Bezborodov Sergey

Intrusion Detection System and Intrusion Prevention System with Snort provided by Security Onion.

Bachelor's Thesis
Information Technology

May 2016

DESCRIPTION

MAMK University of Applied Sciences	**Date of the bachelor's thesis** 06.05.2016
Author(s) Bezborodov Sergey	**Degree programme and option** Information Technology

Name of the bachelor's thesis

Intrusion Detection Systems and Intrusion Prevention System with Snort provided by Security Onion.

Abstract

In this thesis I wanted to get familiar with Snort IDS/IPS. I used the Security Onion distribution with a lot of security tools, but I concentrated on Snort. Also I needed to evaluate Security Onion environment and check what features it provides for processing with Snort. During the work I needed to figure out the pros and cons of using Security Onion with Snort as a security system for network. I compared it with alternatives and briefly describe it.

As result I installed Security Onion, work with the environment, configured different features, created and modified rules and so on.

I think this thesis will be helpful for people who want to use IDS/IPS for their network, it should help them to choose IDS/IPS vendor, make Security Onion and Snort installation, make comparison with another one and just get familiar with the network security tools. Also, this thesis can be a part of big research of network security tools, because now is impossible to find such detailed guides and literatures about any IDS/IPS tools. It is a good idea to combine many researches about it and make a good library.

This thesis can be used for further development of Snort and Security Onion as it only on the development phase now, it will provide base for new researching with new versions.

Subject headings, (keywords)

IDS,IPS, SecurityOnion

Pages	**Language** English	**URN**
Remarks, notes on appendices		
Tutor Matti Koivisto	**Employer of the bachelor's thesis**	

CONTENTS

1 INTRODUCTION

Global network named as the internet has become a part of our life. People interact with it every day and many of them link their life with it. Shopping, business, banking, transactions, payments and many more aspects of life are carried out the internet. With the growth of functions provided by the internet the amount of detractors has also grown. A lot of harmful software, viruses and ways to hack anything in it, develops day by day.

In this case, network security has become a major issue. Each user wants to be safe while using any service. It is an important aspect for network providers to guarantee security in their environments. Although losses for a single user may not be so big or harmful, but for large companies security breaches can mean millions of dollars. The surface of attacking can be everything, personal data, bank accounts, software, user accounts, local network, router, etc. That is why security tools develop all the time to meet the new forms of malicious, harmful software and the number of hackers.

There are many different ways and methods to increase network security but in this thesis I concentrate on Intrusion Prevention (IPS) and Intrusion Detection System (IDS). They cover the big part of network security and allow us to control major aspects. The purpose of IDS is to find different types of malware activities that are dangerous for computers and devices. Such activities include: network attacks against vulnerable services, attacks against privilege escalation, unauthorized access to sensitive files as well as the actions of malware (viruses, Trojans and worms).

IPS can be seen as an extension of IDS. Its purposes are to find malware activities and to prevent them. The main difference between them is that IPS has to keep track activities in real time and to have fast reaction on it.

This thesis bases in the Linux distribution called Security Onion which provides a lot of tools for network security, but our main area of interest is Snort. Snort – a free IPS and IDS with open source, is capable of performing packet logging and real-time traffic analysis in IP-based networks. It is widely used for the active blocking or passively detecting a variety of attacks and probes, such as the attempts to buffer overflows, hidden port scans, attacks on web applications, SMB sensing and the attempts to determine the operating system. The software is primarily used to prevent penetration and block attacks.

The study is divided into two parts: theoretical and practical. In the theoretical part I analyze the alternatives of IPS and IDS, and find the advantages of using Snort in comparison with other similar tools. The next steps are to figure out how Snort works, and to determine major things needed to install it for a small business network (preparing for installation).

The practical aim of the study is to install and configure Snort based IDS/IPS. Finally, I will test it, work with Snort rules and the environment features. As a result of testing it from both sides we can clearly say how useful tool it is.

The structure of the study is as follows. Chapter 2 covers the working principles of IDS and IPS followed by the introduction to Snort. The practical part of the study starts in Chapter 3. It contains the installation, configuration and the testing of the systems. In Chapter 4 I report the findings of my test and practical implementation. Finally, I sum up my findings in Chapter 5.

2 INTRUSION DETECTION AND PREVENTION

2.1 What are Security Onion and Snort?

2.1.1 Security Onion

There are many Linux distributions which have paid special attention to the network security and security tools. Security Onion - an Ubuntu based Linux distribution - is one example. It contains a set of specific tools for security including Snort, Bro, Suricata, Sguil, Squert, Snorby, Xplico, NetworkMiner and others (Security Onion 2016). These programs exist and are developed as a independent decision-making tools. But because of the highly specialized purpose, not all Linux distributions have them in their repositories. And, their installation from the source code may be difficult. The latest version of Security Onion solves this problem very efficiently and successfully.

But the main thing is not even that. There are other distributions, created specifically for the use of any of these programs. Security Onion gathered together almost everything you need for providing network security. The author of the Security Onion Doug Burks claims that his distribution can be used as an Intrusion Detection System (IDS) as a system and network monitoring (Burks 2008). The user is prompted to configure a very simple wizard, after which working tools are beginning to work. We check how it looks in practice in Chapter 3.

Briefly about major programs included to the Security Onion (Security Onion 2016):

Snort: It is a system of detection and prevention of network intrusions (NIDS / NIPS). The program detects intrusion attempts by analyzing network traffic in real time. The program uses sets of rules called Sourcefire VRT, which are regularly updated. Despite the fact that the program itself is free, the spreading of these rules is limited to the paid subscription. The user can download them for free only with a delay of one month after the appearance of the program or when upgrading to a new version. I describe Snort in more detail in the next section.

Suricata: It has similar functions as Snort. The advantages of the program are its modularity, high performance and automatic recognition protocols. Rulesets recommended are the same as for Snort, despite the fact that there are alternative rules.

Sguil: This network monitoring system collects and analyzes network events from various utilities, including the Snort and Suricata. Sguil gives administrators the ability to monitor real-time events in the network by the client with a graphical interface and to generate reports with it. All information is stored in the database Mysql.

Snorby: It is a web-based application for monitoring network security. It displays information of Snort’s and Suricata’s events.

VGO: It is an independent network intrusion detection and traffic analysis which monitors real time, creates the notification and keeps a log of events.

ELSA: This is a powerful application for centralized processing system logs, which uses Syslog-NG, MySQL and Sphinx. ELSA provides the administrator with a web interface like a search engine. The user can configure email alerts and perform tasks on a schedule. It is possible to create reports in the form of graphs.

Xplico: Unlike other programs that listen to network traffic, Xplico selects the packet stream data only - the content they carry. Xplico is able to recover the data transmitted on the protocols SIP, IRC, HTTP, IMAP, POP, SMTP, FTP, and others. A complete list of applications is included in Security Onion.

2.1.2 Snort

The Snort IDS and IPS system became a worldwide famous feature to protect your network. It is based on the following five modules (Snort 2016):

Packet sniffer: It is responsible for the capture of data transmitted over the network for subsequent transmission to the decoder. It does this with the help of a library called DAQ (Data AcQuisition). The sniffer can work "in the gap", in passive mode or read network data from a prepared file.

Decoder packets: This module deals with parsing the headers of captured packets, parsing them, finding anomalies and deviations from the RFC, the analysis of TCP flags, except for certain protocols of further analysis and other similar work. It focuses on the decoder stack TCP / IP.

Preprocessors: If the decoder analyzes the traffic on the 2nd and 3rd levels of the reference model, the preprocessors are designed for a more detailed analysis and normalization protocols on the 3rd, 4th and the 7th levels. Among the most popular preprocessors can be called frag3 (working with fragmented traffic), stream5 (reconstruction of TCP flows), http_inspect_ (normalization of HTTP traffic), DCE / RPC2, sfPortscan (used to detect port scans) and various decoder protocols Telnet, FTP, SMTP, SIP, SSL, SSH, IMAP and so on.

Intrusion detection engine: This engine is made up of two parts. Rules constructor collects a lot of different major rules (attack signatures) in a single set, optimized for further using in the subsystem for the inspection of the captured and processed traffic. (Snort 2016.)

Output module. Upon detection of attacks Snort can give (to record or display) a message in different formats - file, syslog, ASCII, PCAP, Unified2 (binary format for quick and easy handling).

Before and after the Snort appearance there were many different intrusion detection systems, but soon Snort earned its status as a famous de-facto standard. It has more than four million downloads from the site www.snort.org. What caused such a love for Snort? The main issue might be the description language used for network security policy violations. On the one hand, this language is very simple. And generally for the detection of attacks or other violations of security policies can be written in just a couple of minutes (or even faster). However, filters, complex queries, combining rules established thresholds and consideration of slots allow to the user to write really complex network event handlers.
Working principle shown in Figure 1.

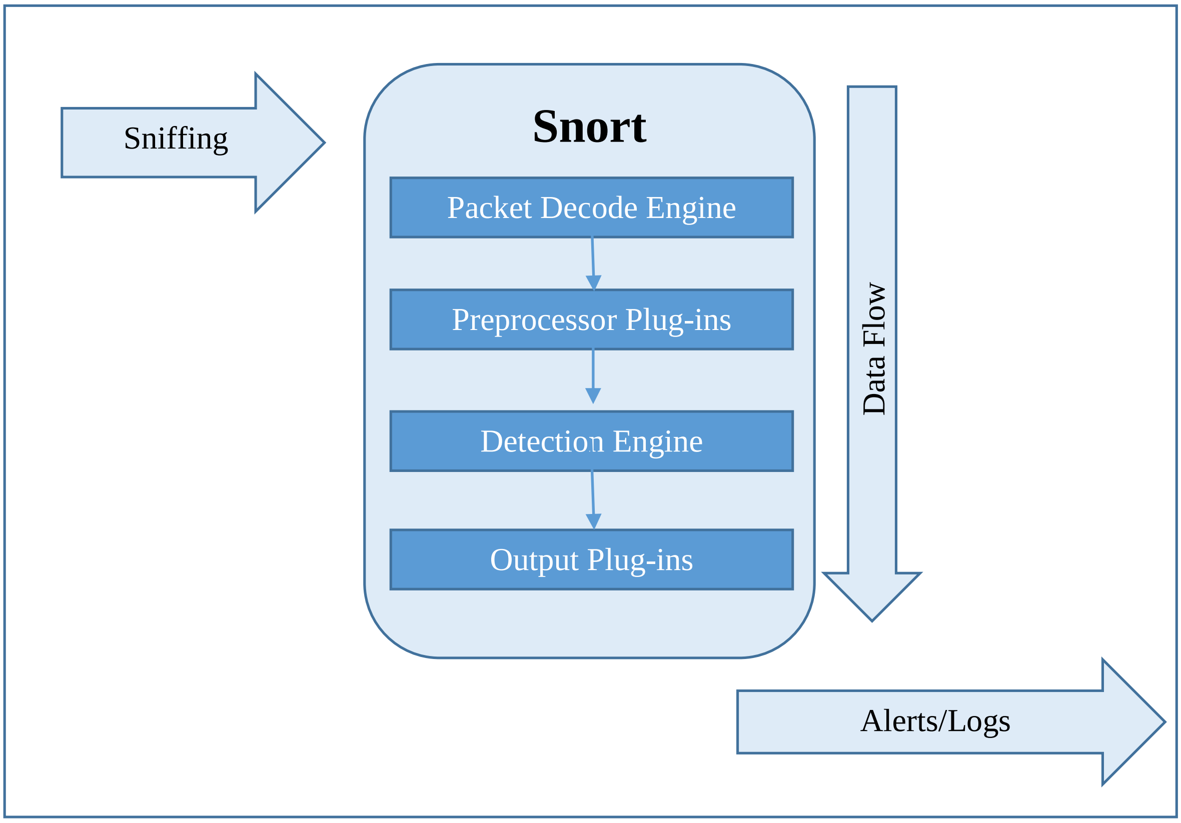


Figure 1. Snort working principles

2.2 IDS and IPS working principles

In this section I go through the working principles of Intrusion Detection and Prevention Systems. This section is mainly based on the publication of National Institute of Standards and Technology (NIST) (Scarfone & Mell 2007).

IDS is software or hardware tool for detecting facts of unauthorized access to a computer system or network commonly through the internet. It provides an extended level of a computer system's security.

The architecture of IDS includes:

- sensory subsystem, designed for collection events, related to the security of the system
- analysis subsystem, designed for detecting attacks and suspicious activity based on sensory subsystem
- storage, providing storing of the first events and the results of analysis
- control console, for configuring IDS, watching security statement and checking analysis subsystem incidents

There are different kinds of IDS. The most typical ones are:

- Network-based IDS (NIDS) – to detect intrusions by checking network traffic and monitors for few hosts. NIDS gets access to network traffic by connecting to a hub or a switch configured to ports mirroring or a network TAP device.
- Protocol-based IDS (PIDS) – system that detects and analyses communication protocols with related systems and users. For a web-server such IDS commonly monitors HTTP and HTTPS protocols. During using HTTPS, the IDS must be allocated in the interface where it can monitor HTTPS packets before their encryption and sending to the network.
- Application Protocol-based IDS (APIDS) – system that monitors and analyses data by using specific protocols for particular applications.
- Host-based IDS (HIDS) – system that is allocated in a host and detects intrusions and bases on the analysis of system callings, application logs, file modifications, host statements and other sources.
- Hybrid IDS – system that contains two or more way of IDS. The data of host agents combine with network information for creation clear view of network secure.

2.2.1 Passive and active systems of intrusion detection

In the passive IDS an information about security issues are written into application log file, also alerts are sent to console and/or system administrator by directed link. The active system is known as Intrusion Prevention System (IPS) that reacts to malicious activity through connection dropping or reconfiguring a firewall for blocking traffic from malefactor. This action can run automatically or through operator commands.

IPS is a software or hardware system of network and computer security that detects intrusion and malicious activity and prevents it. The IPS system can looks like an extension of IDS, as it has the same purpose to attacks. However, their difference is that IPS must work in real time and reacts to attacks immediately.

There are many different ways to classify the IPS systems. Scarfone and Mell (2007) use the following classification:

- Network-based IPS (NIPS) monitors traffic in the network and blocks suspicious data stream.
- Wireless Intrusion Prevention Systems (WIPS) monitor actions in the wireless networks. Commonly, it detects wrong configured wireless access points, man-in-the-middle attacks, Mac addresses spoofing.
- Network Behavior Analysis (NBA) analyzes network traffic and looks for untypical streams, such as DoS and DDoS attacks.
- Host-based Intrusion Prevention (HIPS) is resident program, it detects suspicious actions on the computer.

2.2.2. Methods of reacting to attacks

When an attack has been recognized there are many different ways to react. The most typical methods are:

After attack start. Methods implement already after attack was detected. This means that even in the case of successful protection system can be harmed.

Connection blocking. If for an attack used TCD connection, it will be closed by sending to each one or someone TCP packet with RST flag. In that case malefactor loses possibilities to continue the attack by using this network connection. This method performs with using networks sensors, commonly. It has two major cons:

- It does not support protocols, other than TCP, which doesn’t need pre-installed connection (such as UDP and ICMP).

- This method can be used only after malefactor got unauthorized access.

User's account blocking. If multiple user accounts have been compromised as a result of the attack, or were their sources, then performed their blocking through sensor host system. To lock it, the sensors have to be running on behalf of the account that has administrator rights. Also, the blocking can occur at a given time, which is determined by the settings of IPS.

Computer network host blocking. If from one of the host has been detected an attack, then host sensors can block it. Also can occurs network interface blocking of this host, or a router or switch, with which is connected to the network. Unlocking can take a specified period of time or by the activation of the security administrator. Lock will not be canceled due to restart, or disconnecting from the network host. Just to neutralize the attacks can be blocked the target, host computer network.

Attack blocking with Firewall. IPS create and send new configurations to Firewall. Firewall will use it to filter traffic from malefactor. This reconfiguration can perform in automatic way with OPSEC standards (ex. SAMP, CPMI). For Firewalls which do not support OPSEC protocols, for interaction with IPS can be used adaptor-module:

- It will get commands about changes of Firewall configurations.
- It will edit Firewall configurations for parameters modification.

Changing the configurations of the communication device. For SNMP protocol, IPS analyzes and changes MIB database parameters. Device's agent is needed to block an attack. Also TFTP, Telnet and others protocols can be used for.

Active suppression of attack source. Theoretically, this method used when others are useless. IPS detects and blocks packets of the malefactor. After IPS attacks his node, in case, if his address is known and attack will not disturb other legal nodes.

Such method has been realized in:

- NetBuster – prevent intrusion of “Trojan horse". It can be used as «fool-the-one-trying-to-NetBus-you». In this case NetBuster finds the malware and detects computer started it, then send this malware back.
- Tambu UDP Scrambler works with UDP ports. The product acts not only such as fictitious UDP-port, it can “paralyze” hacker's devices through UDP flooder.

At the beginning of the attack. Method are used before attack reach the goal.

2.2.3 Sensors

Intrusion Prevention Systems are based on sensors, which recognize abnormal activity. There sensors can be classified as network and host based alternatives.

Network sensors allocated in channel communication gap for analysis of passing packets. For this purpose, network sensors equipped with two network adaptors, which works in “mixed mode” – receiving and transferring. It writes all passing packets in buffer memory, whence IPS can get one. In case of attack packets can be deleted. Analysis of packets are based on signature or behavioral methods.

Host sensors can be used to detect both remote and local attacks as follows:

- Remote attacks, when malefactor sends series of packets. Sensors detects and analyze packets on different interaction layers. It helps to prevent attacks through crypto-secured IPsec and SSL/TLS connections.
- Local attacks, when malefactor do something, what violates system security. Sensors intercept all system calls to applications, analyzes it and blocks calling which pose a risk.

2.3 Comparing IDS with firewall and IPS development

Intrusion Prevention Systems were at the intersection of two technologies: firewalls and intrusion detection systems (IDS). The first is able to pass traffic through itself, but only analyzed IP-packet headers. The second, on the contrary, is "able to" all the things that have been deprived of firewalls, that is, analyze the traffic, but could not in any way affect the situation, as established in parallel and the traffic itself is not passed through. Taking the best of each technology emerged IPS system. Firewall limits entrance to host or subnetwork some kinds of traffic for intrusion prevention, but does not monitor intrusions inside network.

The formation of modern IPS-systems went in four directions. The first direction - IDS development in the inline-IDS. In other words, it was necessary to build a system in IDS-network series instead of parallel. The solution was simple and effective: IDS placed between protected and unprotected resources.

The second direction of IPS not less logical: the evolution of the firewall. As you can imagine, it did not have depth analysis of traffic passing through itself. Adding functional deep penetration into the body of data and understanding of the transmission protocols allowed to become firewalls these IPS-systems.

The third "source" of development came from antivirus solutions. From the struggle with "worms," "Trojan horses" and other malicious software to the IPS-systems it was quite close. Perhaps, this direction gave start for HIPS.

Finally, the fourth direction was the creation of IPS-system "from scratch".

Like all other security solutions also IPS has its own problems Three major problems were identified:

1. a large number of false positives
2. reaction automation
3. a large number of administrative tasks

With the development of systems, these problems were solved successfully. So, for example, to reduce the percentage of false positives began to use event correlation system that "prioritize" for the event and helped IPS-system effectively perform its tasks. All this has led to the emergence of next-generation IPS-systems (Next Generation IPS - NGIPS). NGIPS must have the following minimum features:

1. Work in real-time without affecting (or with minimal impact) on the network activity of the company
2. To act as a single platform that combines all the advantages of the previous generation of IPS, as well as new features: control and monitoring applications; use information from third party sources (database vulnerabilities, geolocation data, etc.); analysis of the file contents.

2.4 Snort (IDS/IPS vendor) alternatives

Top five free enterprise network IDS/IPS tools listed by Techtarget (2016) are:

- Security Onion
- OSSEC
- Open WIPS-NG
- Suricata
- Bro IDS

Security Onion is most flexible system, includes all this tools and allows to work together. We already reviewed this Snort vendor in Section 2.1 and will use it for our thesis work.

OSSEC (Open Source Host-based Intrusion Detection System) - is a host system intrusion detection. If you have an aim to checking file integrity monitoring on your servers, logging of various actions on the server, obtain security events from your servers (or any other) and notification of these events, display a variety of reports and much more, the HIDS OSSEC - the perfect solution under these tasks. OSSEC can work locally, according to the scheme agent<->server and in hybrid mode (Agency> server-> server). Like most of IDS vendors it offers multiply tools for system security, what can be used with the core functionality of IDS (OSSEC, 2016):

The following extensions exist in OSSEC Table 1.

Table 1. OSSEC extensions

Unix-specific: • Unix PAM • sshd (OpenSSH) • Solaris telnetd • Samba • Su • Sudo	FTP-servers: • ProFTPd • Pure-FTPd • vsftpd • Microsoft FTP Server • Solaris ftpd

NIDS: • Cisco IOS, IDS / IPS module • Snort IDS (snort full, snort fast and snort syslog)	Database: • PostgreSQL • MySQL
Mail servers: • Imapd and pop3d • Postfix • Sendmail • Vpopmail • Microsoft Exchange Server •	Internet applications: • Horde IMP • SquirrelMail • Modsecurity
Firewall: • Iptables • Solaris IPFilter • AIX ipsec / firewall • Netscreen • Windows Firewall • Cisco PIX • Cisco FWSM • Cisco ASA	Web-servers: • Apache HTTP Server (access logs and error) • IIS web server (including extensions NSCA and W3C) • Error Logs Zeus Web Server
Other: • Named (BIND) • Squid proxy • Zeus eXtensible Traffic Manager	Windows Event Logs (login, logon information for auditing, etc.) Logs Windows Routing and Remote Access Unix authentication tools (adduser, logins, etc.)

Open WIPS-NG is a free wireless IDS/IPS. It relies on a server, sensors and interfaces. It runs on commodity hardware. WIPS-NG based on Airckrack-NG, author created both systems, and WIPS-NG borrowed scanning, detection and prevention tools from one. WIPS-NG has extension plugins for more flexibility. It is not so famous and developed system, it has not detailed documentation, and community is not so big in comparison with other systems. But it allows to perform Wireless Intrusion Prevention System for small enterprises with small budget. (Open WIPS-NG, 2016)

Suricata is another open source IPS / IDS system. Founded by developers who worked on the IPS version of Snort. The main difference of Suricata to Snort - more advanced system IPS, multitasking, as a result of high productivity, allowing traffic to handle 10Gbit on conventional equipment, and more, including full support for Snort rules format. That is why Suricata competes with Snort in IDS/IPS providing.

Suricata such as Snort consist of few modules (capturing, collection, decoding, detecting and output), captured traffic passing in one flow before decoding, it is optimal way for decoding, but it loads system much. But in comparison with the Snort it can be changed by configuring, separate flows after capturing and specify how flows will separate between processors. It gives wide possibilities for optimization of traffic handling to the particular devices in the particular networks.

There are developed tools of HTTP-traffic inspection based on HTP Library, created by Ivan Ristic, the author of ModSecurity. Supports recovery and verification of transmitted HTTP files, parsing compressed content, the possibility of identification URI, cookie, titles, user-agent, the body of the request and response. This opportunity of the Suricata, by the way, in some networks are used for logging of HTTP-traffic without detection. The content in the stream can be isolated behind a mask and using regular expressions, files identify possibility by name, type or check the MD5-sum.

Initially, supported IPv6 decoding, also tunnels IPv4-in- IPv6, IPv6-in-IPv6, Teredo and others. The modular layout of the engine makes it possible to quickly connect a new element for capturing, decoding, analysis or packet processing. For interception the traffic uses multiple interfaces - NFQueue, IPFRing, LibPcap, IPFW, AF_PACKET, PF_RING. Unix Socket mode allows you to automatically parse PCAP-files previously captured by another program (sniffer, for example).

At the beginning Snort did not have IPS. In the Suricata IPS works from the first version of it, and performs by the regular tools of th OS’s batch filter. Linux used two kinds of IPS: through queue NFQUEUE, what can be threated on the user layer, and through “zero copy”

AF_PACKET mode (appeared in ver.1.4.). AF_PACKET has good speed, but requires two network interfaces, system have to work as gateway.

Main difference of Suricata - it can use not only own develops, also it can use Snorts develops. Snort rulesets are suitable for Suricata, such as: Sourcefire VRT, OpenSource Emerging Threats (ETOpen) and commercial Emerging Threats Pro. It has unified output (Unified2), so the result can be analyzed with using usual backends - Barnyard2, Snortsnarf, Snorby, Aanval, BASE, FPCGUI, Sguil and Squert NSM-system. It also has possible output in PCAP, Syslog, files and the like. For example, Suricata logs the keys and certificates appearing in the TLS / SSL-connections. In latest releases appeared Eve log, which forms the output of events in the JSON format for alerts. Availability JSON Suricata greatly simplifies integration with third-party applications, including system monitoring and visualization of logs (such as Kibana). Suricata settings and rules made in YML file format, it is more evident and simplifies automated processing.

One advantage of Suricata is the processing in 7th OSI level that enhances its ability to detect malware application. The engine automatically detects and parses protocols (IP, TCP, UDP, ICMP, HTTP, TLS, FTP, SMB, SMTP, etc.), so the rules cannot strictly tie to the port number, as in the Snort, is enough to specify the protocol and the action. Next Suricata modules themselves sorts out the traffic and find a protocol, even if a non-standard port uses. Native rules resembles to the Snort’s one. Rule contains components: action (pass, drop, reject or alert), header (IP / port source and destination) and a description (what to look for). The current supply of native rules is a little, and some also disabled (commented out) in the files, so now should to be more focused on Snort’s rulesets.

Some applications open several TCP-connections between nodes. Some IDS cannot see the whole picture, and treated each stream separately. Suricata riles widely used concept of the flowbits. For the tracking of the number of rules operations used different session variables (e.g., via flowint), what allows to create the counters and flags, and then check them. Such approach is easy to cope with trying password guessing. In the Version 2.1 exist possibility to easily track the rules IP source - IP destination (xbits), which allows even easier to detect malicious traffic, distributed across multiple connections.

In recent releases appeared subsystem IP Reputation, loads and uses in the rules data from different databases containing lists of hosts reputation. A special mechanism ensures a quick search and comparison with IP-addresses. (Suricata, 2016)

Although Suricata is a good alternative for an IDS it has some challenges. Suricata has less community than Snort, as result has less own developing. And it looks more complicated “from the box”. And also for using full functional product it requires more resources than Snort.

Bro IDS

Bro has the following features:

- Flexibility. It uses a scripting language that allows you to set for each protected object its monitoring rules. Furthermore, Bro initially does not aim to the detection of any specific attack, there is no dependence on the signatures.
- Efficiency. Bro is designed to work in networks with a very large volume of traffic and can be used in various large projects. In particular, it supports separated architecture.
- A deep analysis of the traffic. Supports analyzers for multiple protocols, which works with a high-level semantic analysis even at the application level.

Bro is a framework for creating a network IDS / IPS, and has a multi-level modular structure:

- The mechanism of packet capture. De facto, this mechanism is almost always uses data libpcap purposes, allowing Bro does not depend on the platform and the underlying network layer.
- Event Mechanism (Event Engine, also called Core) converts the incoming packet sequence in the primary event. These events reflect the basic information about network activity - so that every HTTP-request generates an event, which describes the address, port, and the requested URL version of the HTTP protocol. This mechanism, however, does not take any decision with respect to color events - that is, at this level is unknown, malicious or not.
- Actually the top level of the script interpreter (Policy Script Interpreter – reacts to each event what needed, registers his handler corresponding to a particular script. Events are placed in the FIFO queue. Scripts also determine the steps that are used to detect

malicious traffic, and policy that is applied when it is detected and written in its own scripting language Bro.

In practice, this platform can be used not only for the construction of IDS, also for other traffic analysis. It can, in principle, be replaced (and / or extends) Wireshark, highlighting and analyzing only the necessary traffic. Its recent versions include experimental support of ElasticSearch, full-text search engine. (Bro, 2016)

2.5 Advantages and disadvantages of Snort

After getting familiar with alternatives of Snort it is time to point out the main advantages and disadvantages of Snort itself (see Table 2) below.

Table 2. Advantage and disadvantages of Snort (Xaker 2016)

Advantages:	**Disadvantages:**
• Free for use • Available for Linux and Windows platforms • Flexible customization for many environments • Can be used in decentralized mode • Support auto-update rules • Opportunity to use MySQL database • Can be hidden in network • Exist web-interface • Low system requires (in comparison with Suricata)	• Intrusion Detections occurs behind firewall • Should be configured exactly for particular environment to avoid “false positives” • Not so easy for beginners • Unidirectional Ethernet cables should be used for sensors installation to avoid security concerns

2.6 Snort rules

Snort rules use lightweight and simple rule language. Most Snort rules are written on a single line. (Snort manual 2016)

It is divided into two logical sections: the rule header and the rule options. The rule header, the first part of Snort rule, contains:

- the rule's action

- protocol
- source and destination IP
- addresses and netmasks
- the source and destination ports information

The rule option section contains alert messages and information on which parts of the packet should be inspected to determine if the rule action should be taken.

Rule’s actions have 5 default functions, but if you run Snort in inline mode there are additional 3 functions:

- alert - generate an alert using the selected alert method, and then log the packet
- log - log the packet
- pass - ignore the packet
- activate - alert and then turn on another dynamic rule
- dynamic - remain idle until activated by an activate rule, then act as a log rule
- drop - block and log the packet
- reject - block the packet, log it, and then send a TCP reset, if the protocol is TCP, or an ICMP port unreachable message, if the protocol is UDP.
- sdrop - block the packet, but do not log it.

The next section of rules is the protocol. At the moment, there are only four protocol types: TCP, UDP, ICMP, and IP. IP addresses define the source or destination IP for rule, it can be a single address and range of addresses. Port Numbers have the same function as IP addresses and also have flexible configuration: ranges, single and negation.

Direction operator is used for indication of orientation or direction of traffic that the rule applies to. This operator uses the symbols: “<”, “>” and “<>”.

Active/Dynamic rules give more flexibility to your system, uses for specific options, it allows to user activate another rules when particular alert is indicated.

Rule’s options are the second part of rule and have four categories:

- General
- Payload
- Non-payload
- Post-detection

General options are shown in Table 3.

Table 3. General options

msg	tells the analyzer what to write in alert message
reference	allows including references to external attack identifications
gid	identifies what part of Snort identifies the event
sid	identical number of rule
rev	identifies a revision of Snort rule
classtype	categorizes an attack type
priority	sets priority to the event
metadata	allows to add additional information

Payload options are shown in Table 4.

Table 4. Payload options

content	This allows to search specific content in traffic
protected_content	This is the same function to content, but looks for particular bites
hash	This specifies the hashing algorithm
length	This specifies length of content
nocase	This says to Snort look for particular pattern and ignore case, content modification
rawbytes	This allows user to look into raw data, content modification
depth	This specifies level of Snort divining into data, content modification
offset	This allows the user to specify a rule where to start searching for a predetermined value within the package, content modification
distance	Snort indicates to how much data in the packet should be ignored after the last match, to again seek a specified value, content modification
within	This indicates the N number of bytes within which the search for the next match must be after a previous, content modification
http_client_body	This restricts the search to the values of HTTP client request body, content modification
http_cookie	This restricts the search values in the intercepted cookie headers, content modification
http_raw_cookie	This is used to search for titles’ UNNORMALIZED cookie headers, content modification

http_header	This allows the user to search the captured headers, content modification
http_raw_header	This is used to find the values in the headers UNNORMALIZED, content modification
http_method	This allows to intercept data transfer method, content modification
http_uri	This restricts the search values NORMALIZED URI requests, content modification
http_raw_uri	This restricts the search UNNORMALIZED URI requests, content modification
http_stat_code	This restricts the search values extracted data Status Code, content modification
http_stat_msg	This restricts the search values extracted data Status Message, content modification
http_encode	This allows the system to signal if the HTTP client-request or HTTP-server response have specified coded type
fast_pattern	This allows setting "mask" which can apply the same value within a single rule, content modification
uricontent	This is used for searching NORMALAIZED URI request
urilen	This determines the exact length, minimum length, maximum length or range of URI-length requests
isdataat	This checks whether the specified value is present in a certain place package
pcre	This is connected in the rules for the use of perl-compatible regular expressions
pkt_data	This sets the pointer used for the detection values in not encoded traffic
file_data	This sets the pointer to locate the data
base64_decode	This is used to decrypt the data encrypted in Base64 encoded
base64_data	This same to the file_data with base64 decoding
byte_test	This is used to specify the value of the byte to detect
byte_jump	This allows to create rules for length encoded protocols
byte_extract	This option allows to read a certain number of bytes of payload packets and store them in a variable
ftpbounce	This helps to detect ftp scanning

asn1	This decodes packet or a portion of the packet and looks for a variety of malicious coding
cvs	This is a plugin to detect vulnerabilities CVSS
And also payload have some preprocessor options such as: dce_iface, dce_opnum, dce_stub_data, sip_method, sip_stat_code, sip_header, sip_body, gtp_type, gtp_info, gtp_version, ssl_version, ssl_state.	

Non-payload options are shown in Table 5:

Table 5. Non-payload options

fragoffset	used to compare parts of biased IP-addresses with respect to the decimal value
ttl	value is used to check time-to-life IP-address
tos	is used to test IP TOS
id	is used to check ID Ip-address
ipopts	checks certain IP-addresses option
ragbits	is used to verify the packet fragmentation and reserved byte IP header
dsize	used to verify the size of packets within the payload
flags	is used to verify bit TCP flags
flow	used in conjunction with the tracking sessions
flowbits	used in conjunction with the preprocessor session tracking
set	organizing the bits in a group for a specific flow
setx	it sets the bit group excluding the other bits of it
unset	removes the bits belonging to any group in the current stream or clears all bits in the specified group
toggle	If flowbit installed, cancel the installation. If not installed – establishes
isset	checks one bit or group of bits, if there is the specified bit among them
isnoset	opposite to the isset
noalert	returns only false
reset	reset all statements of groups or clean bits of it
seq	is used to check specific TCP sequence numbers

ack	is used to check the number of TCP acknowledgment
window	is used to check the value of the packet TCP window size
itype	used to check the value of ICMP type
icode	used to find the values of the ICMP code
icmp	id is used to search for specific ICMP ID-values
icmp_seq	used to search for a specific ICMP sequence value
rpc	is used to check the version number and procedural queries in SUNRPC
ip_proto	allows you to compare the size of the header IP-based
same-ip	allows rules to check if the source and destination IP-addresses match
stream_reassembly	allows rules to enable or disable the streaming traffic bulkhead at the specified value
stream_size	allows to analyze the traffic rules for compliance with TCP Sequence parameter specified number of bytes

Post-detection options are shown in Table 6:

Table 6. Post-detection rules

logto	register all packets for specified rule to separated file
session	is used to retrieve user data from the TCP-session
tag	allows to record more data rules
activates	allows the author to add various events upon the occurrence of certain events
activated_by	allowing the author to dynamically enable rules for a particular event occurs
count	it allows rules to indicate how many interception of packages should be after activation the rule
replace	change detected matches to specified value
detection_filter	sets the level that must be exceeded by the source or final destination before the event occurs the rule

2.7 Working and installation features

Snort works on Linux and Windows platforms, but for Windows it takes a longer time for installation and there are some issues with Windows the reputation tools.

But in case of thesis for Snort research will be used Security Onion version. Security Onion only works with Ubuntu 14.04. So, it does not provide any Windows OS systems.

There are two ways of installation: to download the Security Onion .iso file and run it as a new installation, or it is possible to download Ubuntu 14.04 (or if you have already) and then add Security Onion PPA and packages. After choosing one of them you need to verify checksums. For this there are different installation guides. Before installation it is good idea to check hardware requirements. In case of this thesis I will use VMware Machine, so it is highly recommended to provide enough resources.

3 PRACTICAL IMPLEMENTATION

3.1 Basic installation

Installation starts from downloading the .iso file which includes Ubuntu 14.04 and the Security Onion pack. After getting the .iso file I run it on work machine. Installation includes basic settings for further work. Below I give a step-by-step description of the installation process.

1. During the first run of this .iso we get an option window as shown in Figure 2.

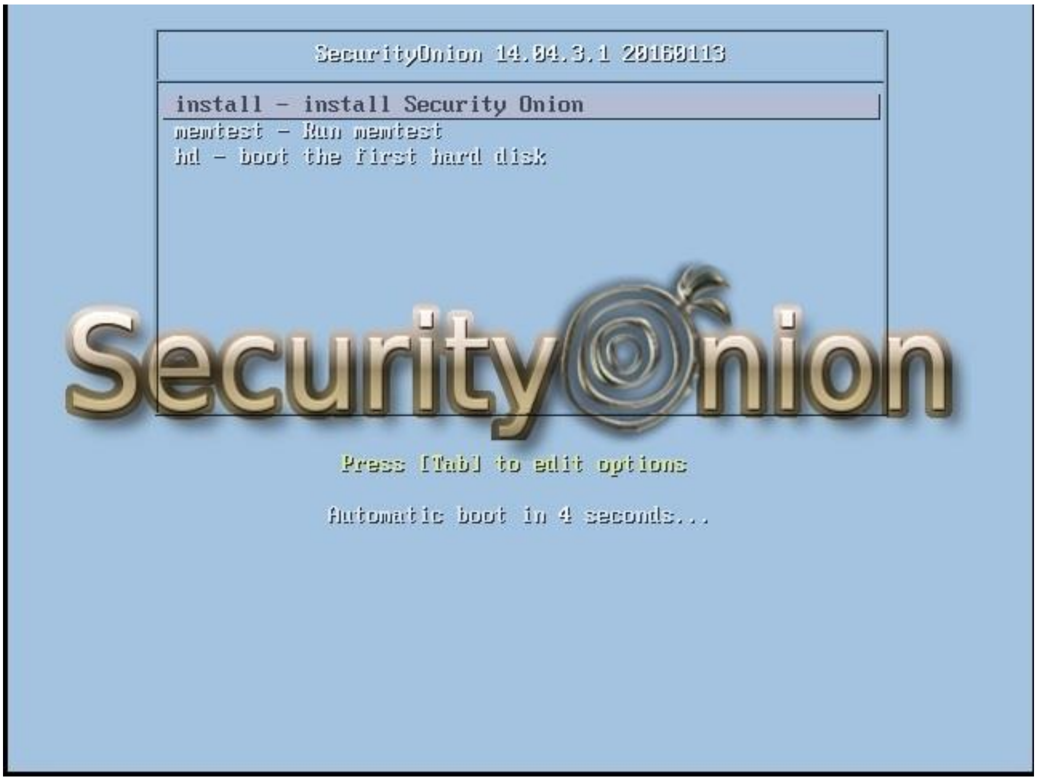
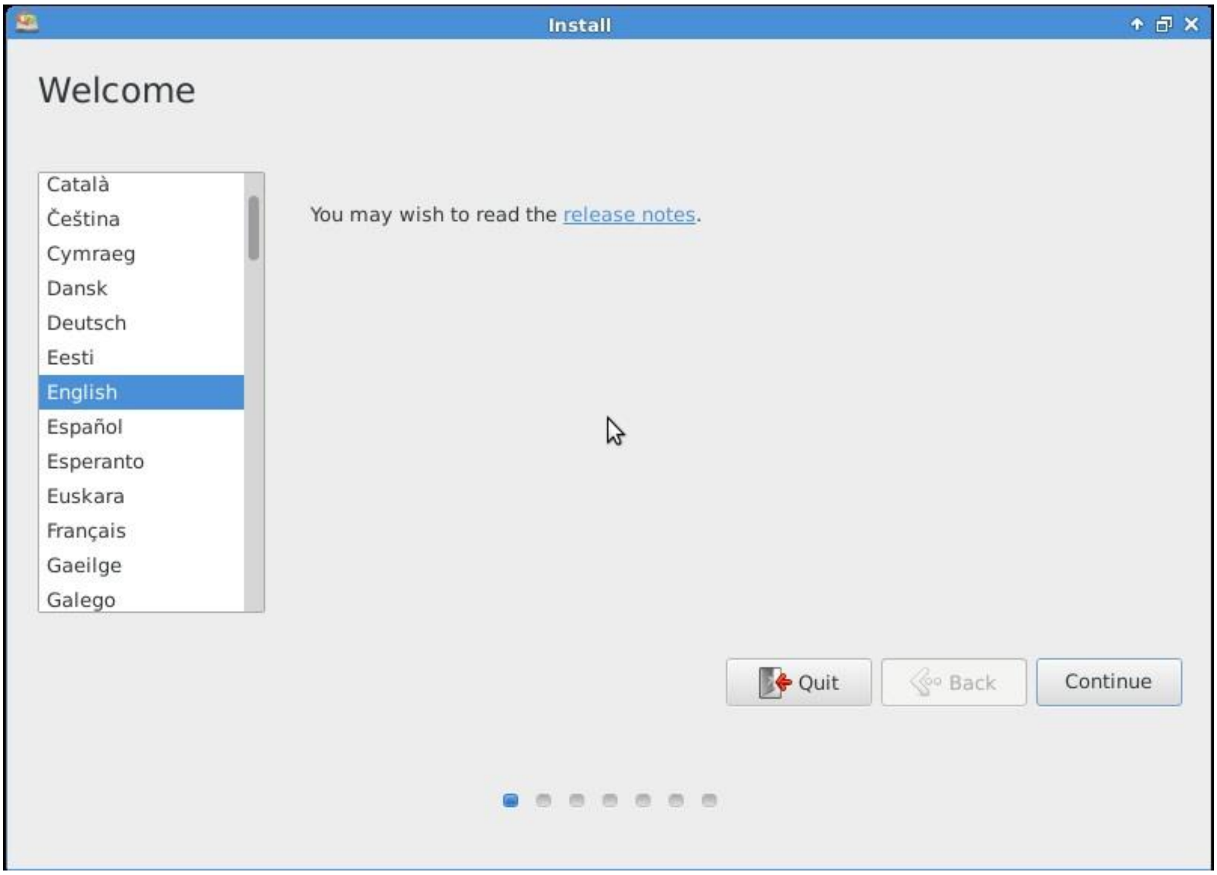

SecurityOnion 14.04.3.1 20160113
install - install Security Onion
memtest - Run memtest
hd - boot the first hard disk
SecurityOnion
Press [Tab] to edit options
Automatic boot in 4 seconds...

FIGURE 2. Booting menu

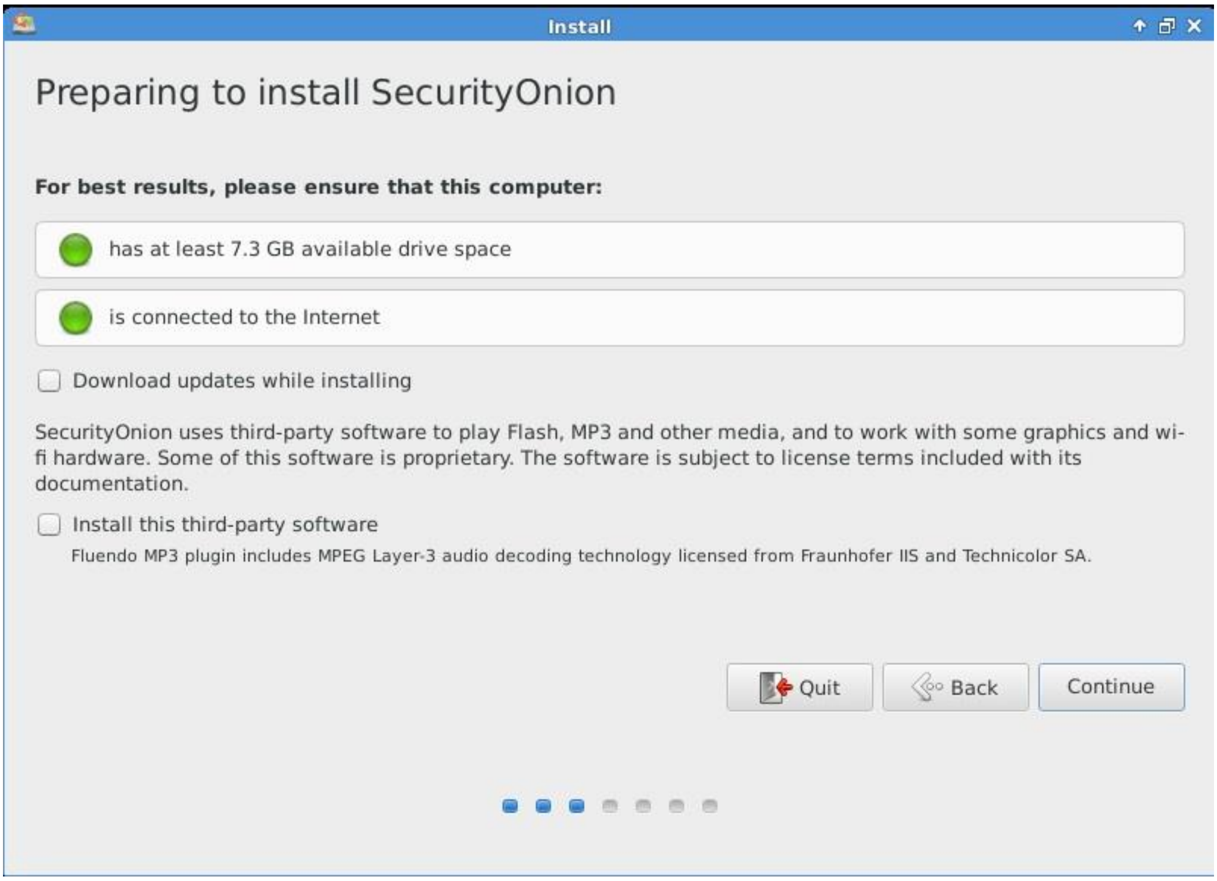
2. After this we have to choose the language for installing and further work. Also, it can be changed after the installation. Figure 3 shows the language selection screen.



You may wish to read the release notes.

FIGURE 3. Language selestion

3. The next step asks the user to check disk space and internet connection. Also it allows setting auto updating and installing third-party software (see Figure 4).



Preparing to install SecurityOnion

For best results, please ensure that this computer:

● has at least 7.3 GB available drive space

● is connected to the Internet

☐ Download updates while installing

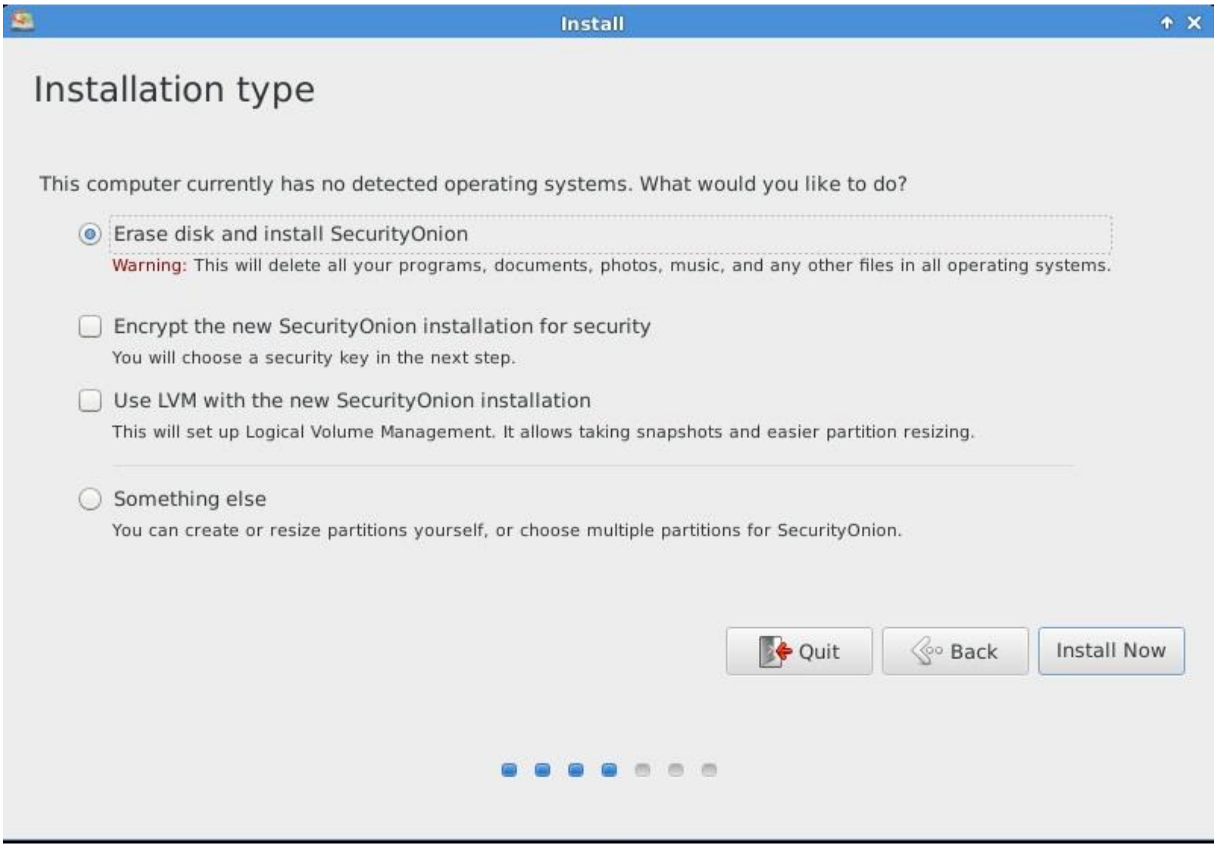
SecurityOnion uses third-party software to play Flash, MP3 and other media, and to work with some graphics and wi-fi hardware. Some of this software is proprietary. The software is subject to license terms included with its documentation.

☐ Install this third-party software

Fluendo MP3 plugin includes MPEG Layer-3 audio decoding technology licensed from Fraunhofer IIS and Technicolor SA.

FIGURE 4. Compatibility with the computer and additional software

4. At this phase shown in Figure 5 the user has to choose he installation method of the new operating system.



Installation type

This computer currently has no detected operating systems. What would you like to do?

● Erase disk and install SecurityOnion

Warning: This will delete all your programs, documents, photos, music, and any other files in all operating systems.

☐ Encrypt the new SecurityOnion installation for security

You will choose a security key in the next step.

☐ Use LVM with the new SecurityOnion installation

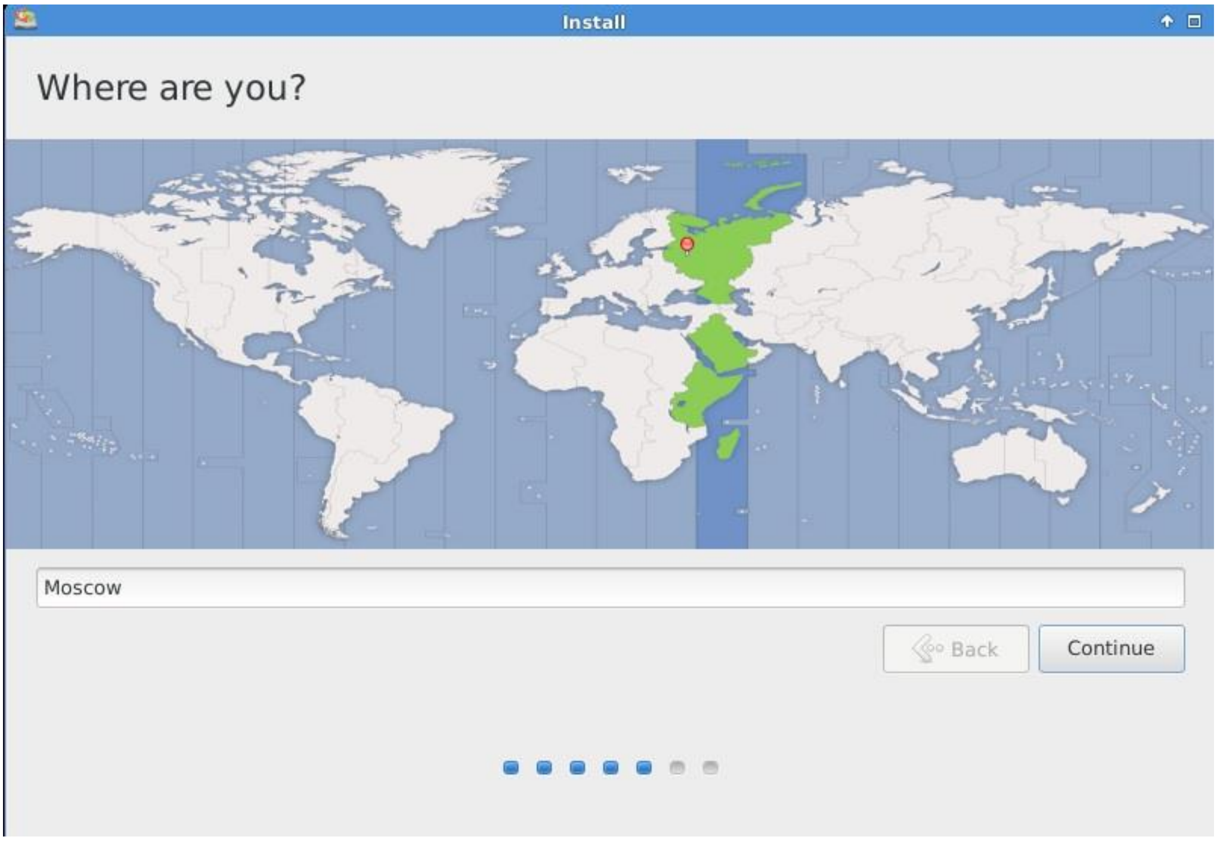
This will set up Logical Volume Management. It allows taking snapshots and easier partition resizing.

○ Something else

You can create or resize partitions yourself, or choose multiple partitions for SecurityOnion.

FIGURE 5. Installation type

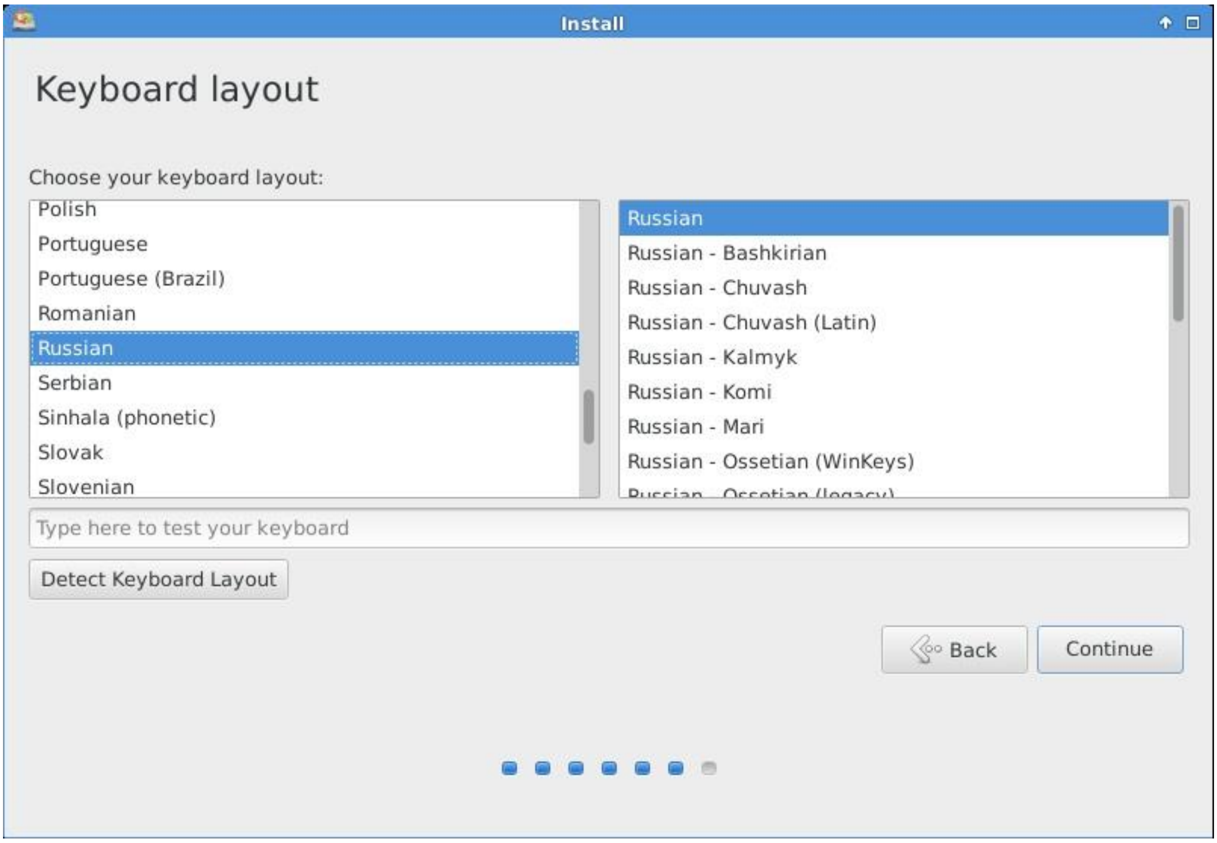
5. After this the user has to set up the location from the maps of Figure 6.



Moscow

FIGURE 6. Location choosing

6. Next step is keyboard settings as shown in Figure 7.



Keyboard layout

Choose your keyboard layout:

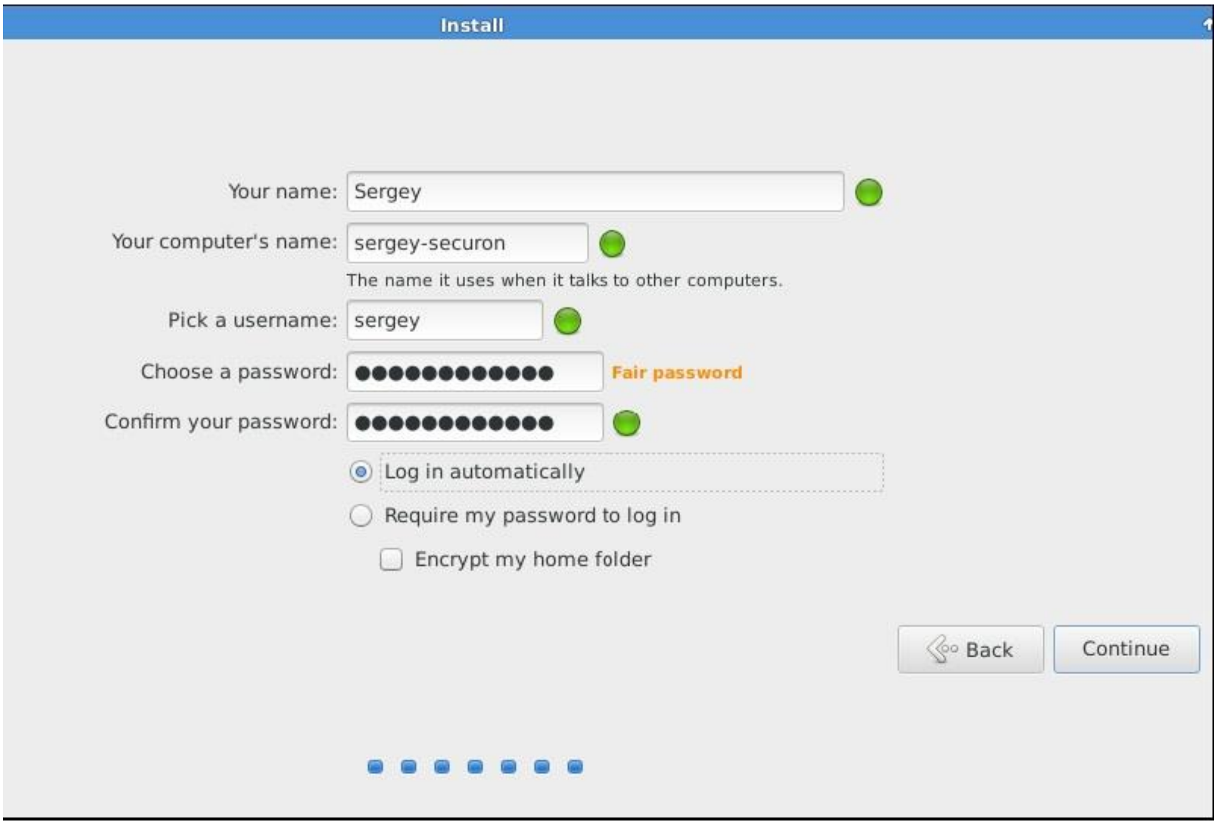
- Polish
- Portuguese
- Portuguese (Brazil)
- Romanian
- Russian
- Serbian
- Sinhala (phonetic)
- Slovak
- Slovenian

- Russian
- Russian - Bashkirian
- Russian - Chuvash
- Russian - Chuvash (Latin)
- Russian - Kalmyk
- Russian - Komi
- Russian - Mari
- Russian - Ossetian (WinKeys)

Type here to test your keyboard

FIGURE 7. Keyboard settings

7. Finally, the user fills in personal data, work machine name and password in the fields shown in Figure 8.



Your name: Sergey

Your computer's name: sergey-securon

The name it uses when it talks to other computers.

Pick a username: sergey

Choose a password: ●●●●●●●●●●●● Fair password

Confirm your password: ●●●●●●●●●●●●

● Log in automatically

○ Require my password to log in

☐ Encrypt my home folder

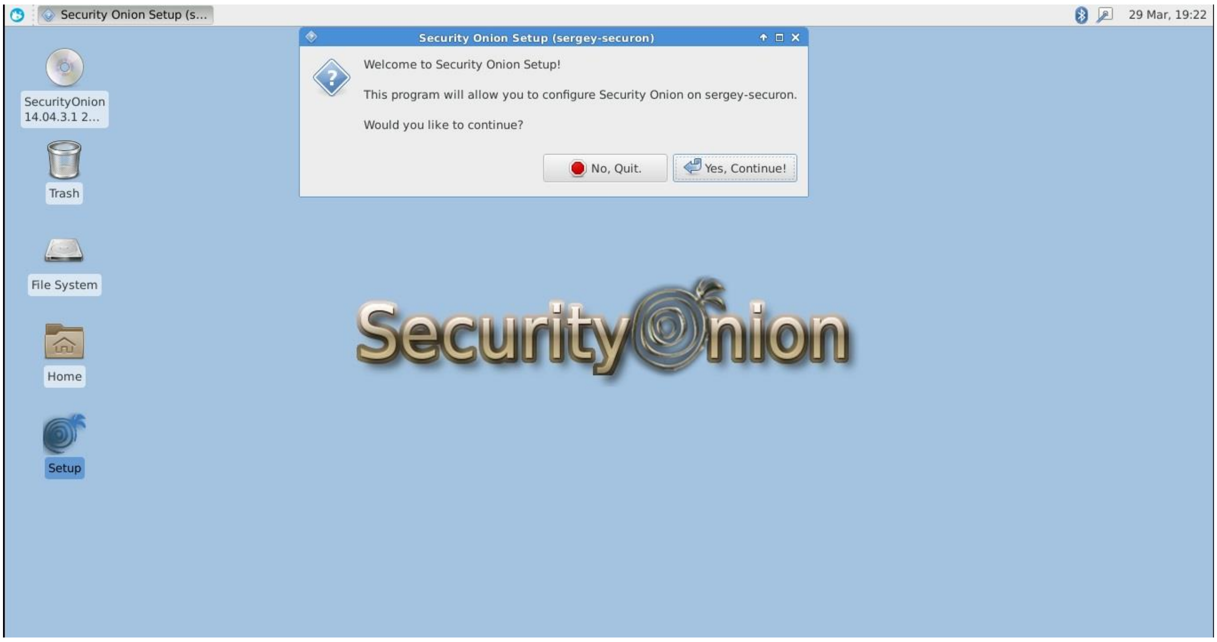
FIGURE 8. Username and password filing

8. Now all installation steps are completed and the user just needs to wait for the unpacking and installation.

3.2 Configuration

It is strongly recommended to update Security onion before next steps.

1. Configuration starts by clicking the icon on the desktop (Figure 9).



Security Onion Setup (sergey-securon)

Welcome to Security Onion Setup!

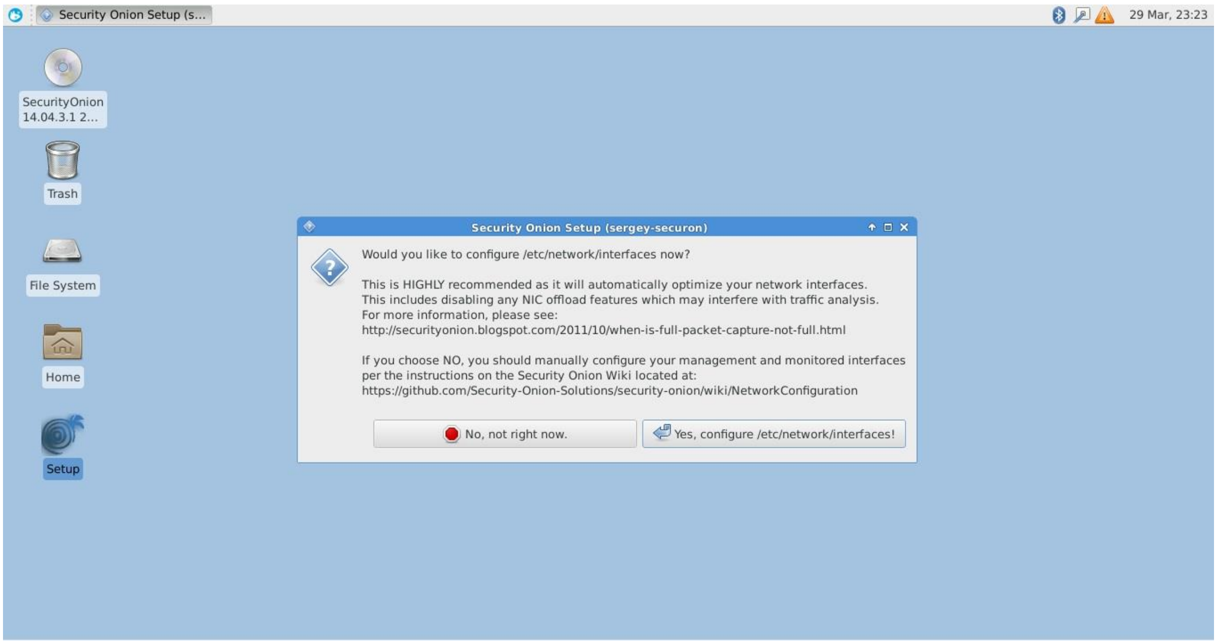
This program will allow you to configure Security Onion on sergey-securon.

Would you like to continue?

● No, Quit. | Yes, Continue!

FIGURE 9. Wizard start

2. It will ask the user about the root password and then introduction window appears.
3. To the user will be advised to configure network interfaces as shown at the Figure 10.


Security Onion Setup (s...
29 Mar, 23:23
SecurityOnion 14.04.3.1 2...
Trash
File System
Home
Setup
Security Onion Setup (sergey-securon)
No, not right now.
Yes, configure /etc/network/interfaces!

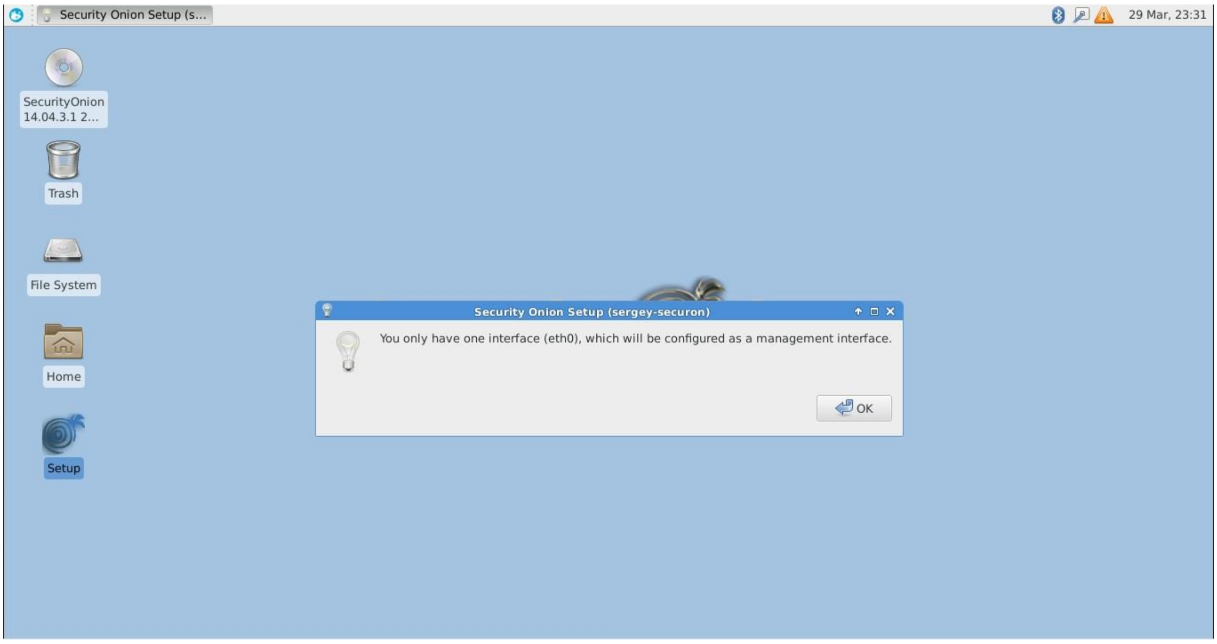
Would you like to configure /etc/network/interfaces now?

This is HIGHLY recommended as it will automatically optimize your network interfaces.
This includes disabling any NIC offload features which may interfere with traffic analysis.
For more information, please see:
http://securityonion.blogspot.com/2011/10/when-is-full-packet-capture-not-full.html

If you choose NO, you should manually configure your management and monitored interfaces per the instructions on the Security Onion Wiki located at:
https://github.com/Security-Onion-Solutions/security-onion/wiki/NetworkConfiguration

FIGURE 10. Network interface configuring

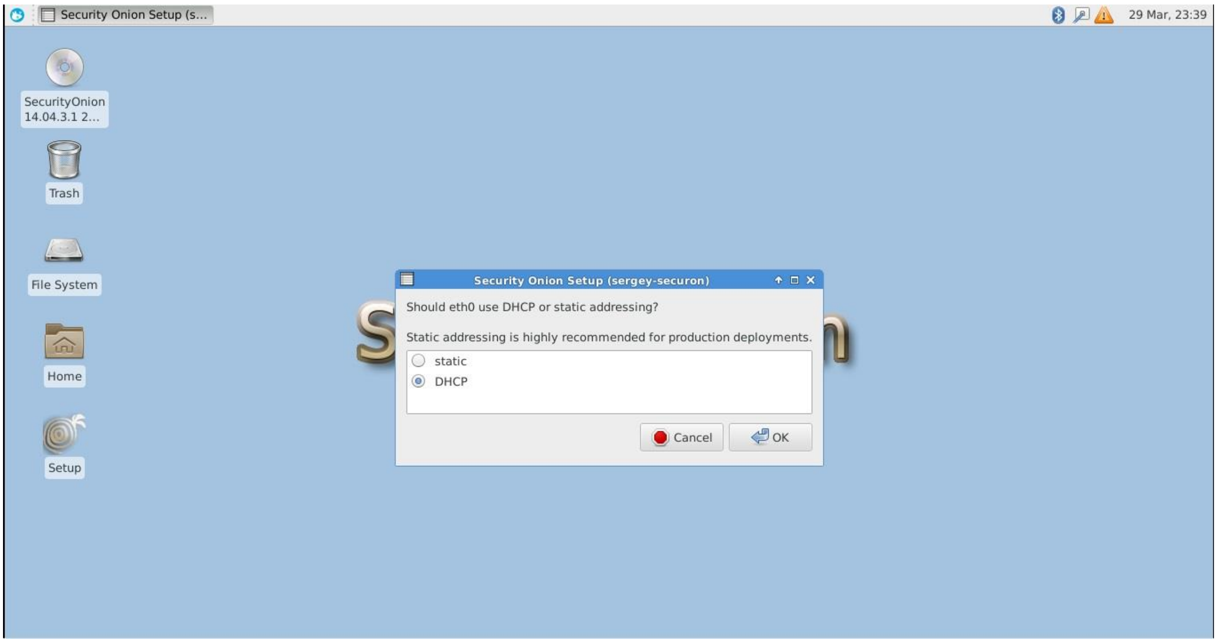
4. If the user has few interfaces, he can choose the management interface. As I have only one interface I got this dialog box shown in Figure 11 saying that your default interface has been chosen as management one. So if the user has one interface it will be monitored by default.


Security Onion Setup (s...
29 Mar, 23:31
SecurityOnion 14.04.3.1 2...
Trash
File System
Home
Setup
Security Onion Setup (sergey-securon)
OK

You only have one interface (eth0), which will be configured as a management interface.

FIGURE 11. Notification

5. Next, the user has to configure the interface for static IP or DHCP (Figure 12)


Security Onion Setup (s...
29 Mar, 23:39
SecurityOnion 14.04.3.1 2...
Trash
File System
Home
Setup
Security Onion Setup (sergey-securon)
Cancel
OK

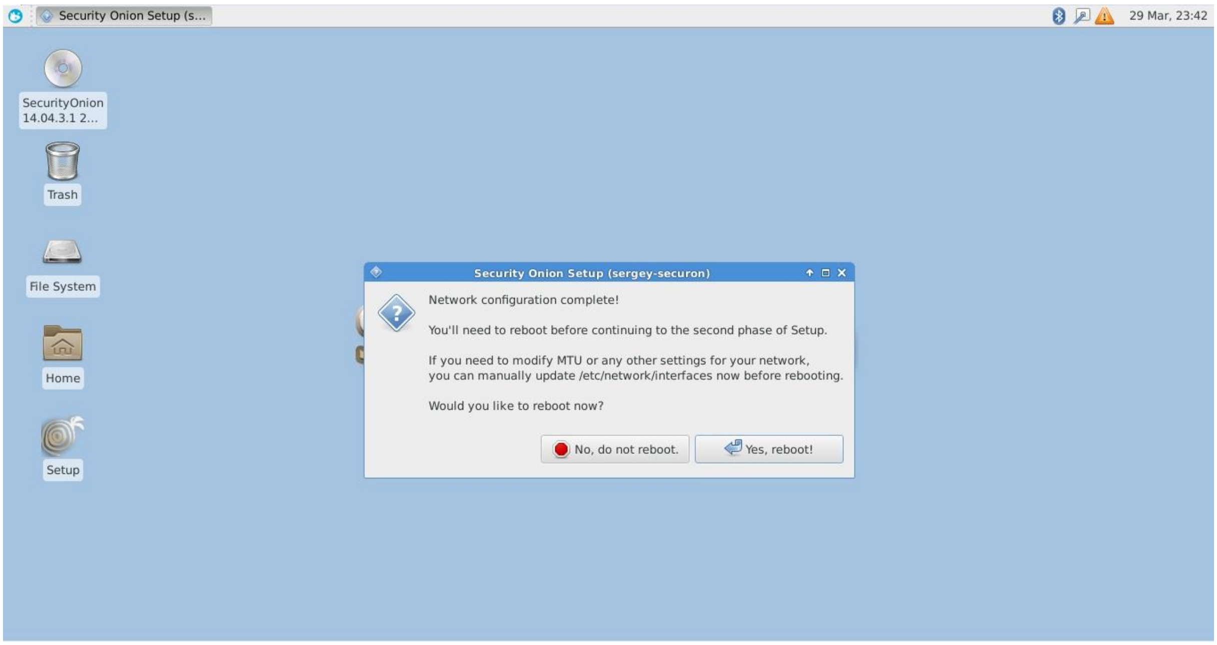
Should eth0 use DHCP or static addressing?

Static addressing is highly recommended for production deployments.

○ static
● DHCP

FIGURE 12. Setting of addressing

6. System will ask to confirm the changes and to reboot (Figure 13).


Security Onion Setup (s...
29 Mar, 23:42
SecurityOnion 14.04.3.1 2...
Trash
File System
Home
Setup
Security Onion Setup (sergey-securon)
No, do not reboot.
Yes, reboot!

Network configuration complete!

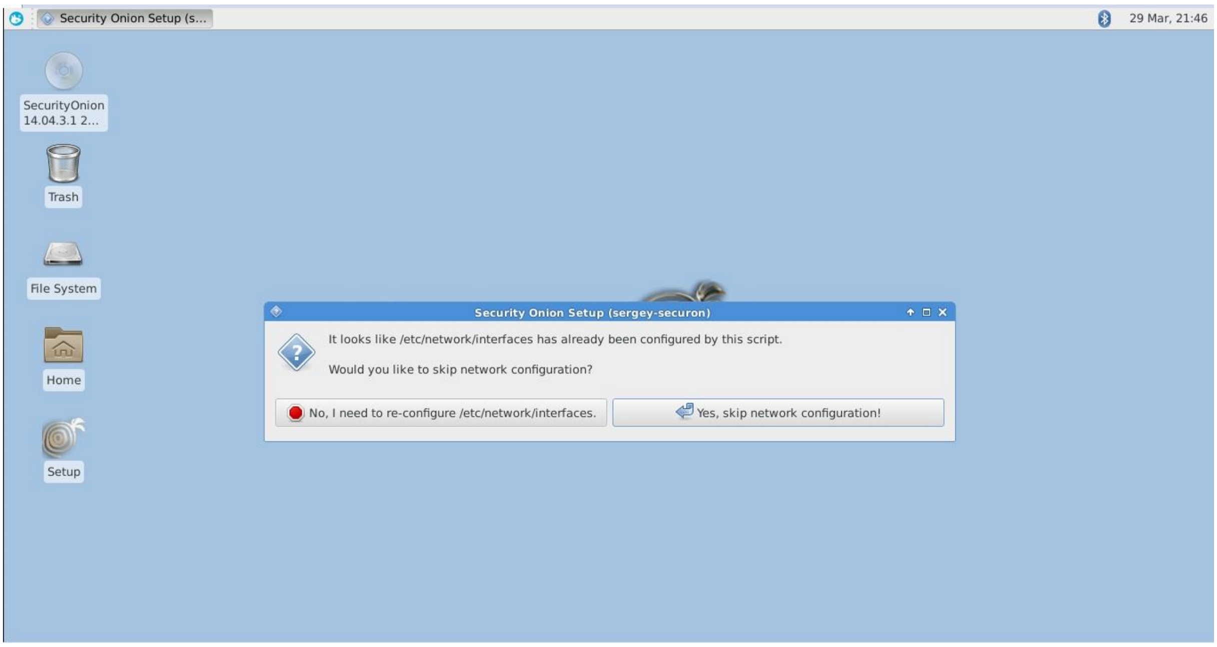
You'll need to reboot before continuing to the second phase of Setup.

If you need to modify MTU or any other settings for your network,
you can manually update /etc/network/interfaces now before rebooting.

Would you like to reboot now?

FIGURE 13. Confirmation of settings

7. After rebooting the user needs to go to Setup again and to skip network configuration or to make some changes as shown in Figure 14.

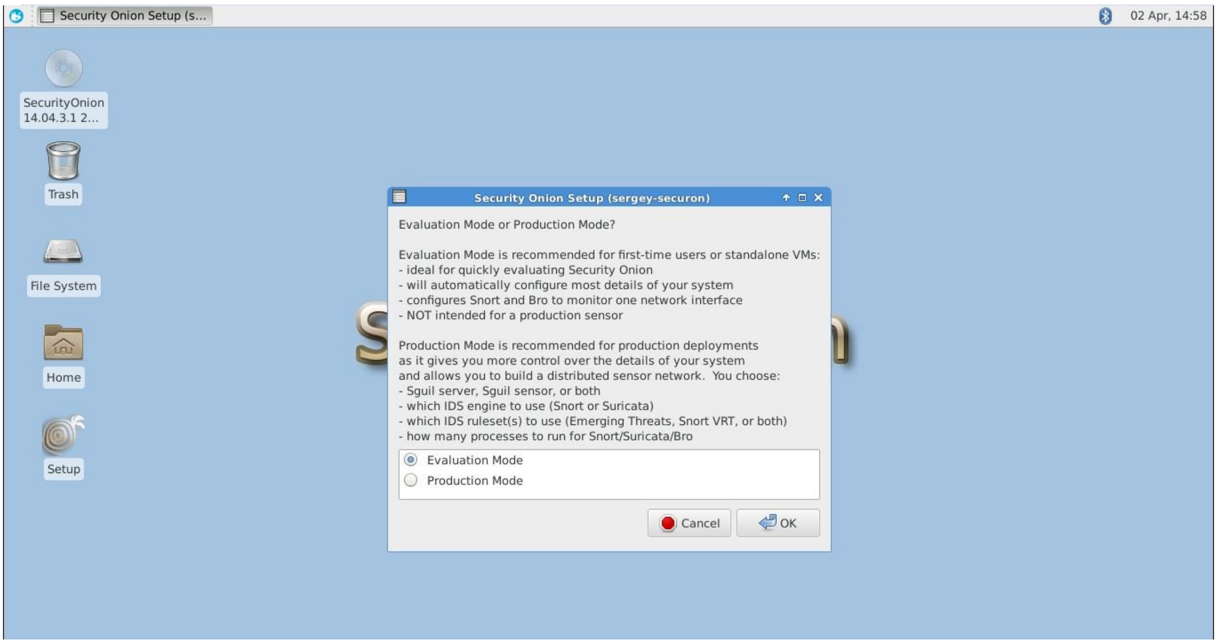

Security Onion Setup (s...
29 Mar, 21:46
SecurityOnion 14.04.3.1 2...
Trash
File System
Home
Setup
Security Onion Setup (sergey-securon)
No, I need to re-configure /etc/network/interfaces.
Yes, skip network configuration!

It looks like /etc/network/interfaces has already been configured by this script.

Would you like to skip network configuration?

FIGURE 14. Possibility to reconfigure network interfaces

8. Now the user can choose the way of installation (quick setup and advanced one), features of each one are shown in Figure 15.


Security Onion Setup (s...
02 Apr, 14:58
SecurityOnion 14.04.3.1 2...
Trash
File System
Home
Setup
Security Onion Setup (sergey-securon)
Cancel
OK

Evaluation Mode or Production Mode?

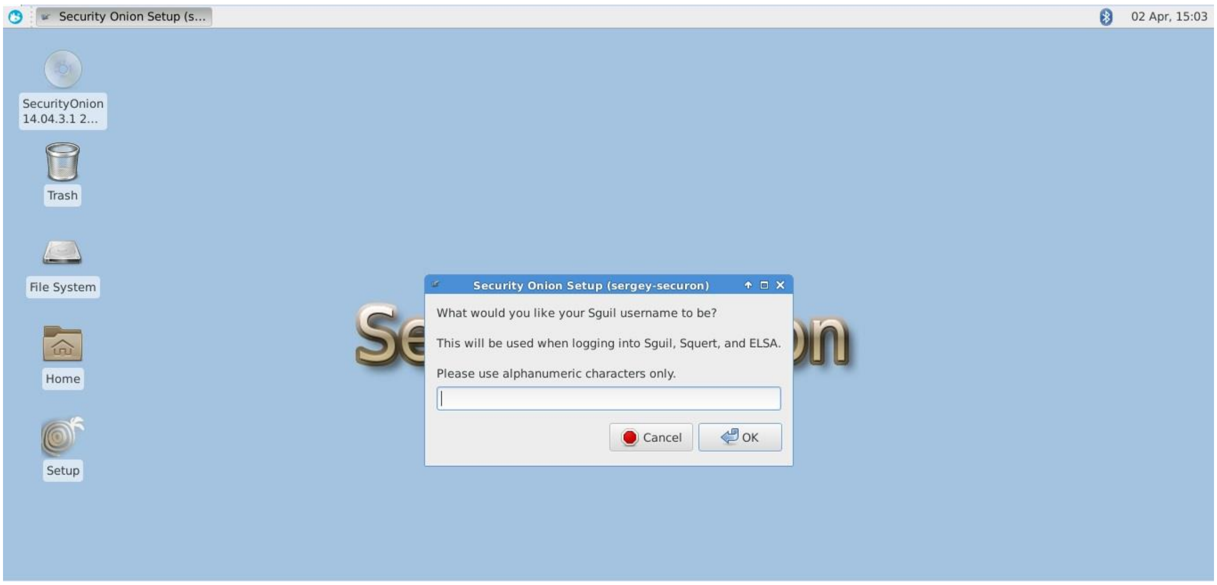
Evaluation Mode is recommended for first-time users or standalone VMs:
- ideal for quickly evaluating Security Onion
- will automatically configure most details of your system
- configures Snort and Bro to monitor one network interface
- NOT intended for a production sensor

Production Mode is recommended for production deployments as it gives you more control over the details of your system and allows you to build a distributed sensor network. You choose:
- Sguil server, Sguil sensor, or both
- which IDS engine to use (Snort or Suricata)
- which IDS ruleset(s) to use (Emerging Threats, Snort VRT, or both)
- how many processes to run for Snort/Suricata/Bro

● Evaluation Mode
○ Production Mode

FIGURE 15. Mode choosing

9. After this the user can setup the username and password for Sguil (Figure 16).


Security Onion Setup (s...
02 Apr, 15:03
SecurityOnion 14.04.3.1 2...
Trash
File System
Home
Setup
Security Onion Setup (sergey-securon)

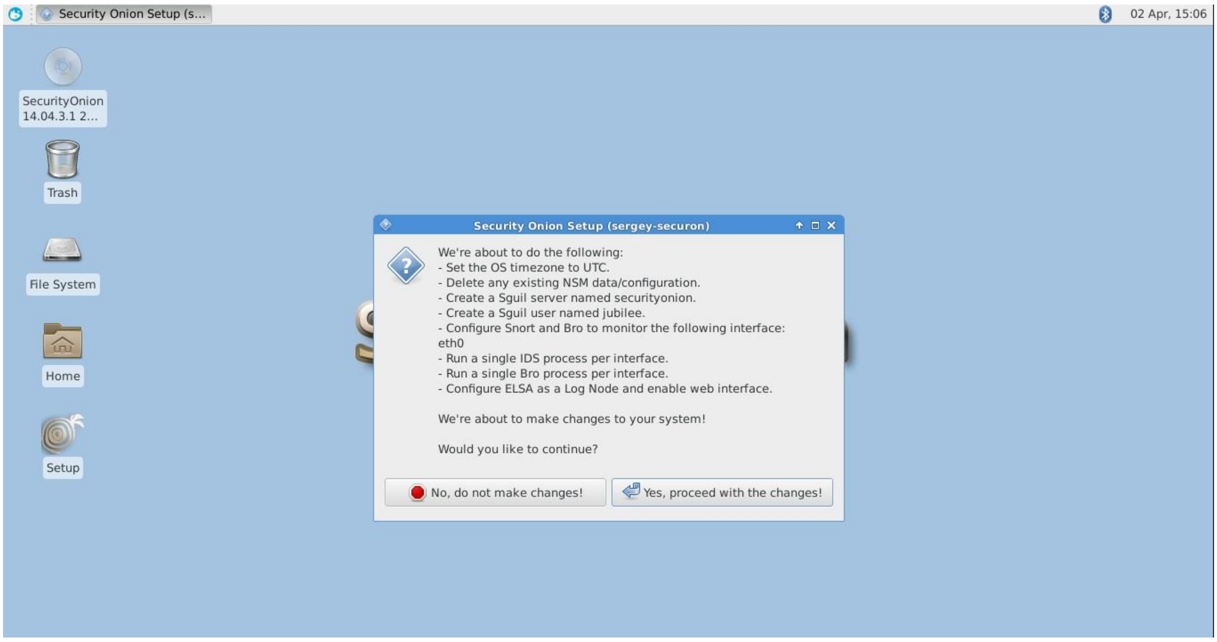
What would you like your Sguil username to be?

This will be used when logging into Sguil, Squert, and ELSA.

Please use alphanumeric characters only.

FIGURE 16. Username and password choosing

10. This is the last step, if the user has only one interface and he will get a notification are shown in Figures 17 and 18.

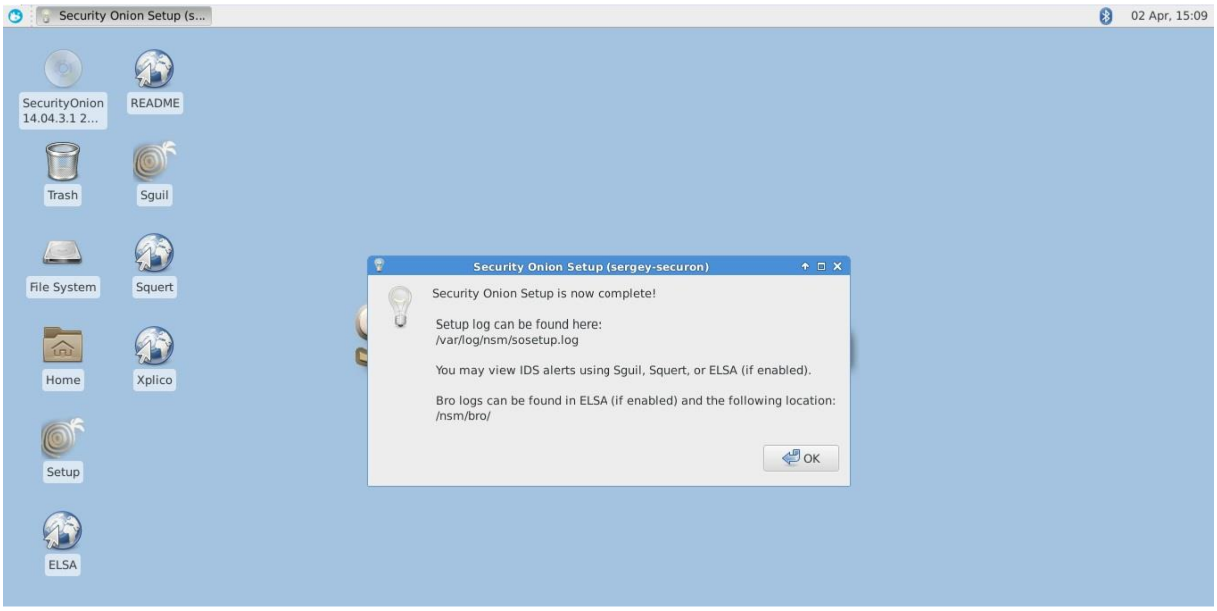

Security Onion Setup (s...
02 Apr, 15:06
SecurityOnion 14.04.3.1 2...
Trash
File System
Home
Setup
Security Onion Setup (sergey-securon)

We're about to do the following:
- Set the OS timezone to UTC.
- Delete any existing NSM data/configuration.
- Create a Sguil server named securityonion.
- Create a Sguil user named jubilee.
- Configure Snort and Bro to monitor the following interface:
eth0
- Run a single IDS process per interface.
- Run a single Bro process per interface.
- Configure ELSA as a Log Node and enable web interface.

We're about to make changes to your system!

Would you like to continue?

FIGURE 17. Confirmation


Security Onion Setup (s...
02 Apr, 15:09
SecurityOnion 14.04.3.1 2...
README
Trash
Sguil
File System
Squert
Home
Xplico
Setup
ELSA
Security Onion Setup (sergey-securon)
OK

Security Onion Setup is now complete!

Setup log can be found here:
/var/log/nsm/sosetup.log

You may view IDS alerts using Sguil, Squert, or ELSA (if enabled).

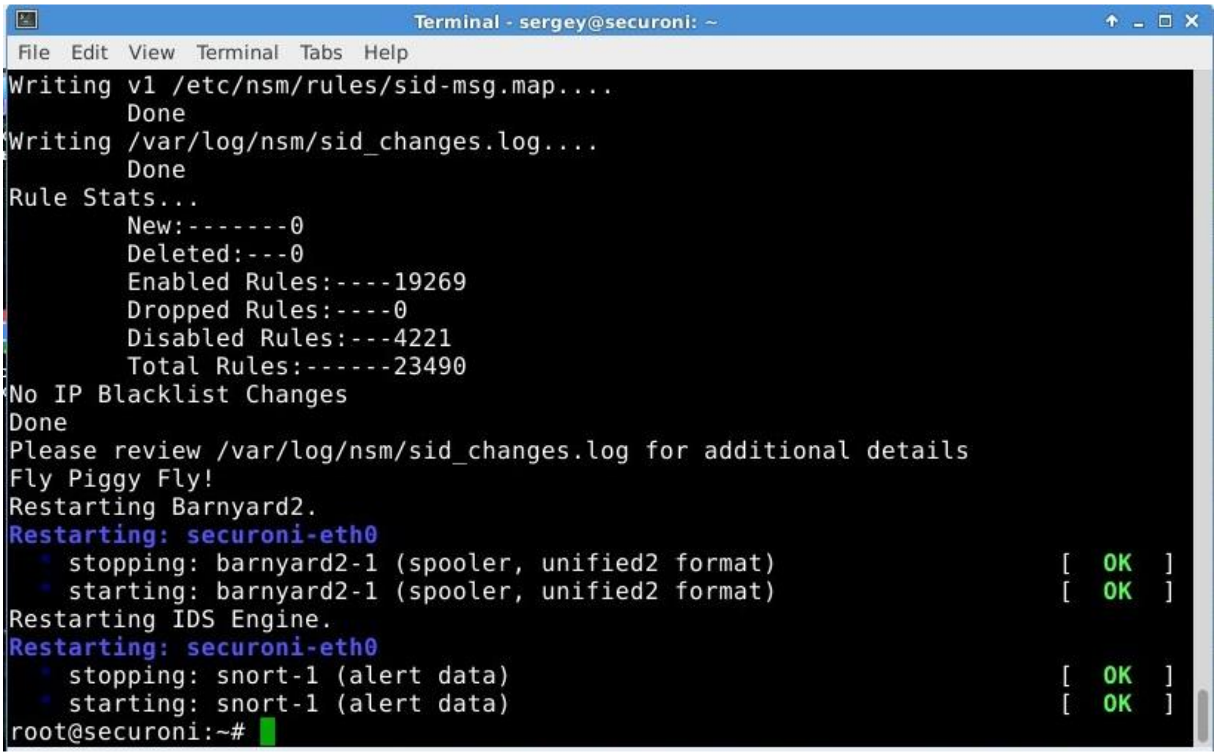
Bro logs can be found in ELSA (if enabled) and the following location:
/nsm/bro/

FIGURE 18. Notification of completed installation

Now Security Onion has been installed, and the user has to update Snort rules using the following command:

usr/bin/sudo-update

This command should be executed with root rights to update the rules. Figure 19 shows the output of the command.



```
Writing v1 /etc/nsm/rules/sid-msg.map....
        Done
Writing /var/log/nsm/sid_changes.log....
        Done
Rule Stats...
        New:-------0
        Deleted:---0
        Enabled Rules:----19269
        Dropped Rules:----0
        Disabled Rules:---4221
        Total Rules:------23490
No IP Blacklist Changes
Done
Please review /var/log/nsm/sid_changes.log for additional details
Fly Piggy Fly!
Restarting Barnyard2.
Restarting: securoni-eth0
    stopping: barnyard2-1 (spooler, unified2 format)                  [  OK  ]
    starting: barnyard2-1 (spooler, unified2 format)                  [  OK  ]
Restarting IDS Engine.
Restarting: securoni-eth0
    stopping: snort-1 (alert data)                                    [  OK  ]
    starting: snort-1 (alert data)                                    [  OK  ]
root@securoni:~#
```

FIGURE 19. Rule updates

3.3 Testing the system

3.3.1 Run as IPS

After installation Security onion is already running as IDS and it has the basic rules from Snort. Basically running Security Onion as an IPS requires manual configuration and is not supported “from the box”. (Security Onion Solutions 2016). To test it as IPS some IPS rules need to be created.

A good way to test that Snort works properly and works as IPS is to create an easy rule such as the following:

1. Customize the rule set by adding **include $RULE_PATH/test.rules** to the **/etc/nsm/securoni-eth0/snort.conf** through root rights
2. Create **test.rules** file in **/etc/nsm/rules/**
3. Type in the file **reject tcp any any -> any any (msg:"Test of Snort qwerty123"; content:"qwerty"; classtype:shellcode-detect; sid:310; rev:1;)**-here particular networks can be specified, but as there is no other network for simplifying, we can chose ”Any”. This rule says that in case of detecting incoming tcp packets with subline **qwerty123**, then to the source will be sent RST and the session will be droped.
4. Save and update rules by **/usr/bin/rule-update**

Now this IPS rule is running. To test it via NetCat has been send query **qwerty123** to our defending port. Here is the output of log file **/var/log/nsm/securoni-eth0/alert/** :

```
[**] [1:310:1] Test of Snort qwerty123 [**]
[Classification: Executable Code was Detected] [Priority: 1]
04/07-12:03:12.155213136 172.16.249.1:56473 ->
172.16.249.130:8080
TCP TTL:64 TOS:0x0 ID:1241 IpLen:20 DgmLen:59 DF
***AP*** Seq: 0x9510F391 Ack: 0xC40C0E14 Win: 0x8218 TcpLen:
32
TCP Options (3) => NOP NOP TS: 125531844 9470333
```

And this function can be done in another way. The easy way is to the check event rule by Sguil that we want to block. And cut this rule from the shown rule file and paste to **test.rules**, where change **alert** to **reject/drop.** But do not forget to remove this rule from first rule file, otherwise it will cause an error.

To use .rules and .conf files in the GUI mode the user has to install additional software, in case of this project it is **gedit**.

3.3.2 Rules modification

Now, let’s assume, that the user uses BitTorrent (here can be any program, connection or any things exist in the current rules) and wont to get any alerts related to P2P BitTorrent connections and BitTorrent file downloads.

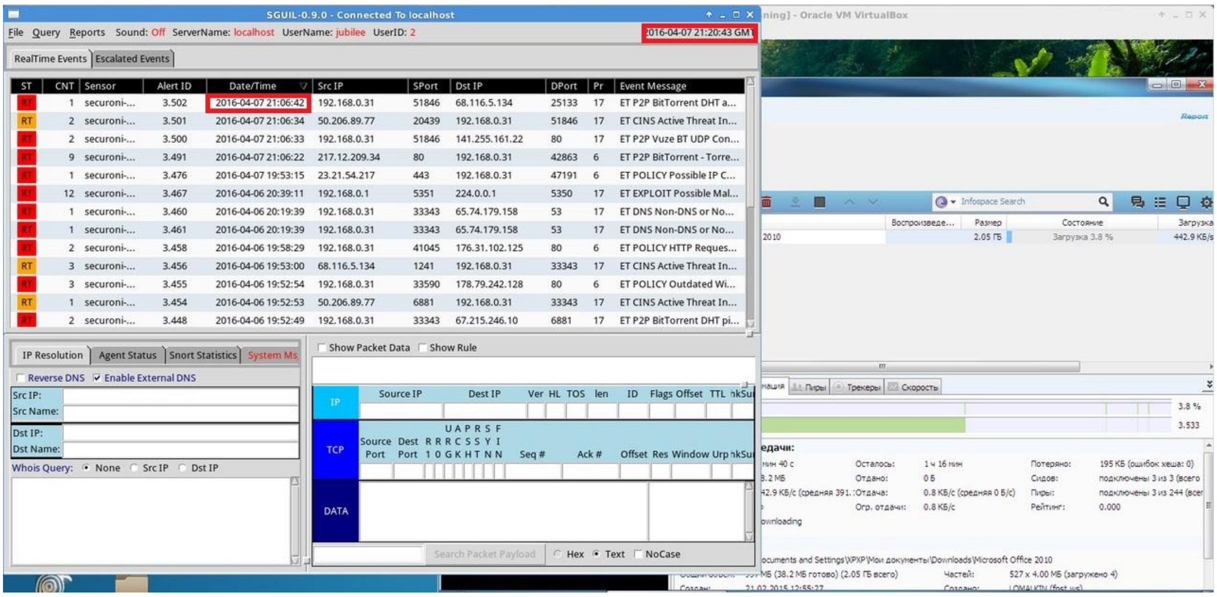
The user has to disable any rules related to it by the command in **disablesid.conf**, which allocates in **/etc/nsm/pulledpork**.

Configuration instruction:

1. Open Terminal.
2. Type **sudo -i** to get root rights.
3. Type **gedit /ets/nsm/pulledpork/disablesid.conf .**
4. Add a line in the opened file **pcre:BitTorrent .**
5. Save file and update rules by **/usr/bin/rule-update.**
6. Type **/usr/bin/rule-update** in the terminal to update rules for Snort.
7. Restart analyze program (Sguil).

This feature can be used in different ways. This example is just one of many.

Figure 20 shows that it does not alert BitTorrent actions anymore (check the system time and alerts).



Sound: Off ServerName: localhost UserName: jubilee UserID: 2 2016-04-07 21:20:43 GMT

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	1	securoni-...	3.502	2016-04-07 21:06:42	192.168.0.31	51846	68.116.5.134	25133	17	ET P2P BitTorrent DHT a...
RT	2	securoni-...	3.501	2016-04-07 21:06:34	50.206.89.77	20439	192.168.0.31	51846	17	ET CINS Active Threat In...
RT	2	securoni-...	3.500	2016-04-07 21:06:33	192.168.0.31	51846	141.255.161.22	80	17	ET P2P Vuze BT UDP Con...
RT	9	securoni-...	3.491	2016-04-07 21:06:22	217.12.209.34	80	192.168.0.31	42863	6	ET P2P BitTorrent - Torre...
RT	1	securoni-...	3.476	2016-04-07 19:53:15	23.21.54.217	443	192.168.0.31	47191	6	ET POLICY Possible IP C...
RT	12	securoni-...	3.467	2016-04-06 20:39:11	192.168.0.1	5351	224.0.0.1	5350	17	ET EXPLOIT Possible Mal...
RT	1	securoni-...	3.460	2016-04-06 20:19:39	192.168.0.31	33343	65.74.179.158	53	17	ET DNS Non-DNS or No...
RT	1	securoni-...	3.461	2016-04-06 20:19:39	192.168.0.31	33343	65.74.179.158	53	17	ET DNS Non-DNS or No...
RT	2	securoni-...	3.458	2016-04-06 19:58:29	192.168.0.31	41045	176.31.102.125	80	6	ET POLICY HTTP Reques...
RT	3	securoni-...	3.456	2016-04-06 19:53:00	68.116.5.134	1241	192.168.0.31	33343	17	ET CINS Active Threat In...
RT	3	securoni-...	3.455	2016-04-06 19:52:54	192.168.0.31	33590	178.79.242.128	80	6	ET POLICY Outdated Wi...
RT	1	securoni-...	3.454	2016-04-06 19:52:53	50.206.89.77	6881	192.168.0.31	33343	17	ET CINS Active Threat In...
RT	2	securoni-...	3.448	2016-04-06 19:52:49	192.168.0.31	33343	67.215.246.10	6881	17	ET P2P BitTorrent DHT pi...

☐ Reverse DNS ☑ Enable External DNS

Src IP:

Src Name:

Dst IP:

Dst Name:

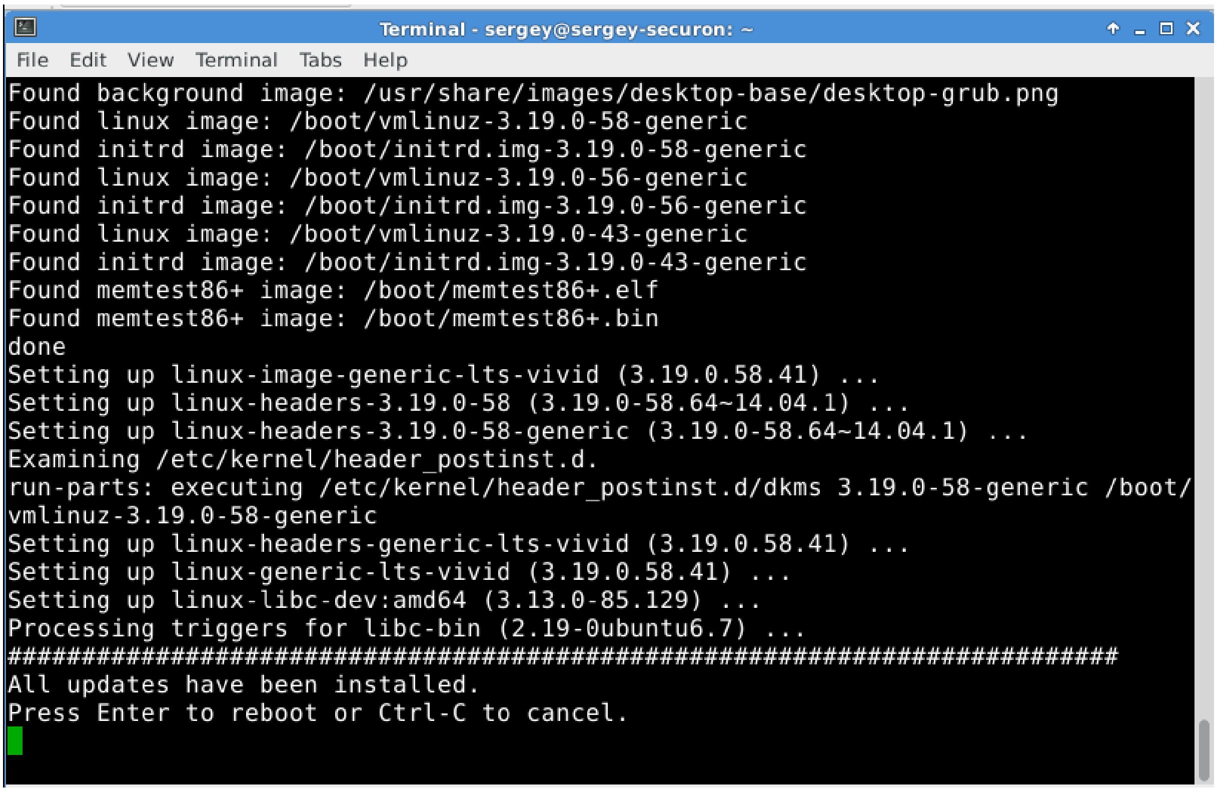
Whois Query: ● None ○ Src IP ○ Dst IP

FIGURE 20. Shows no alerts after new rules accepting while BitTorrent is running

3.3.3 Update

Security Onion and Ubuntu update is a simple thing. In the last version of Security Onion has a preinstalled tool **soup,** which means Security Onion Update.

The **sudo soup** command in the terminal installs all updates of the entire deployment (Figure 21).


Terminal - sergey@sergey-securon: ~
File Edit View Terminal Tabs Help

```
Found background image: /usr/share/images/desktop-base/desktop-grub.png
Found linux image: /boot/vmlinuz-3.19.0-58-generic
Found initrd image: /boot/initrd.img-3.19.0-58-generic
Found linux image: /boot/vmlinuz-3.19.0-56-generic
Found initrd image: /boot/initrd.img-3.19.0-56-generic
Found linux image: /boot/vmlinuz-3.19.0-43-generic
Found initrd image: /boot/initrd.img-3.19.0-43-generic
Found memtest86+ image: /boot/memtest86+.elf
Found memtest86+ image: /boot/memtest86+.bin
done
Setting up linux-image-generic-lts-vivid (3.19.0.58.41) ...
Setting up linux-headers-3.19.0-58 (3.19.0-58.64~14.04.1) ...
Setting up linux-headers-3.19.0-58-generic (3.19.0-58.64~14.04.1) ...
Examining /etc/kernel/header_postinst.d.
run-parts: executing /etc/kernel/header_postinst.d/dkms 3.19.0-58-generic /boot/
vmlinuz-3.19.0-58-generic
Setting up linux-headers-generic-lts-vivid (3.19.0.58.41) ...
Setting up linux-generic-lts-vivid (3.19.0.58.41) ...
Setting up linux-libc-dev:amd64 (3.13.0-85.129) ...
Processing triggers for libc-bin (2.19-0ubuntu6.7) ...
############################################################################
All updates have been installed.
Press Enter to reboot or Ctrl-C to cancel.
```

FIGURE 21. Security Onion updating

3.3.4 Configuration Backup

Security Onion does not have preinstalled backup software for backup service. It has only rule backups when the user run **/usr/bin/rule-setup.** To get configuration backups the user needs to use other software such as **rdiff-backup**.

To make full backup of configurations:

1. Run terminal.
2. Get root rights for example **sudo –I.**
3. Install rdiff-backup **apt-get install rdiff-backup.**
4. Make backup of files/directories needed, for example **rdiff-backup -v 4 /etc/nsm/ /etc/nsmbackup .**

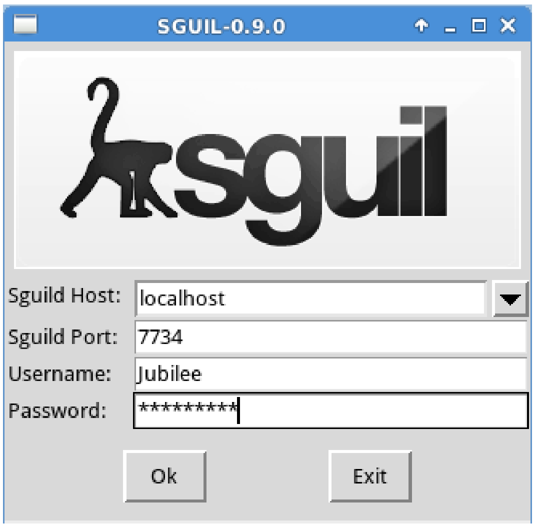
Now the backup is completed

3.4 Environment to work with Snort

The Security Onion distribution has preinstalled software to work with IDS systems. In case of this thesis Sguil has been used.

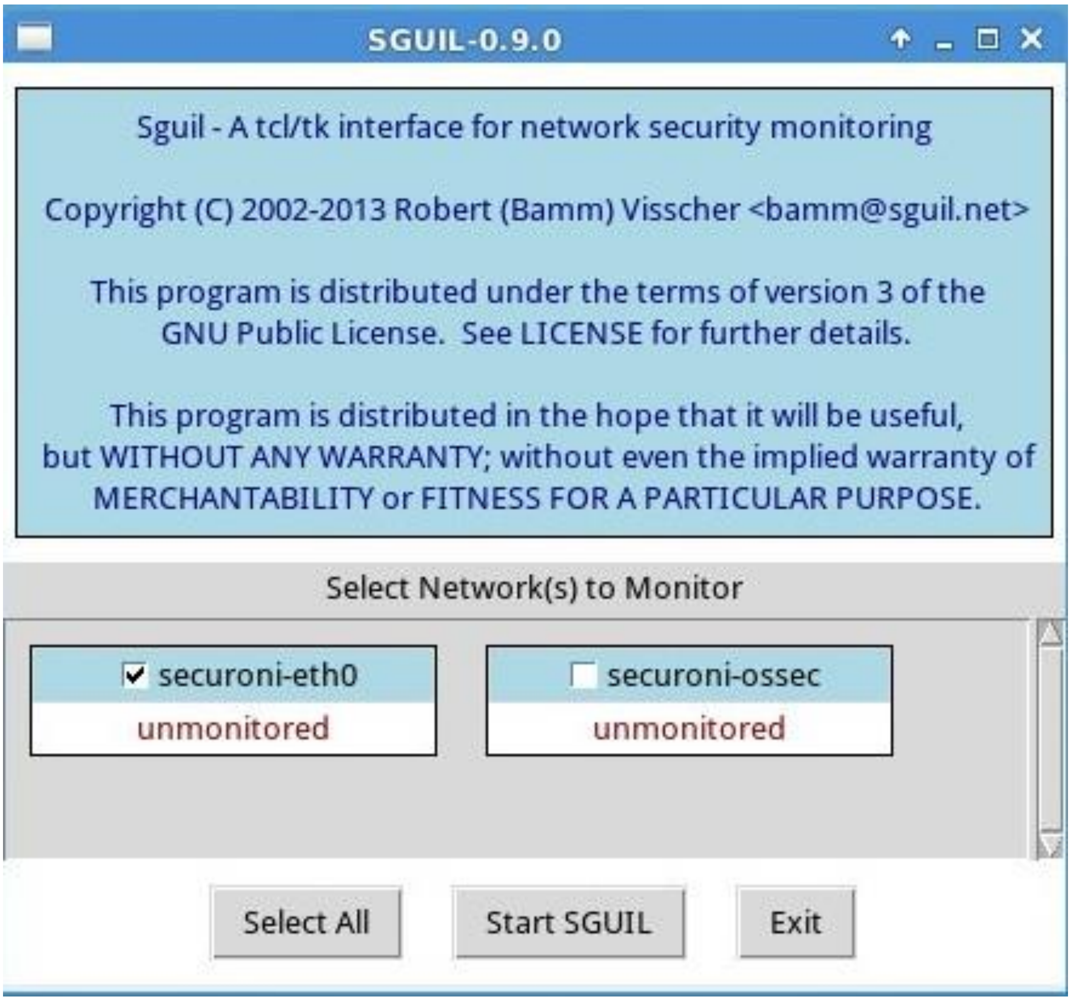
Sguil is a software based program (most event analysis programs are browser based) and has a lot of useful features. Moreover, it is a real time tool. It logs events and provides wide-function analysis. All events mark in way of their dangerous for the system.

Work with Sguil starts from the login to the program and from choosing monitoring network (Figures.22 and 23).



Sguild Host: localhost
Sguild Port: 7734
Username: Jubilee
Password: *********

FIGURE 22. Sguil start



Sguil - A tcl/tk interface for network security monitoring

Copyright (C) 2002-2013 Robert (Bamm) Visscher <bamm@sguil.net>

This program is distributed under the terms of version 3 of the GNU Public License. See LICENSE for further details.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

Select Network(s) to Monitor

☑ securoni-eth0
unmonitored

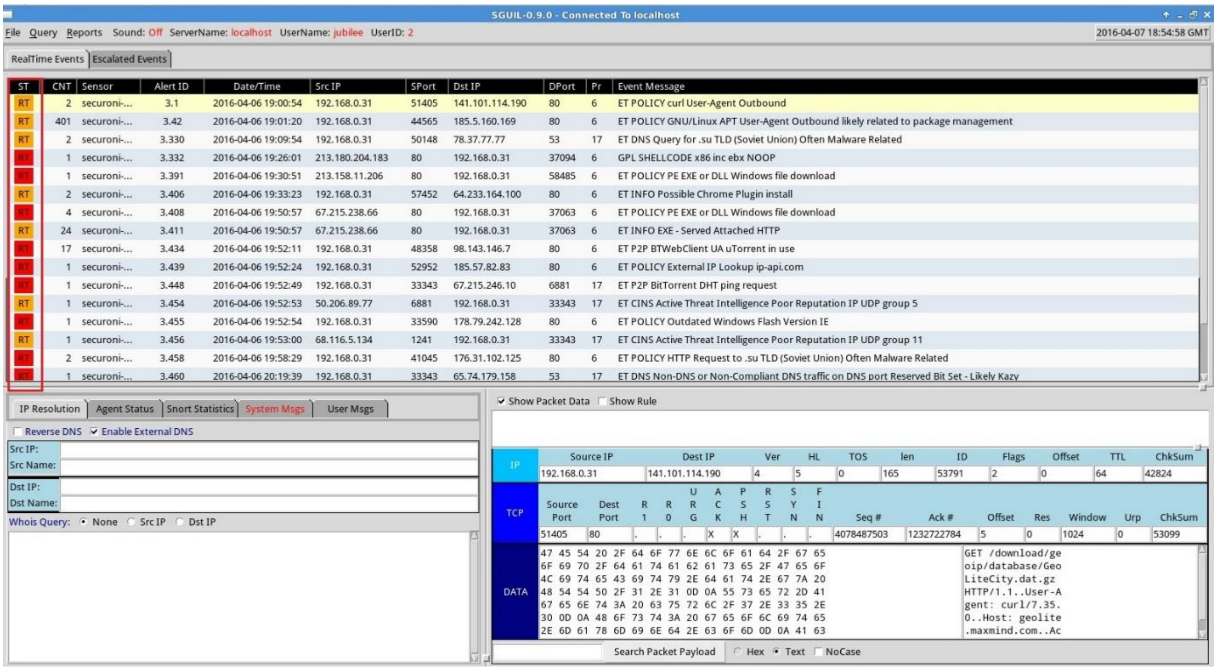
☐ securoni-ossec
unmonitored

FIGURE 23. Network selection

Sguil is a user friendly environment and intuitively understandable.
It has seven categories of alerts (Sguil 2016):

Category I	Unauthorized Root/Admin Access
Category II	Unauthorized User Access
Category III	Attempted Unauthorized Access
Category IV	Successful Denial of Service Attack
Category V	Poor Security Practice or Policy Violation
Category VI	Reconnaissance/Probes/Scans
Category VII	Virus Infection

Potential danger of the event is indicated with different color (Figure.24).



Sound: Off ServerName: localhost UserName: jubilee UserID: 2

2016-04-07 18:54:58 GMT

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	2	securoni-...	3.1	2016-04-06 19:00:54	192.168.0.31	51405	141.101.114.190	80	6	ET POLICY curl User-Agent Outbound
RT	401	securoni-...	3.42	2016-04-06 19:01:20	192.168.0.31	44565	185.5.160.169	80	6	ET POLICY GNU/Linux APT User-Agent Outbound likely related to package management
RT	2	securoni-...	3.330	2016-04-06 19:09:54	192.168.0.31	50148	78.37.77.77	53	17	ET DNS Query for .su TLD (Soviet Union) Often Malware Related
RT	1	securoni-...	3.332	2016-04-06 19:26:01	213.180.204.183	80	192.168.0.31	37094	6	GPL SHELLCODE x86 inc ebx NOOP
RT	1	securoni-...	3.391	2016-04-06 19:30:51	213.158.11.206	80	192.168.0.31	58485	6	ET POLICY PE EXE or DLL Windows file download
RT	2	securoni-...	3.406	2016-04-06 19:33:23	192.168.0.31	57452	64.233.164.100	80	6	ET INFO Possible Chrome Plugin install
RT	4	securoni-...	3.408	2016-04-06 19:50:57	67.215.238.66	80	192.168.0.31	37063	6	ET POLICY PE EXE or DLL Windows file download
RT	24	securoni-...	3.411	2016-04-06 19:50:57	67.215.238.66	80	192.168.0.31	37063	6	ET INFO EXE - Served Attached HTTP
RT	17	securoni-...	3.434	2016-04-06 19:52:11	192.168.0.31	48358	98.143.146.7	80	6	ET P2P BTWebClient UA uTorrent in use
RT	1	securoni-...	3.439	2016-04-06 19:52:24	192.168.0.31	52952	185.57.82.83	80	6	ET POLICY External IP Lookup ip-api.com
RT	1	securoni-...	3.448	2016-04-06 19:52:49	192.168.0.31	33343	67.215.246.10	6881	17	ET P2P BitTorrent DHT ping request
RT	1	securoni-...	3.454	2016-04-06 19:52:53	50.206.89.77	6881	192.168.0.31	33343	17	ET CINS Active Threat Intelligence Poor Reputation IP UDP group 5
RT	1	securoni-...	3.455	2016-04-06 19:52:54	192.168.0.31	33590	178.79.242.128	80	6	ET POLICY Outdated Windows Flash Version IE
RT	1	securoni-...	3.456	2016-04-06 19:53:00	68.116.5.134	1241	192.168.0.31	33343	17	ET CINS Active Threat Intelligence Poor Reputation IP UDP group 11
RT	2	securoni-...	3.458	2016-04-06 19:58:29	192.168.0.31	41045	176.31.102.125	80	6	ET POLICY HTTP Request to .su TLD (Soviet Union) Often Malware Related
RT	1	securoni-...	3.460	2016-04-06 20:19:39	192.168.0.31	33343	65.74.179.158	53	17	ET DNS Non-DNS or Non-Compliant DNS traffic on DNS port Reserved Bit Set - Likely Kazy

☐ Reverse DNS ☑ Enable External DNS

Whois Query: ● None ○ Src IP ○ Dst IP

☑ Show Packet Data ☐ Show Rule

Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags	Offset	TTL	ChkSum
192.168.0.31	141.101.114.190	4	5	0	165	53791	2	0	64	42824

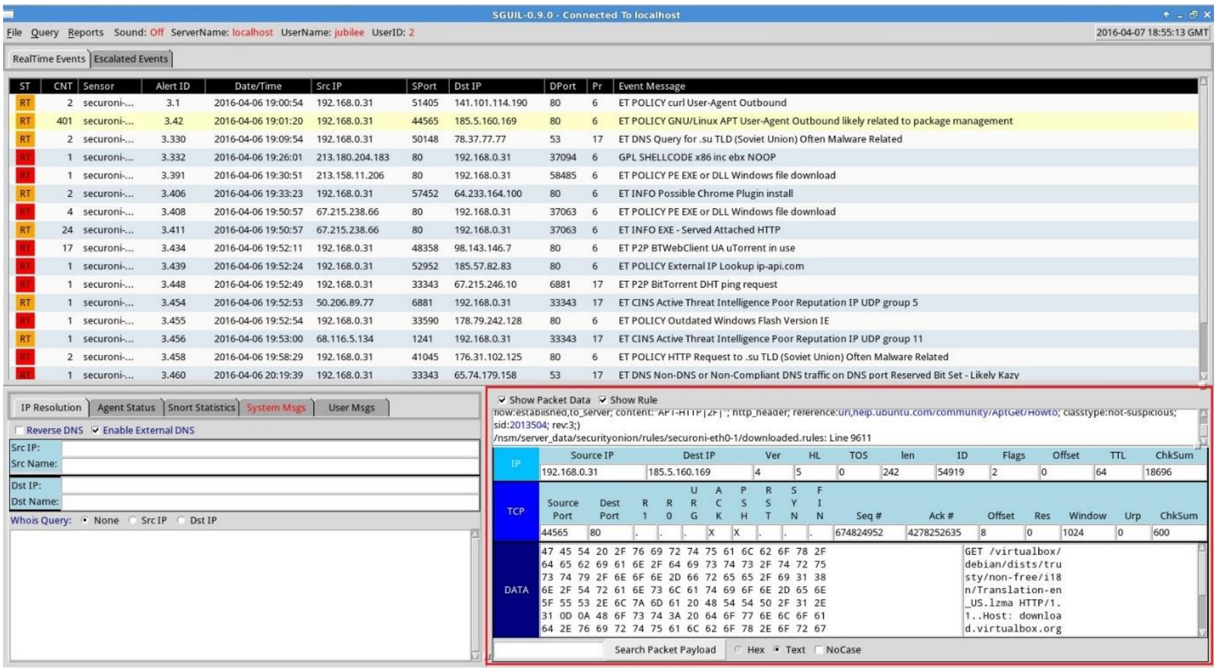
Source Port	Dest Port	R 1	R 0	URG	ACK	PSH	RST	SYN	FIN	Seq #	Ack #	Offset	Res	Window	Urp	ChkSum
51405	80	.	.	.	X	X	.	.	.	4078487503	1232722784	5	0	1024	0	53099

```
47 45 54 20 2F 64 6F 77 6E 6C 6F 61 64 2F 67 65      GET /download/ge
6F 69 70 2F 64 61 74 61 62 61 73 65 2F 47 65 6F      oip/database/Geo
4C 69 74 65 43 69 74 79 2E 64 61 74 2E 67 7A 20      LiteCity.dat.gz
48 54 54 50 2F 31 2E 31 0D 0A 55 73 65 72 2D 41      HTTP/1.1..User-A
67 65 6E 74 3A 20 63 75 72 6C 2F 37 2E 33 35 2E      gent: curl/7.35.
30 0D 0A 48 6F 73 74 3A 20 67 65 6F 6C 69 74 65      0..Host: geolite
2E 6D 61 78 6D 69 6E 64 2E 63 6F 6D 0D 0A 41 63      .maxmind.com..Ac
```

○ Hex ● Text ☐ NoCase

FIGURE 24. Color indicators

The right lower corner a window which shows packet data and rules that has been used for a particular event (Figure 25). Opportunity to check the rules is helpful for creating or modifying existing rules for particular events.



Sound: Off ServerName: localhost UserName: jubilee UserID: 2

2016-04-07 18:55:13 GMT

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	2	securoni-...	3.1	2016-04-06 19:00:54	192.168.0.31	51405	141.101.114.190	80	6	ET POLICY curl User-Agent Outbound
RT	401	securoni-...	3.42	2016-04-06 19:01:20	192.168.0.31	44565	185.5.160.169	80	6	ET POLICY GNU/Linux APT User-Agent Outbound likely related to package management
RT	2	securoni-...	3.330	2016-04-06 19:09:54	192.168.0.31	50148	78.37.77.77	53	17	ET DNS Query for .su TLD (Soviet Union) Often Malware Related
RT	1	securoni-...	3.332	2016-04-06 19:26:01	213.180.204.183	80	192.168.0.31	37094	6	GPL SHELLCODE x86 inc ebx NOOP
RT	1	securoni-...	3.391	2016-04-06 19:30:51	213.158.11.206	80	192.168.0.31	58485	6	ET POLICY PE EXE or DLL Windows file download
RT	2	securoni-...	3.406	2016-04-06 19:33:23	192.168.0.31	57452	64.233.164.100	80	6	ET INFO Possible Chrome Plugin install
RT	4	securoni-...	3.408	2016-04-06 19:50:57	67.215.238.66	80	192.168.0.31	37063	6	ET POLICY PE EXE or DLL Windows file download
RT	24	securoni-...	3.411	2016-04-06 19:50:57	67.215.238.66	80	192.168.0.31	37063	6	ET INFO EXE - Served Attached HTTP
RT	17	securoni-...	3.434	2016-04-06 19:52:11	192.168.0.31	48358	98.143.146.7	80	6	ET P2P BTWebClient UA uTorrent in use
RT	1	securoni-...	3.439	2016-04-06 19:52:24	192.168.0.31	52952	185.57.82.83	80	6	ET POLICY External IP Lookup ip-api.com
RT	1	securoni-...	3.448	2016-04-06 19:52:49	192.168.0.31	33343	67.215.246.10	6881	17	ET P2P BitTorrent DHT ping request
RT	1	securoni-...	3.454	2016-04-06 19:52:53	50.206.89.77	6881	192.168.0.31	33343	17	ET CINS Active Threat Intelligence Poor Reputation IP UDP group 5
RT	1	securoni-...	3.455	2016-04-06 19:52:54	192.168.0.31	33590	178.79.242.128	80	6	ET POLICY Outdated Windows Flash Version IE
RT	1	securoni-...	3.456	2016-04-06 19:53:00	68.116.5.134	1241	192.168.0.31	33343	17	ET CINS Active Threat Intelligence Poor Reputation IP UDP group 11
RT	2	securoni-...	3.458	2016-04-06 19:58:29	192.168.0.31	41045	176.31.102.125	80	6	ET POLICY HTTP Request to .su TLD (Soviet Union) Often Malware Related
RT	1	securoni-...	3.460	2016-04-06 20:19:39	192.168.0.31	33343	65.74.179.158	53	17	ET DNS Non-DNS or Non-Compliant DNS traffic on DNS port Reserved Bit Set - Likely Kazy

☐ Reverse DNS ☑ Enable External DNS

Whois Query: ● None ○ Src IP ○ Dst IP

☑ Show Packet Data ☑ Show Rule

```
flow:established,to_server; content:"APT-HTTP|2F|"; http_header; reference:url,help.ubuntu.com/community/AptGet/Howto; classtype:not-suspicious;
sid:2013504; rev:3;)
/nsm/server_data/securityonion/rules/securoni-eth0-1/downloaded.rules: Line 9611
```

Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags	Offset	TTL	ChkSum
192.168.0.31	185.5.160.169	4	5	0	242	54919	2	0	64	18696

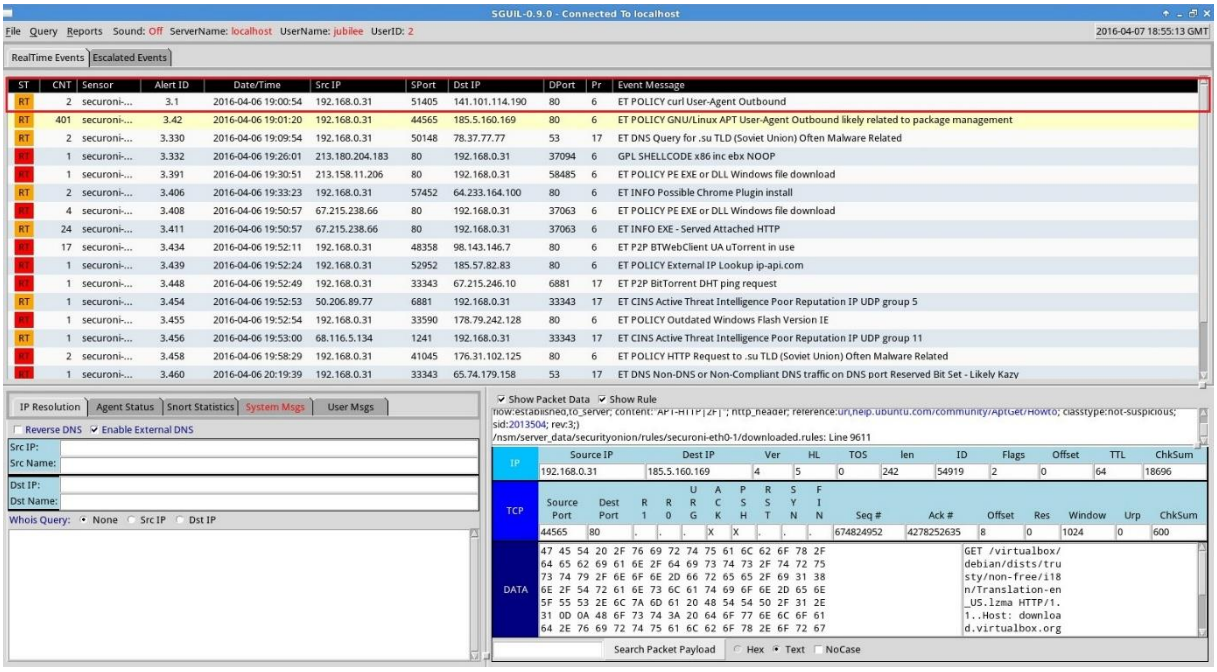
Source Port	Dest Port	R 1	R 0	URG	ACK	PSH	RST	SYN	FIN	Seq #	Ack #	Offset	Res	Window	Urp	ChkSum
44565	80	.	.	.	X	X	.	.	.	674824952	4278252635	8	0	1024	0	600

```
47 45 54 20 2F 76 69 72 74 75 61 6C 62 6F 78 2F      GET /virtualbox/
64 65 62 69 61 6E 2F 64 69 73 74 73 2F 74 72 75      debian/dists/tru
73 74 79 2F 6E 6F 6E 2D 66 72 65 65 2F 69 31 38      sty/non-free/i18
6E 2F 54 72 61 6E 73 6C 61 74 69 6F 6E 2D 65 6E      n/Translation-en
5F 55 53 2E 6C 7A 6D 61 20 48 54 54 50 2F 31 2E      _US.lzma HTTP/1.
31 0D 0A 48 6F 73 74 3A 20 64 6F 77 6E 6C 6F 61      1..Host: downloa
64 2E 76 69 72 74 75 61 6C 62 6F 78 2E 6F 72 67      d.virtualbox.org
```

○ Hex ● Text ☐ NoCase

FIGURE 25. Packet data and rule window

The main window has two tabs: RealTime Events and Escalated Events. These tabs have the same fields and functions, for different alert types (Figure.26). The fields shown on the main screen: State, Count, used Sensor, Alert ID, Date and Time, Source and Destination IP, Source and Destination port, Priority and Event Message.



Sound: Off ServerName: localhost UserName: jubilee UserID: 2

2016-04-07 18:55:13 GMT

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	2	securoni-...	3.1	2016-04-06 19:00:54	192.168.0.31	51405	141.101.114.190	80	6	ET POLICY curl User-Agent Outbound
RT	401	securoni-...	3.42	2016-04-06 19:01:20	192.168.0.31	44565	185.5.160.169	80	6	ET POLICY GNU/Linux APT User-Agent Outbound likely related to package management
RT	2	securoni-...	3.330	2016-04-06 19:09:54	192.168.0.31	50148	78.37.77.77	53	17	ET DNS Query for .su TLD (Soviet Union) Often Malware Related
RT	1	securoni-...	3.332	2016-04-06 19:26:01	213.180.204.183	80	192.168.0.31	37094	6	GPL SHELLCODE x86 inc ebx NOOP
RT	1	securoni-...	3.391	2016-04-06 19:30:51	213.158.11.206	80	192.168.0.31	58485	6	ET POLICY PE EXE or DLL Windows file download
RT	2	securoni-...	3.406	2016-04-06 19:33:23	192.168.0.31	57452	64.233.164.100	80	6	ET INFO Possible Chrome Plugin install
RT	4	securoni-...	3.408	2016-04-06 19:50:57	67.215.238.66	80	192.168.0.31	37063	6	ET POLICY PE EXE or DLL Windows file download
RT	24	securoni-...	3.411	2016-04-06 19:50:57	67.215.238.66	80	192.168.0.31	37063	6	ET INFO EXE - Served Attached HTTP
RT	17	securoni-...	3.434	2016-04-06 19:52:11	192.168.0.31	48358	98.143.146.7	80	6	ET P2P BTWebClient UA uTorrent in use
RT	1	securoni-...	3.439	2016-04-06 19:52:24	192.168.0.31	52952	185.57.82.83	80	6	ET POLICY External IP Lookup ip-api.com
RT	1	securoni-...	3.448	2016-04-06 19:52:49	192.168.0.31	33343	67.215.246.10	6881	17	ET P2P BitTorrent DHT ping request
RT	1	securoni-...	3.454	2016-04-06 19:52:53	50.206.89.77	6881	192.168.0.31	33343	17	ET CINS Active Threat Intelligence Poor Reputation IP UDP group 5
RT	1	securoni-...	3.455	2016-04-06 19:52:54	192.168.0.31	33590	178.79.242.128	80	6	ET POLICY Outdated Windows Flash Version IE
RT	1	securoni-...	3.456	2016-04-06 19:53:00	68.116.5.134	1241	192.168.0.31	33343	17	ET CINS Active Threat Intelligence Poor Reputation IP UDP group 11
RT	2	securoni-...	3.458	2016-04-06 19:58:29	192.168.0.31	41045	176.31.102.125	80	6	ET POLICY HTTP Request to .su TLD (Soviet Union) Often Malware Related
RT	1	securoni-...	3.460	2016-04-06 20:19:39	192.168.0.31	33343	65.74.179.158	53	17	ET DNS Non-DNS or Non-Compliant DNS traffic on DNS port Reserved Bit Set - Likely Kazy

☐ Reverse DNS ☑ Enable External DNS

Src IP:
Src Name:
Dst IP:
Dst Name:

Whois Query: ● None ○ Src IP ○ Dst IP

☑ Show Packet Data ☑ Show Rule

```
flow:established,to_server; content:"APT-HTTP|2F|"; http_header; reference:url,help.ubuntu.com/community/AptGet/Howto; classtype:not-suspicious;
sid:2013504; rev:3;)
/nsm/server_data/securityonion/rules/securoni-eth0-1/downloaded.rules: Line 9611
```

IP	Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags	Offset	TTL	ChkSum
	192.168.0.31	185.5.160.169	4	5	0	242	54919	2	0	64	18696

TCP	Source Port	Dest Port	R 1	R 0	URG	ACK	PSH	RST	SYN	FIN	Seq #	Ack #	Offset	Res	Window	Urp	ChkSum
	44565	80	.	.	.	X	X	.	.	.	674824952	4278252635	8	0	1024	0	600

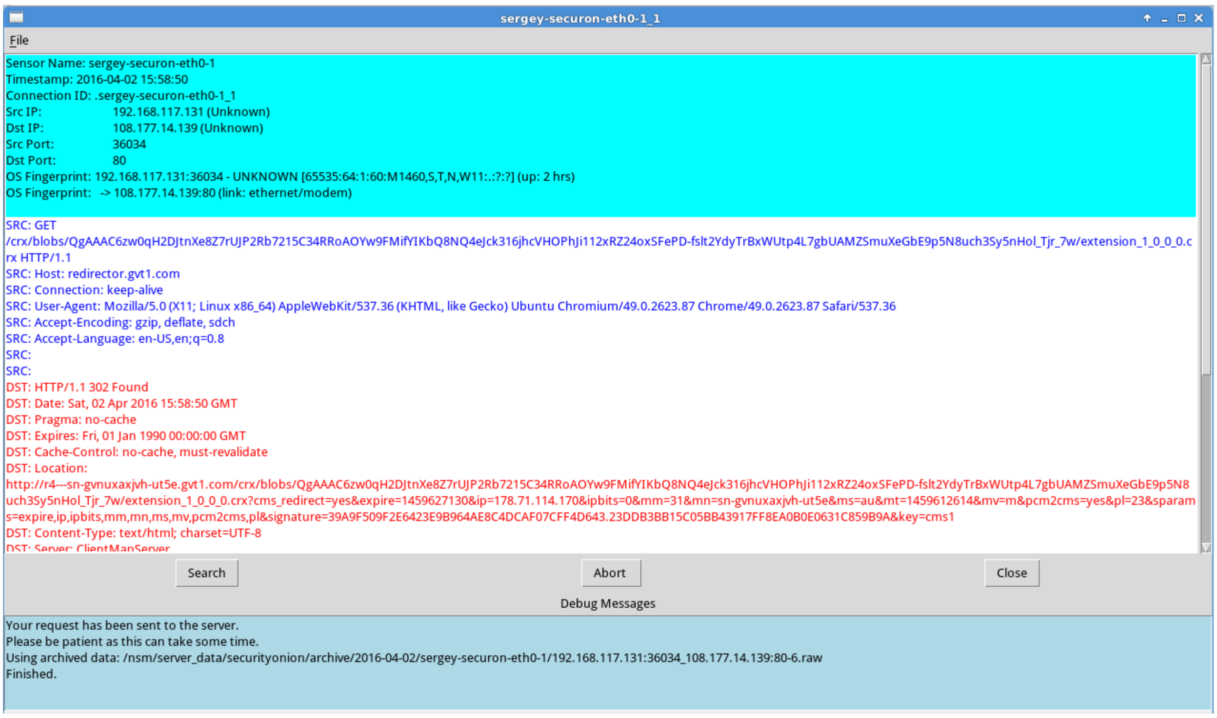
DATA

```
47 45 54 20 2F 76 69 72 74 75 61 6C 62 6F 78 2F    GET /virtualbox/
64 65 62 69 61 6E 2F 64 69 73 74 73 2F 74 72 75    debian/dists/tru
73 74 79 2F 6E 6F 6E 2D 66 72 65 65 2F 69 31 38    sty/non-free/i18
6E 2F 54 72 61 6E 73 6C 61 74 69 6F 6E 2D 65 6E    n/Translation-en
5F 55 53 2E 6C 7A 6D 61 20 48 54 54 50 2F 31 2E    _US.lzma HTTP/1.
31 0D 0A 48 6F 73 74 3A 20 64 6F 77 6E 6C 6F 61    1..Host: downloa
64 2E 76 69 72 74 75 61 6C 62 6F 78 2E 6F 72 67    d.virtualbox.org
```

○ Hex ● Text ☐ NoCase

FIGURE 26. Main screen

The user can see the transcription of each event by clicking the right mouse button and choosing the option needed – "Transcript". This function is provided by Bro (Figure 27).



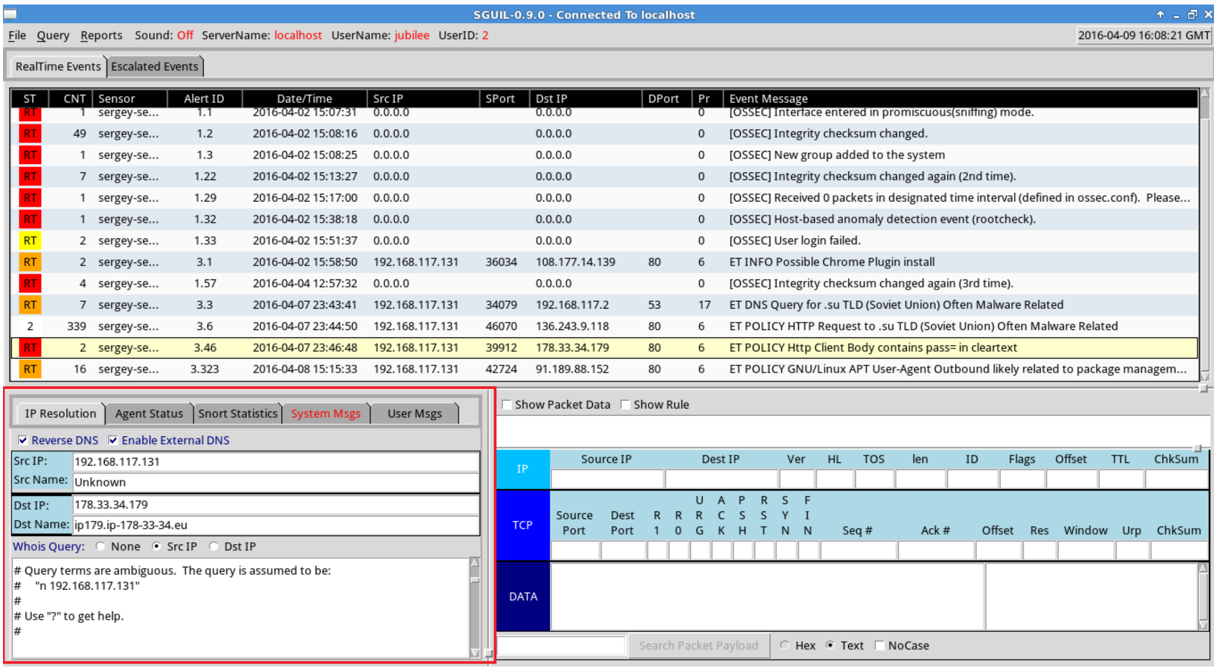
```
Sensor Name: sergey-securon-eth0-1
Timestamp: 2016-04-02 15:58:50
Connection ID: .sergey-securon-eth0-1_1
Src IP:         192.168.117.131 (Unknown)
Dst IP:         108.177.14.139 (Unknown)
Src Port:       36034
Dst Port:       80
OS Fingerprint: 192.168.117.131:36034 - UNKNOWN [65535:64:1:60:M1460,S,T,N,W11:.:?:?] (up: 2 hrs)
OS Fingerprint:   -> 108.177.14.139:80 (link: ethernet/modem)

SRC: GET
/crx/blobs/QgAAAC6zw0qH2DJtnXe8Z7rUJP2Rb7215C34RRoAOYw9FMifYIKbQ8NQ4eJck316jhcVHOPhJi112xRZ24oxSFePD-fslt2YdyTrBxWUtp4L7gbUAMZSmuXeGbE9p5N8uch3Sy5nHol_Tjr_7w/extension_1_0_0_0.c
rx HTTP/1.1
SRC: Host: redirector.gvt1.com
SRC: Connection: keep-alive
SRC: User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Ubuntu Chromium/49.0.2623.87 Chrome/49.0.2623.87 Safari/537.36
SRC: Accept-Encoding: gzip, deflate, sdch
SRC: Accept-Language: en-US,en;q=0.8
SRC:
SRC:
DST: HTTP/1.1 302 Found
DST: Date: Sat, 02 Apr 2016 15:58:50 GMT
DST: Pragma: no-cache
DST: Expires: Fri, 01 Jan 1990 00:00:00 GMT
DST: Cache-Control: no-cache, must-revalidate
DST: Location:
http://r4---sn-gvnuxaxjvh-ut5e.gvt1.com/crx/blobs/QgAAAC6zw0qH2DJtnXe8Z7rUJP2Rb7215C34RRoAOYw9FMifYIKbQ8NQ4eJck316jhcVHOPhJi112xRZ24oxSFePD-fslt2YdyTrBxWUtp4L7gbUAMZSmuXeGbE9p5N8
uch3Sy5nHol_Tjr_7w/extension_1_0_0_0.crx?cms_redirect=yes&expire=1459627130&ip=178.71.114.170&ipbits=0&mm=31&mn=sn-gvnuxaxjvh-ut5e&ms=au&mt=1459612614&mv=m&pcm2cms=yes&pl=23&sparam
s=expire,ip,ipbits,mm,mn,ms,mv,pcm2cms,pl&signature=39A9F509F2E6423E9B964AE8C4DCAF07CFF4D643.23DDB3BB15C05BB43917FF8EA0B0E0631C859B9A&key=cms1
DST: Content-Type: text/html; charset=UTF-8
DST: Server: ClientMapServer
```

```
Your request has been sent to the server.
Please be patient as this can take some time.
Using archived data: /nsm/server_data/securityonion/archive/2016-04-02/sergey-securon-eth0-1/192.168.117.131:36034_108.177.14.139:80-6.raw
Finished.
```

FIGURE 27. Bro event transcription

The last window in the left lower corner has five tabs: IP Resolution, Agent Status where the user can check what agents are working, Snort Statistic, System and User Messages (Figure.28).



Sound: Off ServerName: localhost UserName: jubilee UserID: 2

2016-04-09 16:08:21 GMT

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	1	sergey-se...	1.1	2016-04-02 15:07:31	0.0.0.0		0.0.0.0		0	[OSSEC] Interface entered in promiscuous(sniffing) mode.
RT	49	sergey-se...	1.2	2016-04-02 15:08:16	0.0.0.0		0.0.0.0		0	[OSSEC] Integrity checksum changed.
RT	1	sergey-se...	1.3	2016-04-02 15:08:25	0.0.0.0		0.0.0.0		0	[OSSEC] New group added to the system
RT	7	sergey-se...	1.22	2016-04-02 15:13:27	0.0.0.0		0.0.0.0		0	[OSSEC] Integrity checksum changed again (2nd time).
RT	1	sergey-se...	1.29	2016-04-02 15:17:00	0.0.0.0		0.0.0.0		0	[OSSEC] Received 0 packets in designated time interval (defined in ossec.conf). Please...
RT	1	sergey-se...	1.32	2016-04-02 15:38:18	0.0.0.0		0.0.0.0		0	[OSSEC] Host-based anomaly detection event (rootcheck).
RT	2	sergey-se...	1.33	2016-04-02 15:51:37	0.0.0.0		0.0.0.0		0	[OSSEC] User login failed.
RT	2	sergey-se...	3.1	2016-04-02 15:58:50	192.168.117.131	36034	108.177.14.139	80	6	ET INFO Possible Chrome Plugin install
RT	4	sergey-se...	1.57	2016-04-04 12:57:32	0.0.0.0		0.0.0.0		0	[OSSEC] Integrity checksum changed again (3rd time).
RT	7	sergey-se...	3.3	2016-04-07 23:43:41	192.168.117.131	34079	192.168.117.2	53	17	ET DNS Query for .su TLD (Soviet Union) Often Malware Related
2	339	sergey-se...	3.6	2016-04-07 23:44:50	192.168.117.131	46070	136.243.9.118	80	6	ET POLICY HTTP Request to .su TLD (Soviet Union) Often Malware Related
RT	2	sergey-se...	3.46	2016-04-07 23:46:48	192.168.117.131	39912	178.33.34.179	80	6	ET POLICY Http Client Body contains pass= in cleartext
RT	16	sergey-se...	3.323	2016-04-08 15:15:33	192.168.117.131	42724	91.189.88.152	80	6	ET POLICY GNU/Linux APT User-Agent Outbound likely related to package managem...

☑ Reverse DNS ☑ Enable External DNS

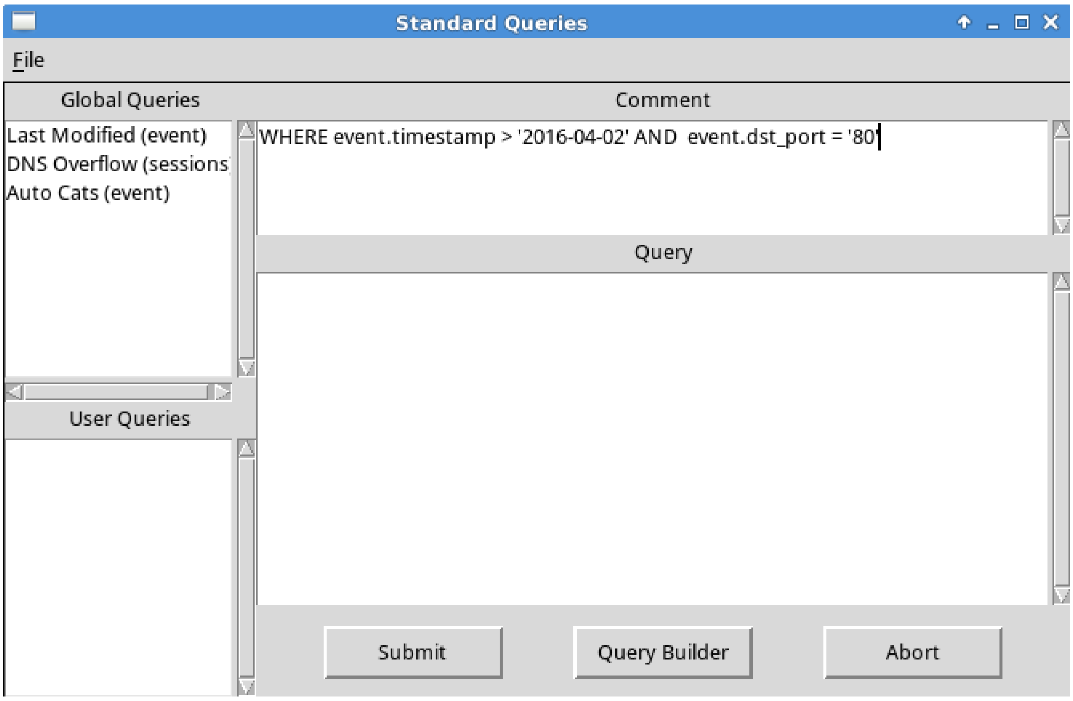
Src IP: 192.168.117.131
Src Name: Unknown
Dst IP: 178.33.34.179
Dst Name: ip179.ip-178-33-34.eu

Whois Query: ○ None ● Src IP ○ Dst IP

```
# Query terms are ambiguous.  The query is assumed to be:
#    "n 192.168.117.131"
#
# Use "?" to get help.
#
```

FIGURE 28. System window

Sguil support SQL queries for working with the alert database and also it has its own query builder. For example, the user wants to see alerts with Destination port 80 all time (Figure 29)

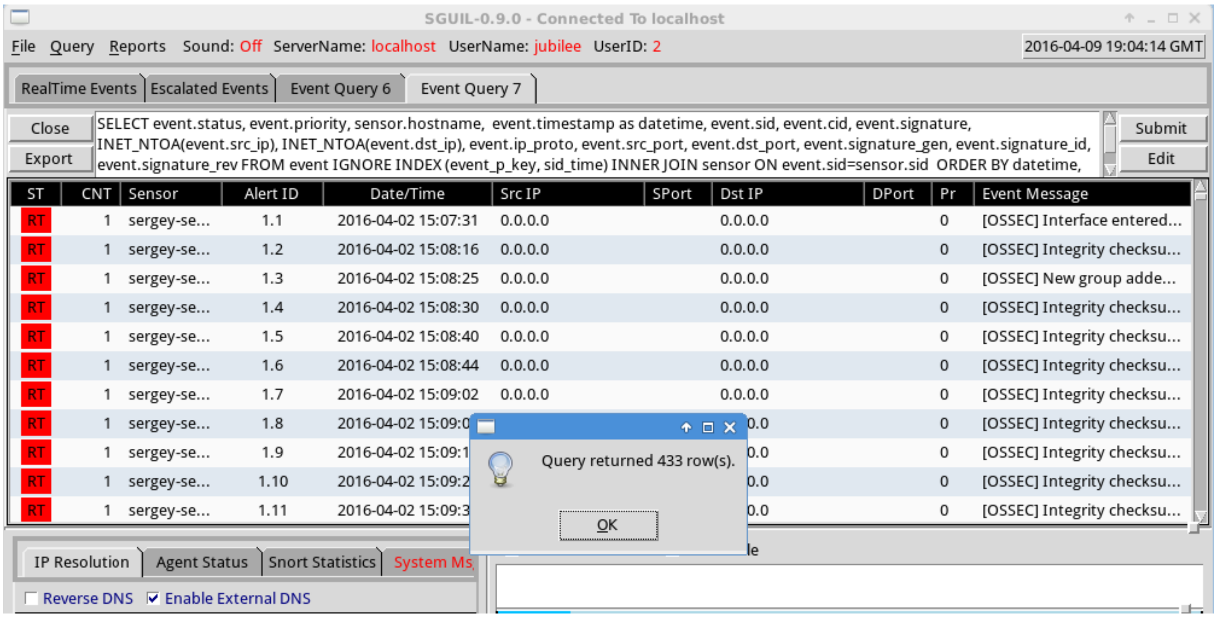


Last Modified (event)
DNS Overflow (sessions
Auto Cats (event)

```
WHERE event.timestamp > '2016-04-02' AND  event.dst_port = '80'
```

FIGURE 29. Sguil query

And the result of this query is shown in Figure 30.



```
SELECT event.status, event.priority, sensor.hostname,  event.timestamp as datetime, event.sid, event.cid, event.signature,
INET_NTOA(event.src_ip), INET_NTOA(event.dst_ip), event.ip_proto, event.src_port, event.dst_port, event.signature_gen, event.signature_id,
event.signature_rev FROM event IGNORE INDEX (event_p_key, sid_time) INNER JOIN sensor ON event.sid=sensor.sid  ORDER BY datetime,
```

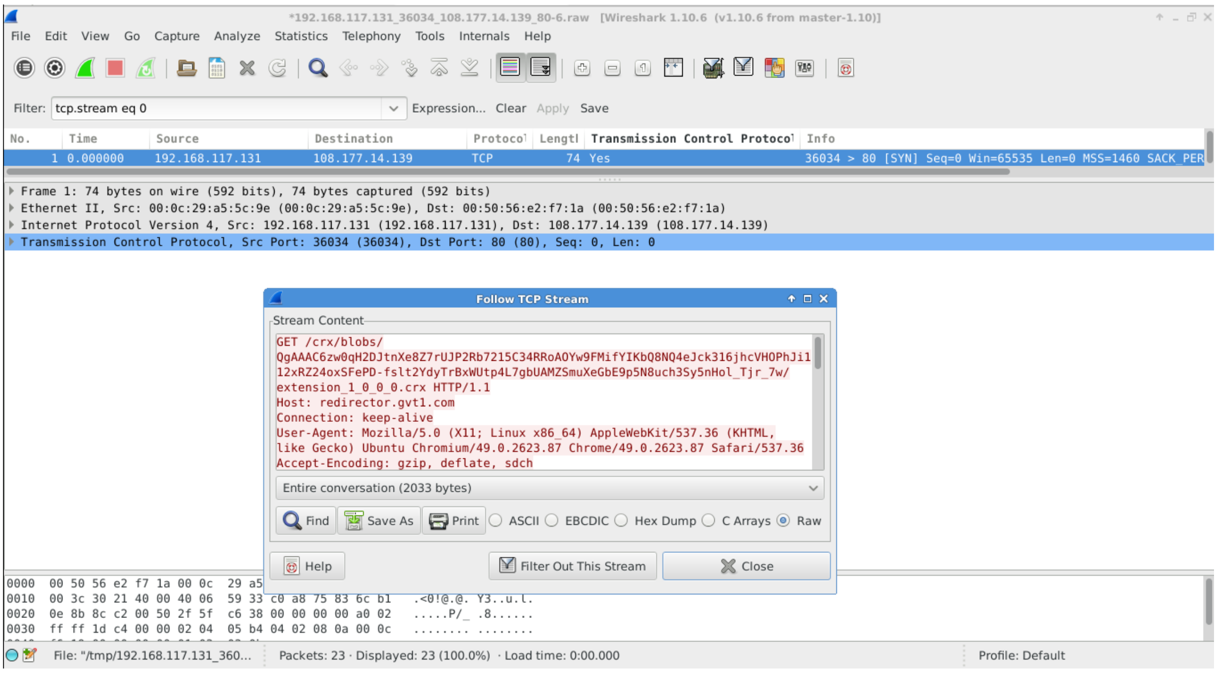
ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	1	sergey-se...	1.1	2016-04-02 15:07:31	0.0.0.0		0.0.0.0		0	[OSSEC] Interface entered...
RT	1	sergey-se...	1.2	2016-04-02 15:08:16	0.0.0.0		0.0.0.0		0	[OSSEC] Integrity checksu...
RT	1	sergey-se...	1.3	2016-04-02 15:08:25	0.0.0.0		0.0.0.0		0	[OSSEC] New group adde...
RT	1	sergey-se...	1.4	2016-04-02 15:08:30	0.0.0.0		0.0.0.0		0	[OSSEC] Integrity checksu...
RT	1	sergey-se...	1.5	2016-04-02 15:08:40	0.0.0.0		0.0.0.0		0	[OSSEC] Integrity checksu...
RT	1	sergey-se...	1.6	2016-04-02 15:08:44	0.0.0.0		0.0.0.0		0	[OSSEC] Integrity checksu...
RT	1	sergey-se...	1.7	2016-04-02 15:09:02	0.0.0.0		0.0.0.0		0	[OSSEC] Integrity checksu...
RT	1	sergey-se...	1.8	2016-04-02 15:09:0			0.0		0	[OSSEC] Integrity checksu...
RT	1	sergey-se...	1.9	2016-04-02 15:09:1			0.0		0	[OSSEC] Integrity checksu...
RT	1	sergey-se...	1.10	2016-04-02 15:09:2			0.0		0	[OSSEC] Integrity checksu...
RT	1	sergey-se...	1.11	2016-04-02 15:09:3			0.0		0	[OSSEC] Integrity checksu...

Query returned 433 row(s).

FIGURE 30. Query output

Sguil provides big opportunities for traffic analysis, especially for Snort, but some functions are not included, these ones can be used with integrated software. The Security Onion has good integration with other software such as: ELSA, WireShark, NetMiner, Bro and Dshield.

Wireshark provides detailed data packet analysis and operating, it is a packet sniffer and the user can trace tcp stream and download transported files or anything else (Figure.31).


*192.168.117.131_36034_108.177.14.139_80-6.raw [Wireshark 1.10.6 (v1.10.6 from master-1.10)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Expression... Clear Apply Save
Follow TCP Stream
Stream Content
Find
Save As
Print
Help
Filter Out This Stream
Close
Profile: Default

Filter: tcp.stream eq 0

```
No.   Time       Source            Destination      Protocol Length Transmission Control Protocol Info
    1 0.000000   192.168.117.131   108.177.14.139   TCP          74 Yes       36034 > 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PER
```

```
Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
Ethernet II, Src: 00:0c:29:a5:5c:9e (00:0c:29:a5:5c:9e), Dst: 00:50:56:e2:f7:1a (00:50:56:e2:f7:1a)
Internet Protocol Version 4, Src: 192.168.117.131 (192.168.117.131), Dst: 108.177.14.139 (108.177.14.139)
Transmission Control Protocol, Src Port: 36034 (36034), Dst Port: 80 (80), Seq: 0, Len: 0
```

```
GET /crx/blobs/
QgAAAC6zw0qH2DJtnXe8Z7rUJP2Rb7215C34RRoAOYw9FMifYIKbQ8NQ4eJck316jhcVHOPhJi1
12xRZ24oxSFePD-fslt2YdyTrBxWUtp4L7gbUAMZSmuXeGbE9p5N8uch3Sy5nHol_Tjr_7w/
extension_1_0_0_0.crx HTTP/1.1
Host: redirector.gvt1.com
Connection: keep-alive
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML,
like Gecko) Ubuntu Chromium/49.0.2623.87 Chrome/49.0.2623.87 Safari/537.36
Accept-Encoding: gzip, deflate, sdch
```

Entire conversation (2033 bytes)

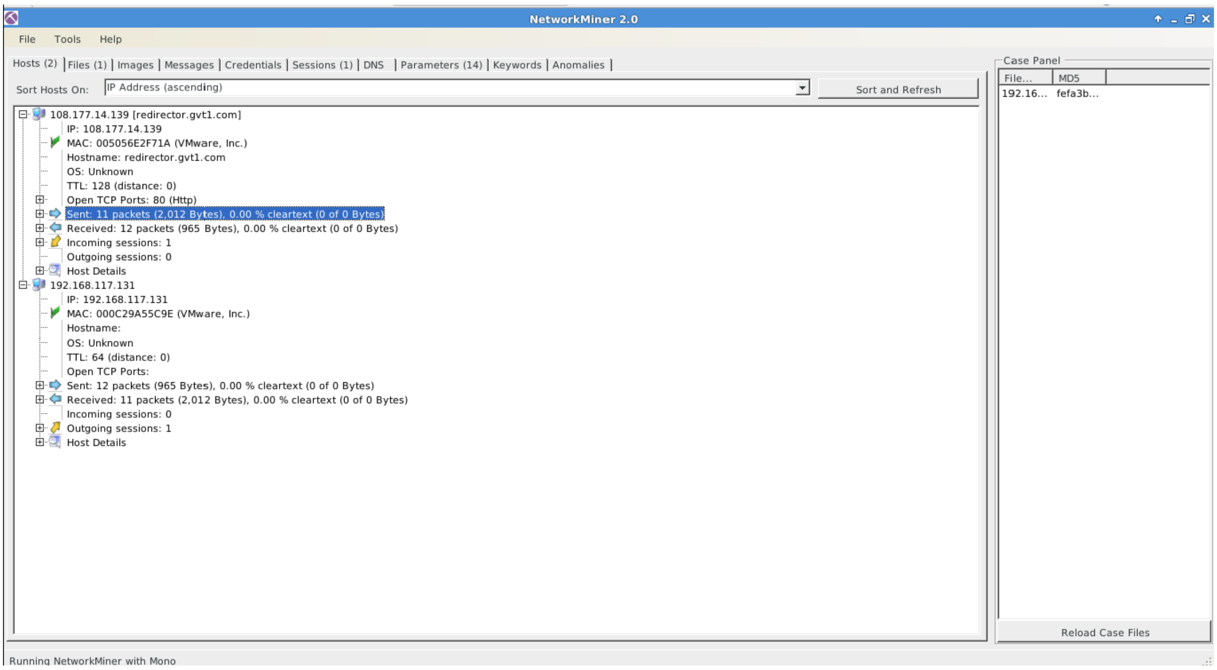
○ ASCII ○ EBCDIC ○ Hex Dump ○ C Arrays ● Raw

```
0000  00 50 56 e2 f7 1a 00 0c  29 a5
0010  00 3c 30 21 40 00 40 06  59 33 c0 a8 75 83 6c b1   .<0!@.@. Y3..u.l.
0020  0e 8b 8c c2 00 50 2f 5f  c6 38 00 00 00 00 a0 02   .....P/_ .8......
0030  ff ff 1d c4 00 00 02 04  05 b4 04 02 08 0a 00 0c   ........ ........
```

File: "/tmp/192.168.117.131_360... Packets: 23 · Displayed: 23 (100.0%) · Load time: 0:00.000

FIGURE 31. Wireshark main screen

NetworkMiner is a data packets analyzer and sniffer, such as the WireShark it captures packets and has the same functions with another interface (Figure.32).



```
108.177.14.139 [redirector.gvt1.com]
    IP: 108.177.14.139
    MAC: 005056E2F71A (VMware, Inc.)
    Hostname: redirector.gvt1.com
    OS: Unknown
    TTL: 128 (distance: 0)
    Open TCP Ports: 80 (Http)
    Sent: 11 packets (2,012 Bytes), 0.00 % cleartext (0 of 0 Bytes)
    Received: 12 packets (965 Bytes), 0.00 % cleartext (0 of 0 Bytes)
    Incoming sessions: 1
    Outgoing sessions: 0
    Host Details
192.168.117.131
    IP: 192.168.117.131
    MAC: 000C29A55C9E (VMware, Inc.)
    Hostname:
    OS: Unknown
    TTL: 64 (distance: 0)
    Open TCP Ports:
    Sent: 12 packets (965 Bytes), 0.00 % cleartext (0 of 0 Bytes)
    Received: 11 packets (2,012 Bytes), 0.00 % cleartext (0 of 0 Bytes)
    Incoming sessions: 0
    Outgoing sessions: 1
    Host Details
```

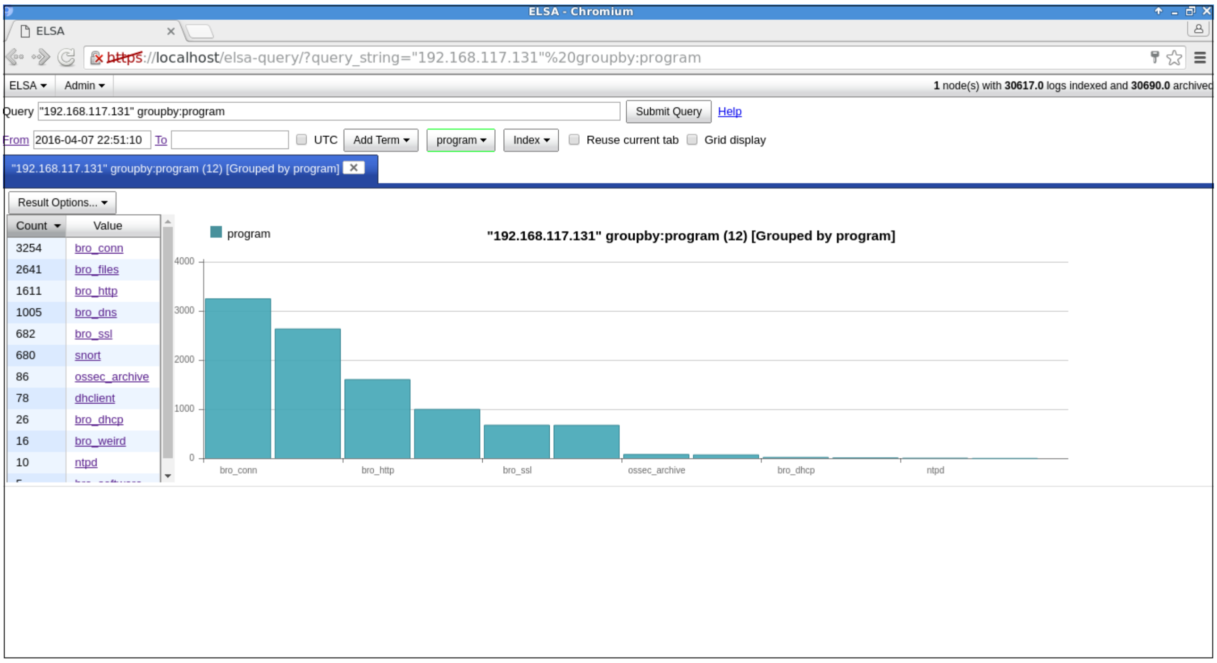
File...	MD5
192.16...	fefa3b...

Running NetworkMiner with Mono

FIGURE 32. NetworkMiner environment

These both programs help to improve IDS/IPS because the output from Snort is not enough to carve files, Snort reacts only to particular sections of the traffic and alerts, when Wireshark and NetworkMiner can capture all the packets from session. And then, the user can add rules based on this data.

ELSA is a browser based traffic analyzer with the feature of IP port lookup, exactly this function is integrated with Sguil (Figure 33)



~~https~~://localhost/elsa-query/?query_string="192.168.117.131"%20groupby:program

1 node(s) with **30617.0** logs indexed and **30690.0** archived

Query "192.168.117.131" groupby:program

From 2016-04-07 22:51:10 To

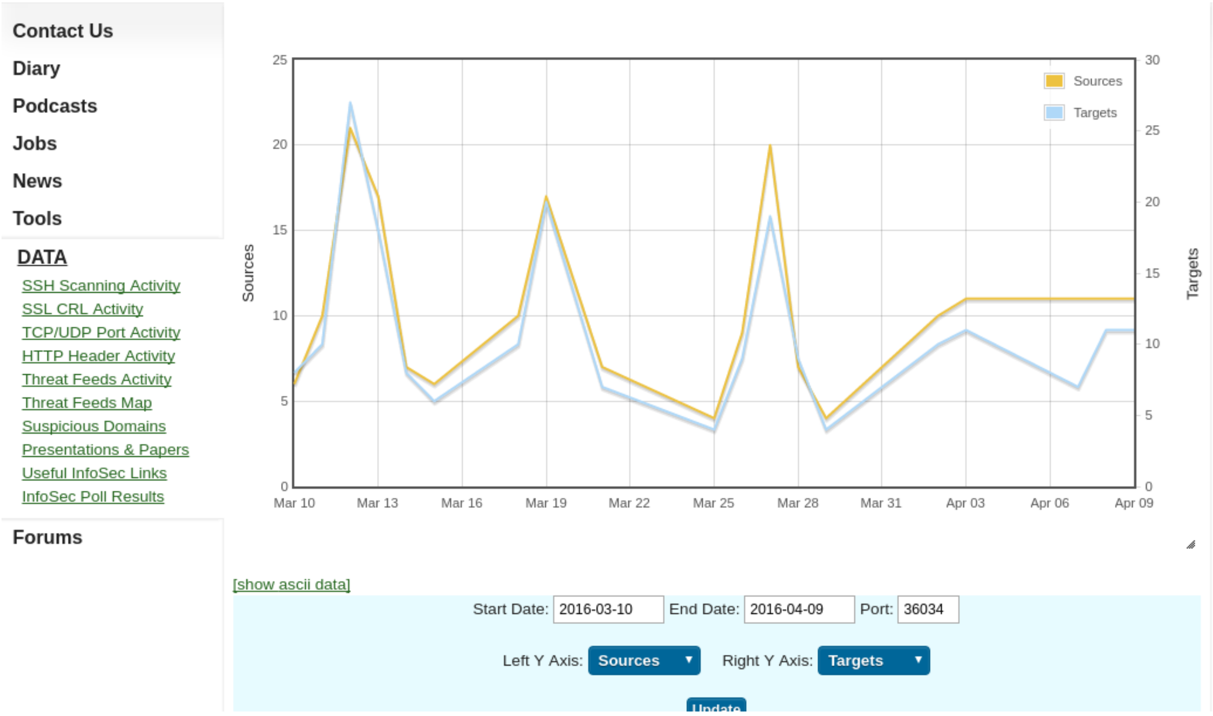
"192.168.117.131" groupby:program (12) [Grouped by program]

"192.168.117.131" groupby:program (12) [Grouped by program]

Count	Value
3254	bro_conn
2641	bro_files
1611	bro_http
1005	bro_dns
682	bro_ssl
680	snort
86	ossec_archive
78	dhclient
26	bro_dhcp
16	bro_weird
10	ntpd

FIGURE 33. ELSA statistics

Dshield is a community-based collaborative firewall log correlation system. It keeps a lot of data collected from all the world. From Sguil you can do fast lookup of needed port or IP (Figure 34).



Start Date: 2016-03-10 End Date: 2016-04-09 Port: 36034

Left Y Axis: Sources Right Y Axis: Targets

FIGURE 34. Dshield statistics

4 RESULTS

Now when the system is installed and works properly, it is time to find the pros and cons. There are not so many disadvantages, and even the existing issues can be solved easily.

4.1 Advantages

Security Onion is a quite complicated system. No one IDS/IPS vendor provides so many functions and software. At this point Security Onion with Snort is very flexible environment. It allows to the user tune his network security as he wish. It makes life easier, because there are a lot of preinstalled tools and configurations. When the user installs clean Snort without Security Onion, the user has to download all packets and scripts. Then the user has to configure preprocessors, detectors, sniffers, agents and sensors and to assign them one to other. It requires a lot of time and deep knowledge of the Unix systems. In the Security Onion the user can use these functions through SO Wizard.

Security Onion has preinstalled sensor management tools, traffic analyzers and packet sniffers and so on. It can be operated without any additional IDS/IPS software.

Easy rule updates and a lot of ready rules can be found in the Internet making this IDS/IPS system very helpful for protecting the networks. Fast development of Snort and Security Onion provides a lot of updates, that improve the security level.

When using the Wizard installation, it advices to the user Best Practice installation, which means that will run more successful installation, based on the global statistic of SO users.

4.2 Disadvantages

On the other hand, this system also has few disadvantages. In the Security Onion Snort does not work as IPS after installation, only as IDS, and the user cannot find any instructions about it on the SO website. Therefore, he has to develop a lot of information related to Snort to configure it as IPS.

Also Security Onion does not support the Wi-Fi network (WLAN) managing from the box. During the installation of SO it disables Network Manager and Wi-Fi connections in the user system. To use WLAN the user has to skip Wizard installation and to do it manually without any instructions.

A lot of integrated software require a lot of time to learn. So, to work more efficient the user should know how to work with different programs. One serious flaw of the system is that SO does not have full backups of the configuration files. It only makes backup of rules automatically. Therefore, to make any backups the user has to use external software.

5 CONCLUSION

The aim of study was to get familiar with the IDS and IPS tools, make comparison of the different vendors, and make some user tests. And for this purpose I have chosen Snort in the Security Onion distribution.

Security Onion is a great system for providing network security. It includes different IDS and IPS vendors and a multiple environment for working. Snort is not less flexible thing. It can be configured as IPS or IDS, the virus detection and even the DDoS protection.

It has some weaknesses is listed in the Chapter 4, but I think most of them, or even all of them will be fixed in the new versions.

On the other hand Security Onion makes these processes faster for newbies and provides more opportunities for experts, in comparison with many other IDS and IPS. And, it is a good alternative to the hardware IDS/IPS for small enterprises that have a limited budget.

Finally, this thesis can be used as a installation guide of Snort for the network security administrators, as it provides base knowledge about it and describe some issues that are not included in the tool’s instructions. In my opinion, it is a good idea to create library and collect as much as possible information about IDS/IPS, and this research can be used as a part of it.

Also I think here is many directions for the further developments. As I had pretty poor hardware for the testing, I could not make stress tests, performance tests and so on, because it requires a lot of the resources. In the server environment it can be implemented, it has more resources to do it, and then researches can have a bigger base for the comparison of the different type network security tools.

The network security theme will be demanded for a long time, perhaps always. And in this case, this topic looks relevant for the thesis. A lot of developers can make their own research and this one can be helpful for them.

6 BIBLIOGRAPHY

Doug Burks, 2015d, Security Onion Solutions [referred 18.02.2016]. Available in www-format:
https://github.com/Security-Onion-Solutions

Cisco, 2016, Snort documentation [referred 25.02.2016]. Available in www-format:
https://www.snort.org/documents

Karen Scarfone, Peter Mell, 2007a, Guide to Intrusion Detection and Prevention Systems (IDPS) [referred 22.02.2016]. Available in www-format:

http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf

Matthew Pascucci, 2013a, Top five free enterprise network intrusion-detection tools [referred 26.02.2016]. Available in www-format: http://searchsecurity.techtarget.com/tip/Top-five-free-enterprise-network-intrusion-detection-tools

OSSEC Project Team, 2015, OSSEC documentation [referred 28.02.2016]. Available in www-format:
https://ossec.github.io/docs/

OISF, 2016, Suricata wiki [referred 28.02.2016]. Available in www-format:
https://redmine.openinfosecfoundation.org/projects/suricata/wiki

Open WIPSS-ng, 2016, official web-site [referred 29.02.2016]. Available in www-format:
http://www.openwips-ng.org/

Bro Project Team, 2016, Bro documentation [referred 29.02.2016]. Available in www-format:
https://www.bro.org/documentation/index.html

David J. Day, 2011, A Performance Analysis Of Snort And Suricata Network Intrusion Detection And Prevention Engines [referred 05.03.2016]. Available for downloading in .pdf:
A Performance Analysis Of Snort And Suricata

Aamir Lakhani, 2016a, Blog about IT security [referred 09.03.2016]. Available in www-format:
http://www.doctorchaos.com/

GLC team, 2016, One of the bigger IT web-newspaper in Russia, security [referred 12.03.2016]. Available in www-format:
https://xakep.ru/

Ubuntu, **2016,** Ubuntu documentation and help wiki [referred 21.03.2016]. Available in www-format:
https://help.ubuntu.com/

Google, 2016, Google web-forum about Security Onion [referred 26.03.2016]. Available in www-format:
https://groups.google.com/forum/#!forum/security-onion

Doug Burks, 2016, Personal blog about Security Onion [referred 27.03.2016]. Available in www-format:
http://blog.securityonion.net/