSION/bin/jar"
(march, 2018)
```

To test that the installation is complete, run the following command in the terminal:

```
$ java -version
```

This should return the installed Java version and information about the operating environment.

To install Grassmarlin, use the terminal and navigate to the directory where Grassmarlin was downloaded. Issue the following command:

```
$ sudo dpkg -i grassmarlin_3.2.1.ubuntu1404-1_amd64.deb
```

After this, step through the installation process. Once the installation process is finished, shut down the virtual machine.

Cloning the Security Onion Virtual Machine

At this point, the core installation of Security Onion and Grassmarlin is complete. To prevent having to go through this installation process again, a full clone of the virtual machine can be made. The cloned virtual machine can serve as a clean starting point for new security analysis engagements.