Open Source IDS/IPS Native Security Rule Migration from IPv4 to IPv6 and their effectiveness and comparison.

T.D.K Pathirana

2014

Open Source IDS/IPS Native Security Rule Migration from IPv4 to IPv6 and their effectiveness and comparison.

A dissertation submitted for the Degree of Master of Science in Information Security

T.D.K Pathirana.

University of Colombo School of Computing

2014

Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Students Name: T.D.K. Pathirana

Registration Number: 2014/MIS/015

Index Number: 14770153

______________________ _____________________

Signature Date

This is to certify that this thesis is based on the work of Mr. T.D.K. Pathirana

under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by:

Supervisor Name: Dr. D.A.S. Atukorale

______________________ ______________________

Signature Date

ABSTRACT

Internet Protocol Version 6 has become the new trend in Internet as its predecessor Internet Protocol Version 4 started its exhaustion since 2011. With the rising traffic of IPv6 all devices building the Internet has been upgraded or rebuilt to support the 128-bit address. But as the number insists there are numerous addresses in the space. Because of that and as it is still young to production environment there are huge number of security vulnerabilities in IPv6. Even though there are great quantity of vulnerabilities, we cannot avoid using IPv6 because of the addressing issue arising for newly connected devices and services. Therefore, we have to use IPv6 with precautions and the best precaution we have today is to deploy an Intrusion detection or prevention system in the network. When dealing with IPS/IDS solutions there are good competitive players in commercial network security world but almost all costs in huge amounts. Therefore, the best solution is to develop a suitable Free and Open Source Software to act as an IPS or IDS. This thesis aims on developing such a system with totally free and cost effective way. For this objective, main issue was there are no good single IDS application that totally supports every security feature. In IPv4, one of the best IDS/IPS FOSS systems is Security Onion Linux based Distribution, it is maintained by a company called Security Onion Solutions and currently it has a high number of user based community. Security Onion runs under an Ubuntu/Debian based environment and it makes the users more attracted to Security Onion as, handling Ubuntu operations are easy than other Linux operating systems. Also because of its big community, problems arising while operations can be easily solved. Security Onion has a fully effective detection rule base due to its community. Therefore, this thesis aims at enabling Security Onion with IPv6 and fine tuning it. Security Onion uses Snort as one of its IDS engines and I will be targeting Snort for the ease of development and as it is already supporting IPv6. Even though Snort has the support, other components in Security onion specially the applications used to process alerts and do the reporting are yet not available in IPv6 mode. Also because of the lack of IPv6 detection rules new rules based on ICMPv6 was created. But for logging and reporting new system is introduced in the thesis using log analyzing tool ELK stack based on ElasticSearch and Kibana. During the process many problems were faced as most of the components pre-installed in Security Onion is not supporting IPv6 and as the developers are also masking IPv6 addresses to 0.0.0.0 to avoid detection of other traffic. As its flexibility in handling big data many specific visualization charts and graphs can be created in Kibana allowing fully user made graphing. Therefore, when Analyzing dual stack systems ELK stack was every efficient and cost effective as it detects threats belonging to both versions.

Keywords: IPv6 IDS/IPS, ELK stack, Security Onion, ICMPv6, Snort rules

Acknowledgment

During my effort on this thesis, it would have not been possible to make this a success without the support of certain individuals who helped me during the whole period. Therefore, I use this to extend my gratitude to all of them.

I am truly indebted to my supervisor Dr. D. A. S. Atukorale for his guidance and superior support in supervising as well as for showing me the correct path when I went rogue and also for the support had in completion of this thesis.

My gratitude also goes to Dr. Manjusri Wickramasinghe for guiding me from the beginning as the project coordinator and showing me the required path that I have to guide my project.

I would also like to give my special gratitude to Dr. K.G.H.D. Weerasinghe, Director ICT at University of Kelaniya where I work for providing me the opportunity to test my systems in live networks at the university.

Also, I should thank my beloved wife for keeping me awake while I do the research in late nights.

Finally, my appreciations also go to Mr. Charith, Mr. Anupama and my colleagues for supporting in various ways during the process by helping me out with their abilities.

Contents

List of Figures

List of Listings

Abbreviations

DDOS – Distributed Denial of Service

DHCPv6 – Dynamic Host Control Protocol version 6

ELK – Elasticsearch, Logstash, Kibana stack

ICMP – Internet Control Messaging Protocol

ICMPv6 – Internet Control Messaging Protocol version 6

IDS – Intrusion Detection System

IoT – Internet of Things

IP – Internet Protocol

IPS – Intrusion Prevention System

IPsec – Internet Protocol Security

IPv4 – Internet Protocol version 4

IPv6 – Internet Protocol version 6

MIPv6 – Mobile IPv6

SO - Security Onion

SO-ELK – Security Onion – Elasticsearch, Logstash, Kibana stack

UTM – Unified Threat Management

Chapter 1 - Introduction

1.1 Background

Today Internet has two main paths, namely Internet Protocol version 4 and Internet Protocol version 6. With the exhaustion of IPv4 addresses in Internet Assigned Numbers Authority(IANA) in 2011 [1] [2] all five Regional Internet Registries (RIR) begin to start their own exhaustions. The Figure 1 shows the decrease path of IPv4 addresses of RIR's.

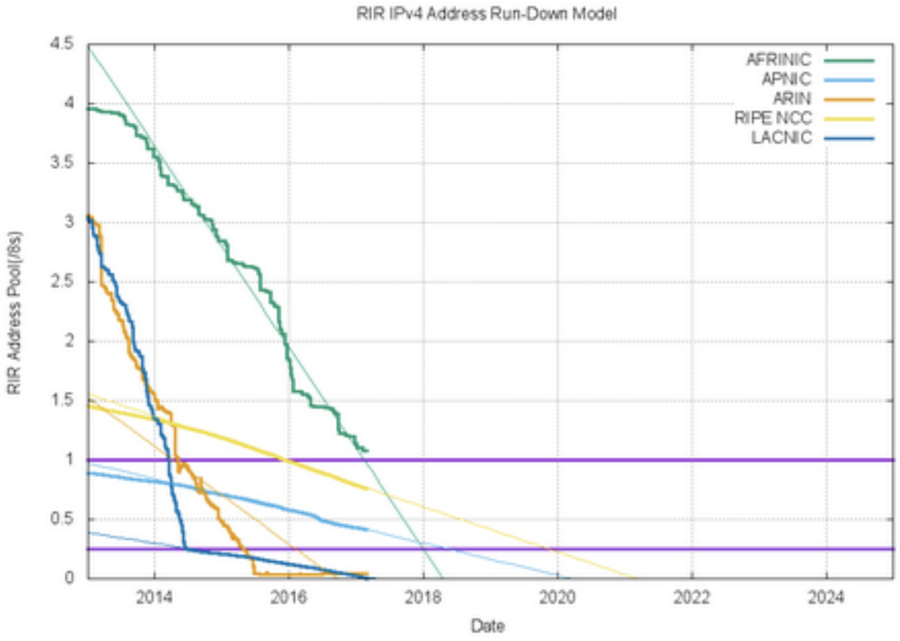
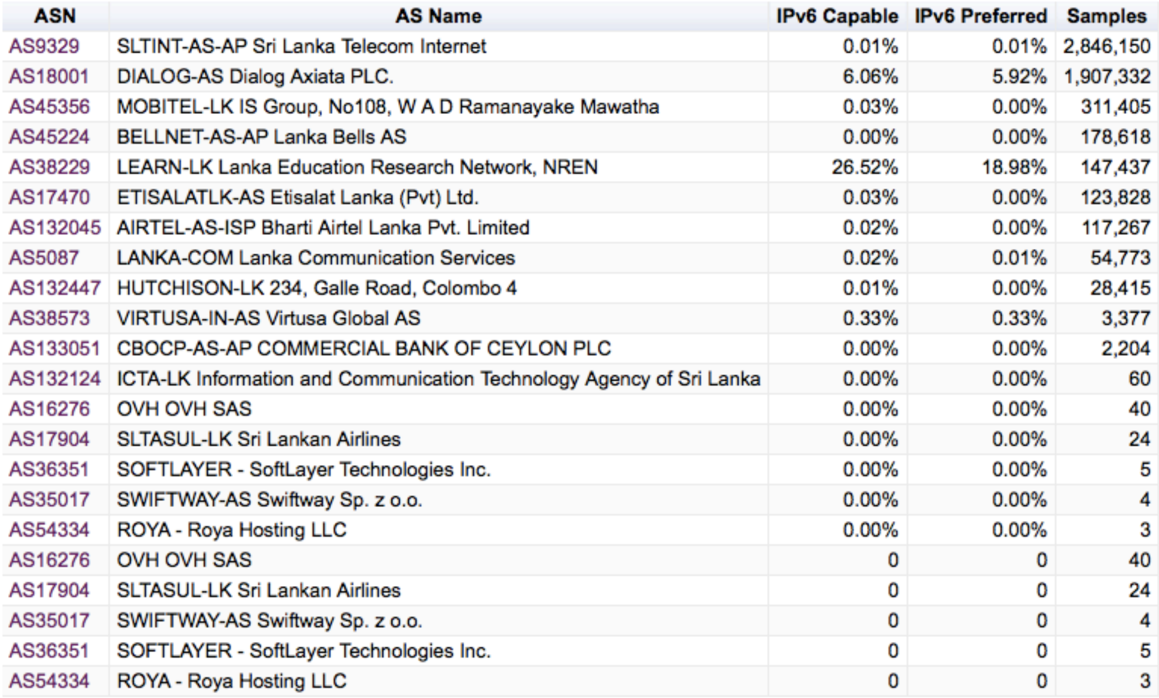


Figure 1: IPv4 Exhaustion as of January 2017, source: ipv4.potaroo.net

Because of this addressing issue Internet Protocol version 6 started to immerge after its introduction a decade ago [3]. According to Internet Corporation For Assigned Names And Numbers (ICANN) the body behind Internet Addressing, all major networks supports IPv6 as of February 2017 [4].

While most of the major ISP's started deploying IPv6, in Sri Lanka all ISP's upgraded their Backbones during past few years. Few started giving IPv6 to public. Figure 2 shows the distribution of IPv6 among Sri Lankan Network Operators as of February 2017 and Figure 3 shows how the world uses IPv6 according to google statistics and Figure 4 shows how Sri Lanka reaching IPv6 Internet.



ASN	AS Name	IPv6 Capable	IPv6 Preferred	Samples
AS9329	SLTINT-AS-AP Sri Lanka Telecom Internet	0.01%	0.01%	2,846,150
AS18001	DIALOG-AS Dialog Axiata PLC.	6.06%	5.92%	1,907,332
AS45356	MOBITEL-LK IS Group, No108, W A D Ramanayake Mawatha	0.03%	0.00%	311,405
AS45224	BELLNET-AS-AP Lanka Bells AS	0.00%	0.00%	178,618
AS38229	LEARN-LK Lanka Education Research Network, NREN	26.52%	18.98%	147,437
AS17470	ETISALATLK-AS Etisalat Lanka (Pvt) Ltd.	0.03%	0.00%	123,828
AS132045	AIRTEL-AS-ISP Bharti Airtel Lanka Pvt. Limited	0.02%	0.00%	117,267
AS5087	LANKA-COM Lanka Communication Services	0.02%	0.01%	54,773
AS132447	HUTCHISON-LK 234, Galle Road, Colombo 4	0.01%	0.00%	28,415
AS38573	VIRTUSA-IN-AS Virtusa Global AS	0.33%	0.33%	3,377
AS133051	CBOCP-AS-AP COMMERCIAL BANK OF CEYLON PLC	0.00%	0.00%	2,204
AS132124	ICTA-LK Information and Communication Technology Agency of Sri Lanka	0.00%	0.00%	60
AS16276	OVH OVH SAS	0.00%	0.00%	40
AS17904	SLTASUL-LK Sri Lankan Airlines	0.00%	0.00%	24
AS36351	SOFTLAYER - SoftLayer Technologies Inc.	0.00%	0.00%	5
AS35017	SWIFTWAY-AS Swiftway Sp. z o.o.	0.00%	0.00%	4
AS54334	ROYA - Roya Hosting LLC	0.00%	0.00%	3
AS16276	OVH OVH SAS	0	0	40
AS17904	SLTASUL-LK Sri Lankan Airlines	0	0	24
AS35017	SWIFTWAY-AS Swiftway Sp. z o.o.	0	0	4
AS36351	SOFTLAYER - SoftLayer Technologies Inc.	0	0	5
AS54334	ROYA - Roya Hosting LLC	0	0	3

Figure 2: Sri Lankan ISP Ipv6 Capability, Source: APNIC (February 2017)

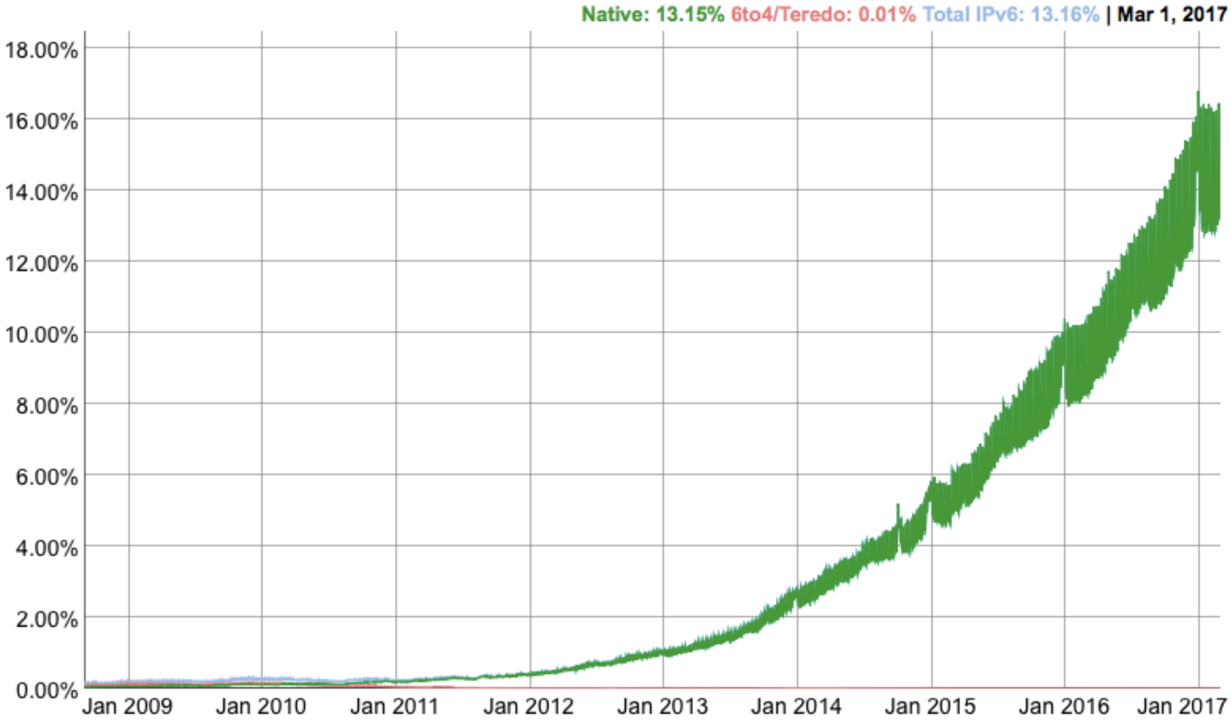


Figure 3: World IPv6 Statistics, source: https://www.google.com/intl/en/ipv6/statistics.html

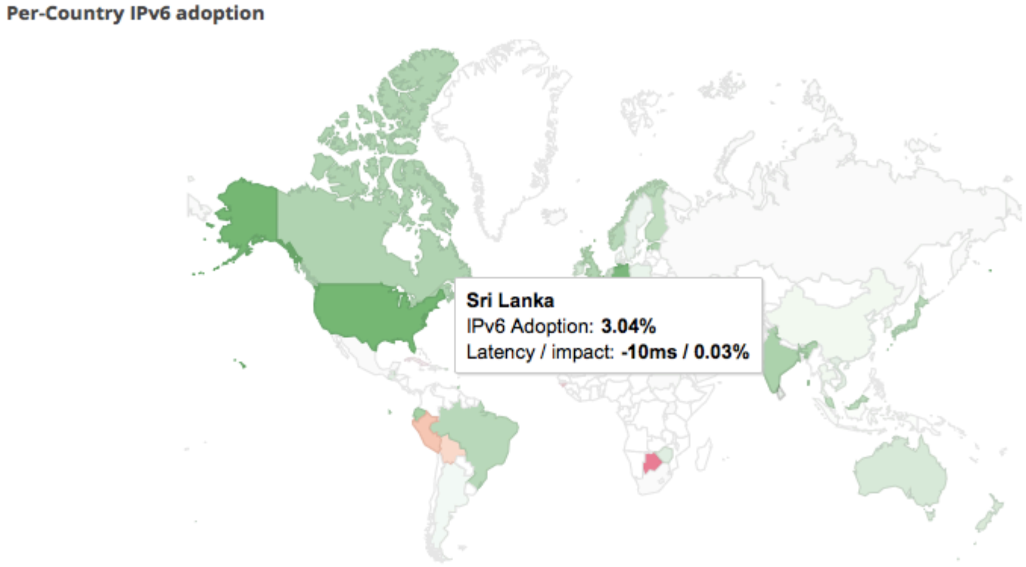


Figure 4: Sri Lanka IPv6 Adoption as of March 2017, Source: https://www.google.com/intl/en/ipv6/statistics.html

As Internet Protocol Version 6 is getting higher in usage and with the introduction of Internet of Things and mobile IP addresses, the security aspect is also rising. Also many service providers are migrating from IP version 4 to IP version 6 with dual stack capability.

Therefore when considering the security perspective, with the up rise of IPv6, many commercial multilayer security devices which were fine tuned for IPv4 are now been developed to detect IPv6 [5] [6] [7], but still fail doing it effectively equal in both versions [8].

Intrusion prevention systems (IPS), also branded as intrusion detection and prevention system (IDPS), is the network security segment that monitor network happenings for malicious actions. The main roles of intrusion prevention systems are to detect malicious activity, log data about this action, try to block/break it, and inform it. When considering open source IDS/IPS solutions which are IPv6 enabled, Snort[1] and Suricata[2] heads the race, but still there are lots of undetectable IPv6 traffic going through them. Even though these tools are very much good in detecting IPv4 threats, due to transitional errors in security detecting methods IPv6 threats are still unseen.

There are lots of version 4 security rules but they fail when used to detect version 6 traffic. As there are certain differences between these two versions, security implications are inevitable and therefore many fresh studies are done for securing native IPv6.

But in a dual stack environment as shown in Figure 5, application can access the network through both versions. Therefore, as we find IPv4 security rules already in IDS/IPS systems it is efficient to make them effective for IPv6 traffic. Then these rules also can be tested against native IPv6.

[1] *Snort* is an open-source, free and lightweight network intrusion detection system (NIDS) software for Linux and Windows to detect emerging threats. [10]

[2] *Suricata* is a high performance Network IDS, IPS and Network Security Monitoring engine. [41]

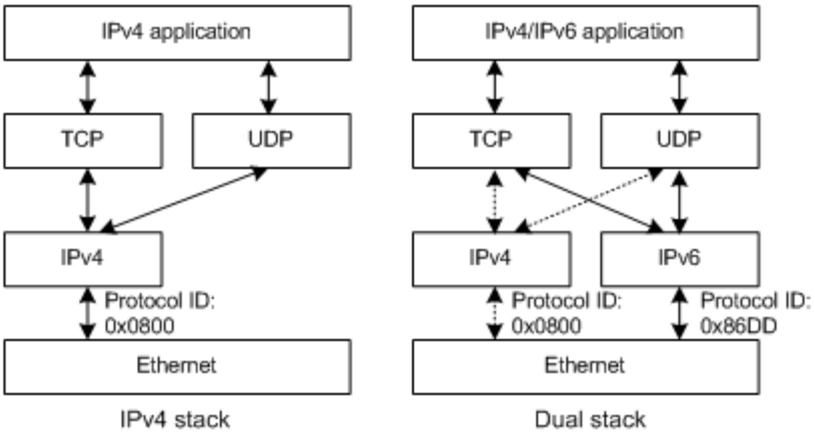


Figure 5: Dual Stack Implementation, Source: http://www.whatismyipaddress.com

1.2 Research Question

Many current multi-layer threat detection systems fail to detect IPv6 traffic correctly and therefore the requirement for such mechanism is crucial. This research is to identify common risks occur because of IPv6 networks and build an open source IDS/IPS based on existing freely available IPv4 signatures and threat anomalies.

1.3 Significance of the Research

Problem identified is about the limitations of the IPS/IDS systems when analyzing IPv6 traffic. As IPv6 will be the future of the Internet, it is a great need creating a system that can detect network threats with good efficiency. Also it is significant to realize that there can be numerous vulnerabilities in IPv6 plus associated attacks are still mainly hypothetical. For the reason that IPv6 has yet been implemented in minor and naively, but technically, in almost all systems, there have been a lesser amount of published actual breaches. It's thinkable that roughly some of the attacks that we are acquainted with or by other security investigators are not efficient in exercise. If nothing else, they work for an unproven implementation and warning to us all beforehand IPv6 security becomes the standard.

1.4 Scope and Objectives

The main objective of this thesis is to build up an open source IPS/IDS solution which will detect IPv6 traffic alerts and to visualize the detected attacks. During the process scope will be to enable an existing Linux Security Distribution - Security Onion [9] to detect IPv6. Doing so main focus will be on the IDS engine Snort and the reporting framework of Security Onion.

Snort Engine is currently built with IPv6 support [10] and I will be creating several ICMP version 6 [11] based rules to detect attacks. Also as Security Onion distribution currently not supporting IPv6, my objective is to enable IPv6 alerts in security onion by introducing a new reporting tool. But this will not focus on moving all tools in Security Onion to IPv6.

1.5 Limitations and Assumptions

As IPv6 deployments are still very fewer in Sri Lanka as an initial setup design, finding will only be limited to the networks belongs to Lanka Education And Research Network [12]. Because of the number of threats can be numerous, I will look only specifically on ICMPv6 and DDOS attacks in this study. Also it is not assumed that IPsec is fully implemented on the systems. During the evaluation, new system is compared with a commercial UTM device which supports IPv6. But as the detection rules are different on both engines, Snort and Commercial Device, the signature results cannot be compared completely. Therefore, it is assumed that if snort detects an ICMPv6 and the Commercial device also detects, then snort ICMPv6 detection is a success.

1.6 Structure of Thesis

Second chapter of this Thesis will describe and contrast the previous work and literature found on IPv6 security and IDS/IPS solutions. It will discuss the current trends on the field of IPv6 Security and as this thesis focus on developing a system that detects IPv6 Chapter two will provide a great insight on the background.

Third chapter will discuss the design of the proposed system and how it is implemented. This chapter may get in to details of highly technical aspects of IPv6 IDS/IPS development as well as the installation processes in Linux environments. Also it will describe how alerts are being visualized in the new system and how to create and populate new data tables.

On Forth chapter, the results have been discussed in the primary installation and a simple comparison has been done with a commercial UTM device with the limitations mention in section 1.5.

On the fifth chapter it is concluded the final results and have discussed the possibilities of deploying the system effectively in production environment. Also the future work are also discussed.

Bibliography, the referred, cited documents and related work have been listed on the next chapter.

Finally, Annexure with relevant documents used or assisted in the thesis process is amended at the end of the thesis. Partial sample source codes are also amended.

Chapter 2 - Review of Related Literature

There is various research done for IPv4 security and IPv6 security separately. But very few address on transition or migrated rules. A paper by Convery & Miller, 2004 [13] on IPv6 and IPv4 threat comparison undoubtedly pointsout that without clear understanding of IPv6 security in Networks may end up in difficulties in smooth transactions, and this will require great training and skill. In Identifing threats most engineers rely on IDS and if these IDS cannot identify threats correctly then again we are on trouble.

According to Convery & Miller, Migration of IPv6 to a dual stack environment will increase the threat index and therefore we need to idenfity which threats we are facing. They also discuss about the best practices of implementing a dual stack network from the Cisco background [13]. Convey & Miller insists that some of the main concerns for placement of a dual-stack Internet facing edge are guaranteeing that we have a good configuration modification control and monitoring for our firewall and internet router.

Thinking of an example, the configuration of the IPv4-only cisco firewall in a test lab may get just over 100 lines. But when IPv6 is included, the configuration is doubled. Just like in any other device, as the soze of the configuration escalates, so does the coincidental error. Addind that with the fact that these hosts now have two separate protocols on which they can be attacked as well as a lack of IPv6 support in current security proficiency, and the chance that the enemy will find a new way into our network with IPv6 increases.

Even with the introduction of IPv6, traditional virus in no way changes. E-mail based viruses or those that infect removable media remain as anyone would expect. But, worms or viruses and intruders that use some ways of Internet scanning to find defenseless hosts may involve in major barriers to promulgation in IPv6. Therefore, further research is essential to identify how important change this would be or what methods the worm writer could engage to improve its propagation efficacy. Anyhow it would seem that a SQL slammer-type worm would be far less operative in an IPv6 background because of its incapability to find hosts to contaminate and thus its incapability to bring about the flooding consequence.

In the RFC 7123 published by Gont & Liu in 2014 specifies the implications of IPv6 on IPv4 Networks and it discuss the methods of filtering the traffic. It also point out the situation occure when there is a tunneling mechanism is present [14].

Journal Article on IPv6 Security by Harith A Dawood in 2012 states about the variuos security holes in IPv6 such as multicast vulnerabilities, extension header vulnerabilities, fragmentation vulnerabilities, etc [15]. The article concludes by mentioning the importance of effective IDS/IPS as the protection against several threats are essential. Also he talks about IPsec availability in IPv6 and how administrators misuse it in configuring.

The IPsec packet format in IPv6 is mostly the same as in IPv4. Figure 6 shows an IPv6 packet where Authentication Header (AH) and Encapsulation Security Payload (ESP) protocols are used. IPv6 AH and ESP extension headers are used to provide authentication and confidentiality to IPv6 packets [3].

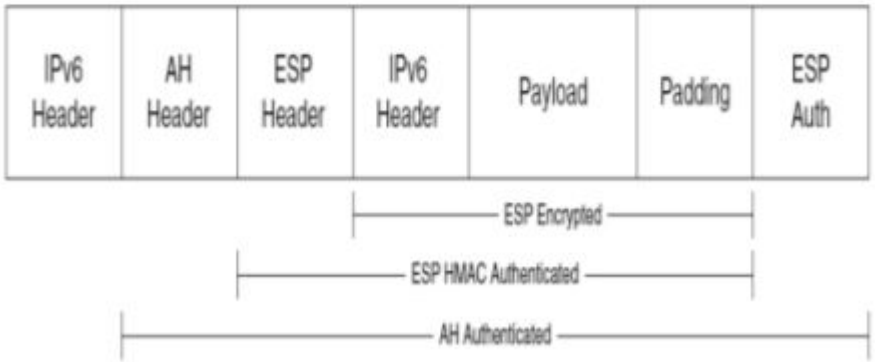


Figure 6: IPv6 Header with IPSec

In the paper, IPv6 Security Issues by Samuel Sotillo states the issues as Denial of Service attacks, Malicious code Distribution, Manin-the-middle attacks, Fragmentation attacks, ARP poisoning and ICMP redirect as well as port scanning to be the most identifierable threats [16]. Also dual stack issues are highlighted as some of the potential vulnerabilities. This paper is totally focused only on threat types and there is no relavent facts on how to detect them. But this provides more insights on ipv6 security issues.

By the paper IPv6 Security Challenges published in Computer Magazine Volume: 42, Issue: 2, Feb.2009 authors shows the effects had on public networks because of the NAT in IPv4 addressing [17]. They spoke about the overhead arised on devices because of the NAT, also they agree on NAT is not a security mechanism. They points out that the IPv6 neworks will provide ene to end connectivity to users and this will deinitely bring lot of security issues.

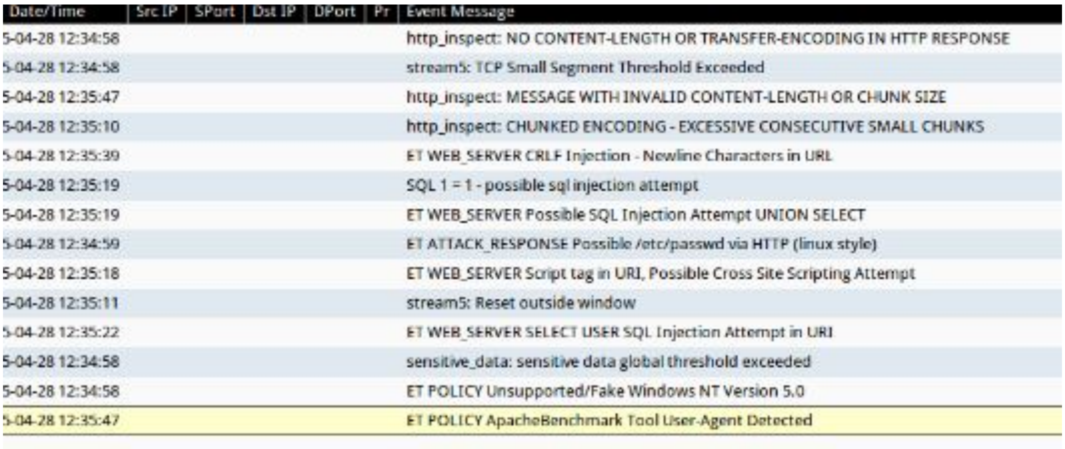
Paper IPv4/IPv6 security and threat comparisons by Emre Durdagi and Ali Buldu ponts out the common threats both versions are faceing [18]. They points attacks related with sniffing, flooding, application layer centric, rogue devices and Man-in-the-middle attacks to be common in both versions while Reconnaissance attacks, IPv6 routing headers attacks, ICMPv6 and multicast attacks, Secure Neihbor Discovery protocol related threats and threats related with transition mechanisms are IPv6 threats. This paper clearly shows how IPv6 can be dangerous because of its hugh address space.

In the article IPv6 security threats and possible solutions by Zagar Drago and Kresimir Grgic describes the possible solutions to several IPv6 related threats [19]. From this article I can get some key points to this thesis as it describes effects on transition mechanisms and generals ways detecting them.

Migration methods and counter-measures are again discussed on the article by John M Chasser in 2010 [20]. On his article transitional methods and their implications are stated and highly recommonds IPv6 security training for archtects and engineers to overcome security implications. When considering traffic for both IPv4 and IPv6 protocols requires inspection by intrusion appliances and firewalls.

This may necessitate use of multiple appliances each with specific rules, or single appliances running both stacks and inspection of all traffic. A major security concern from a standalone site's perspective is ICMPv6 filtering. ICMPv6 messages can be broken into two basic types: informational and error. Error messages should pass through filtering devices while informational messages are dropped as policy allows. ICMPv6 cannot be completely filtered and filtering these messages could result in connection failure problems. These ICMPv6 message should be secured via the SEND protocol [21].

For this theses, Security Onion distribution [9] will be used and the developing community of the distribution confirmed via email that still their distribution is unable in detecting IPv6 correctly(Annexure A). According to the community some parts of Security Onion, Specialy the IPS engine Snort is capable of detecting IPv6 but yet as a whole Security Onion lacks reporting IPv6 related alerts [22]. Figure 7 an image published by SANS organization also illustrate it. In the figure it is clearly visible that the Source and Destination IP addresses and port details are not available. This is because the current data schemes do not support IPv6 data. According to the community those details are saved as integer values on a mysql database where the integer equilent of an IPv6 address cannot be accomodated on that fields. Therefore it is not shown or shown as 0.0.0.0.



Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
5-04-28 12:34:58						http_inspect: NO CONTENT-LENGTH OR TRANSFER-ENCODING IN HTTP RESPONSE
5-04-28 12:34:58						stream5: TCP Small Segment Threshold Exceeded
5-04-28 12:35:47						http_inspect: MESSAGE WITH INVALID CONTENT-LENGTH OR CHUNK SIZE
5-04-28 12:35:10						http_inspect: CHUNKED ENCODING - EXCESSIVE CONSECUTIVE SMALL CHUNKS
5-04-28 12:35:39						ET WEB_SERVER CRLF Injection - Newline Characters in URL
5-04-28 12:35:19						SQL 1 = 1 - possible sql injection attempt
5-04-28 12:35:19						ET WEB_SERVER Possible SQL Injection Attempt UNION SELECT
5-04-28 12:34:59						ET ATTACK_RESPONSE Possible /etc/passwd via HTTP (linux style)
5-04-28 12:35:18						ET WEB_SERVER Script tag in URI, Possible Cross Site Scripting Attempt
5-04-28 12:35:11						stream5: Reset outside window
5-04-28 12:35:22						ET WEB_SERVER SELECT USER SQL Injection Attempt in URI
5-04-28 12:34:58						sensitive_data: sensitive data global threshold exceeded
5-04-28 12:34:58						ET POLICY Unsupported/Fake Windows NT Version 5.0
5-04-28 12:35:47						ET POLICY ApacheBenchmark Tool User-Agent Detected

Figure 7: IPv6 wrong detection in Security Onion, Source:www.sans.org

Also Security Onion utilizers a security monitoring tool named “Sguil” to inspect deep in to packets [23]. But currently it is not supported for IPv6 Traffic, Annexure B shows a mail conversation between myself and the core developer of Sguil, Bamm Visscher. According to Bamm they are not going to upgrade Sguil in near future to support IPv6. Therefore, I had to look for other options bypassing Sguil. First thought was to upgrade Sguil myself, but Bamm insists that it will not be possible for me to do it alone. Then the next was to redirect detected rules to a new system.

When considering the alert process in Security Onion it uses snort one of their IDS engines and according to snort manuals it currently supports IPv6 [24]. But those IPv6 alerts are not correctly shown in “SO” because of the issues mentioned earlier with Sguil. But between the snort and Sguil there is a data spooler called Barnyard2 [25]. According to Barnyard2 manuals it currently supports IPv6. But I may need to recompile it as the version installed in Security Onion does not come with IPv6 support.

Chapter 3 - Research Design

This thesis bases on enabling IPv6 on the Linux distribution - Security Onion (SO) which delivers a lot of tools for network security [9], but main area of interest is Snort, the IDS/ IPS engine of SO. Snort is a free IPS and IDS with open source, is proficient in performing packet recording and real-time traffic analysis of IP networks [10].

3.1 Security Onion

Security Onion is an Ubuntu based Linux distribution containing a set of specific tools for security including Snort, Bro, Suricata, Sguil, Squert, Xplico, NetworkMiner and others [10]. These tools were developed and maintained as independent decision-making tools by separate developers and are freely available under GNU license. But because of the highly specific nature, not all Linux distributions have these tools in their repositories, also their installation from the source code may also be difficult. Therefore, the latest version of Security Onion solves this problem very efficiently and successfully by letting the users access these individual tools in a one place.

3.2 Information Collection

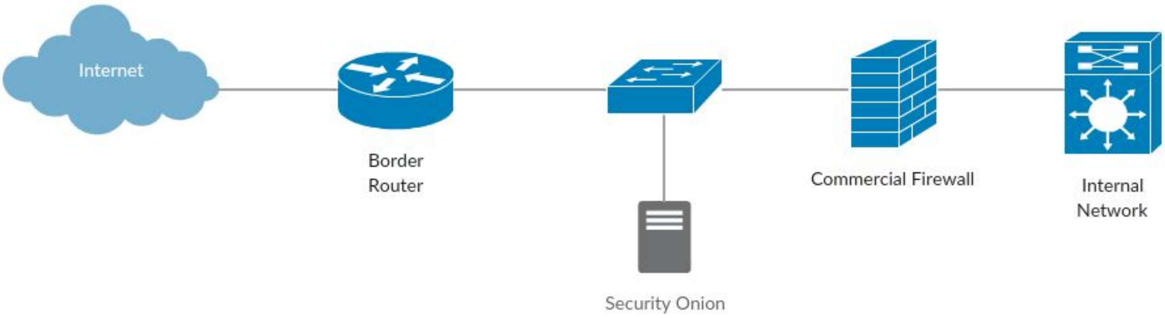


Figure 8: Data Capture Setup

Current real statistics of IPv6 traffic are collected from the University of Kelaniya where I am employed and a test bed was created on a virtual environment with Security Onion installed to check the existing ability in detecting IPv6. As shown in Figure 8 a commercial firewall – Check Point 4800 is currently installed on the network and Check Point IPS reports are collected for the real traffic also for the references.

3.3 Data Analysis

Once the Security Onion and Check Point reports are compared and contrasts to find the capability of IPv6 detecting in real environment. Priority was given to the ICMPv6, DHCPv6 and Unusual traffic generated to single destinations.

3.4 Design of Proposed System

Once the required detection patterns are identified, the corresponding IPv4 rules are transformed into IPv6 detectable form. While doing this, limitations of database occurred, which Security Onion was recording IP addresses as in its full decimal value form and then converting that number to the traditional address form at the retrieval. So if we put a IPv6 address which is extremely bigger, the data field gets overflowed. Therefore, alert storage of Security Onion needed to be changed to match IPv6 traffic, but due to dependencies such as MySQL not having a data type which can facilitate 2^{128} number, it was not possible.

Fine tuning existing Snort detection rules was done according to continuous testing and comparison with the commercial IPS distributions. New Rules had to be created to detect ICMPv6.

In Snort, rules used are in simple and easy to understand rule language. Typically all rules are crated on a single line [24]. Each of these rules have two sections, first is the "rule header" and the second is the "rule options". The rule header, contains the rule's action, protocol , source IP address, destination IP address and net masks ,the source and destination ports details.

The later part of a rule, rule option contains alert messages and data on where the inspection have to be done on packet segments.

Snort Rule have actions with 8 functions; "alert" which will produce an alert expending the selected alert scheme, and log the packet then "log" where packet will only be logged, "pass" that is to ignore the packet flow, activate where an alert is made and then another dynamic rule will be turned on, "dynamic" is a rule which remain idle until it is activated through an activate rule, then it will act as a normal log rule, next "drop" rule will block any requests and pacet will be logged. "Reject" is a rule where it blocks and logs the packet send a TCP reset if its a TCP packet or ICMP port unreachable if its an UDP. Last is "sdrop" where it will block the packet but nothing will be logged.
There are four protocol types seen in protocol session; TCP, UDP, ICMP, and ICMPv6.
IP addresses can be a single address or range of addresses or any and they will define the source and the destinations. Port Numbers are same as IP addresses and also have flexible formation ranges, single and even negation. Direction operator is used to indicate the orientation or route of traffic that a rule is applies to.

Once the Rules are created they will be installed in the Security Onion running rules set and will be tested for detection. As the existing reporting tools are not compatible with IPv6 new system based on Elasticsearch, Logstash, Kibana (ELK stack) [26] will be introduced and new Visualizations are being created. Selection of ELK stack as the reporting tool was done due to its popularity and ability in handling big data in Open Source environment.

3.5 ElasticSearch – Logstash – Kibana

The ELK stack is a software bundle used in big data analytics and are also used for log file correlations [26]. ELK or Elasticsearch, Logstash and Kibana uses java based environment to analyze various logs and are widely used in many industries to handle large data. Along with these three components another application "Filebeats" will be used to collect data and send to ELK stack.

3.5.1 ElasticSearch

Elasticsearch is a analytics engine with rising number of users world-wide in using distributed, and RESTful analyzing and searching [27]. As the main component of the ELK Stack, this can be used to centrally store analytical data so the user can expect to find the unexpected results. Elasticsearch supports many types of searches, it can be structured, unstructured or geographic, metrics or combination. This helps to do a single query in billions of files or logs in a single time, reducing the time and costs for the user. It will help to identify patterns and hidden trends within the data.

3.5.2 Logstash

Logstash is a server-side data processing pipeline which will take multiple of inputs from many sources and instantaneously and convert it to any user defined method and send it to Elasticsearch for analyzing [28].
Logstash has three main parts;
Input: Logstash supports a diversity of input data which is often distributed across many systems and in many layouts. It can pull in events from a many common sources, simultaneously such as logs, web applications, data stocks or any continues data stream.

Filter: Before data goes to a store from source filters will parse events and identify pre-defined fields to build up a structure and transform or converge then to a common type of data format, this is very useful in analysis in accelerated business needs. With the rich and endless library of filters there can be no log file that cannot be parsed.

Output: Even though, Elasticsearch is the final output in this path, Logstash supports a multiple of methods in outputting data, which brings the flexibility in to the methods of searching and sorting making life easier.

3.5.3 Filebeats

Filebeat is a central way of collecting logs or files from multiple of servers or containers which generates logs. This is used as an alternative to connecting thousands of log servers through SSH to collect data [29]. It will not make any downtimes while collecting data and will read logs inline and forward, even any interruption occurred, filebeat will remember where it left off when it comes back.

Filebeat is used to send multiple data in to logstash or elasticsearch directly. If the collector, Logstash is busy when handling high volumes of data, Filebeat will slow the log stream and will be sensitive to Logstash status. As a part of ELK stack Filebeat makes life simple when configuring log importers to Logstash.

3.5.4 Kibana

Kibana is the main component in visualizing data in Elasticsearch and helps to identify and learn patterns in data easily [30]. It has the freedom in selecting the way we shape our data and how we formulate it. All visualizations are interactive and helps the user to build easy access data graphs or charts within seconds.
Kibana is bult-in with numerous types of classical graphs such as line graphs, pie charts, histograms, bar charts etc, also it has the capability to associate with geographical maps and create custom location based data maps.

For this thesis, the newest version of ELK, version 5.2 is used and all the configurations and settings mentioned hereafter are targeted on the 5.2 version. Also all interfaces shown will be only available in the mentioned version.

Therefore, in generally, Filebeat will read the data and sends to Logstash where it will parse it and categorize it to send to Elasticsearch. Once the data reaches Elasticsearch it can be viewed or analyzed by the graphics used in Kibana. As shown in the Figure, this setup will be used as an addition to Security Onion to visualize IPv6 data. Data flow is shown in Figure 9.

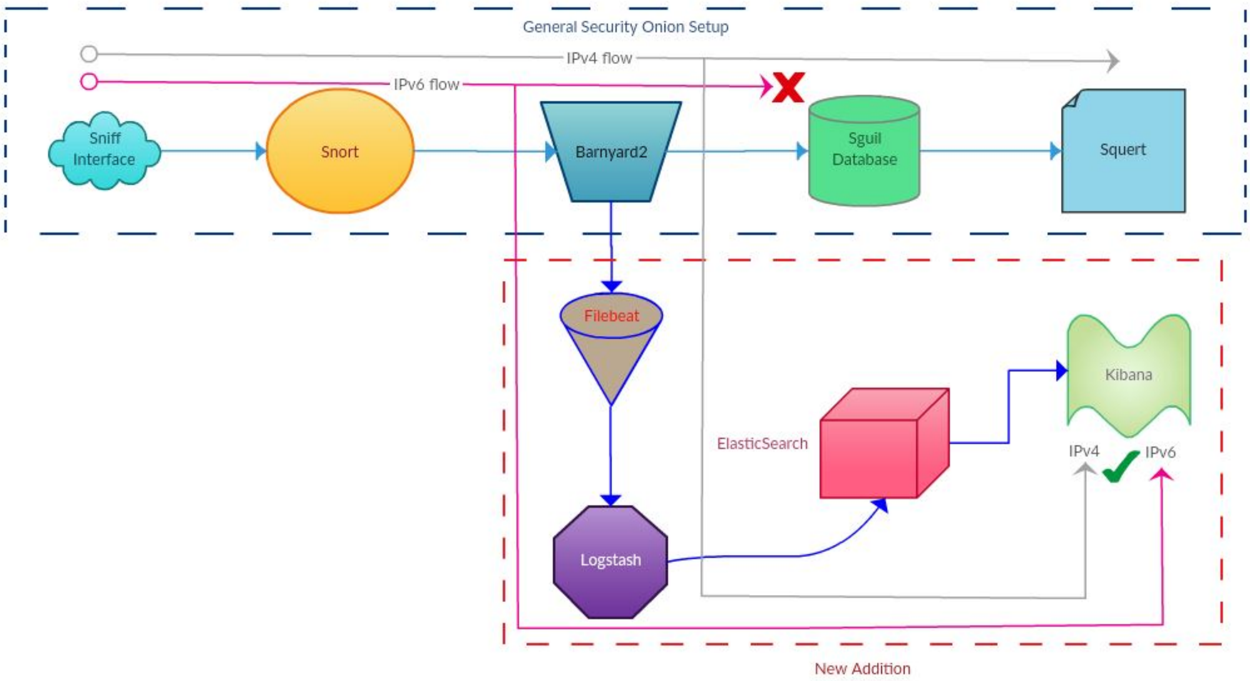


Figure 9: Data Flow of the new system

Once the system is installed IPv4 alerts will be available on both Squert and Kibana and IPv6 will only be available on Kibana Dashboard.

3.6 The Test Bed

Following test bed was created on virtual box as shown in Figure 10 to setup the proposed system and to test it.

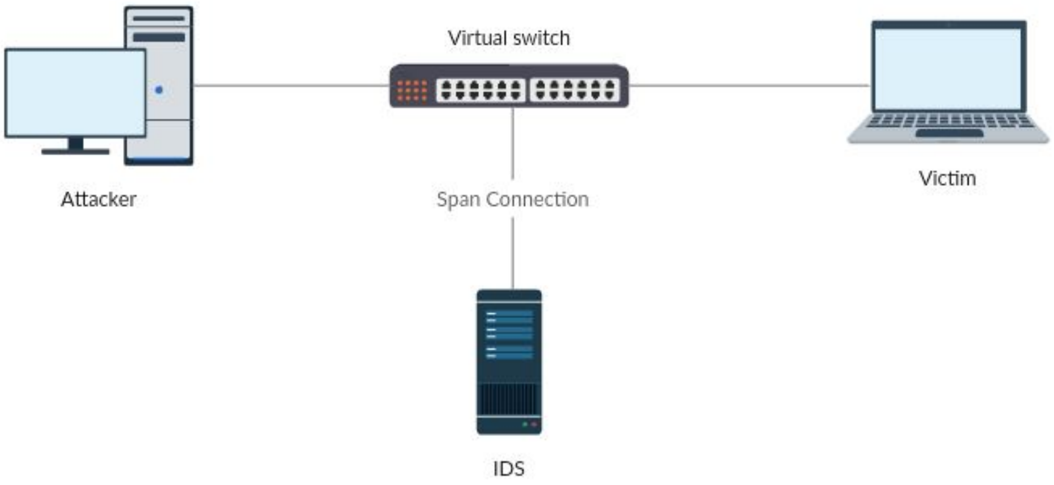


Figure 10 Test Bed Setup

Kali Linux installation was used as the Attacker and a plain Ubuntu server installation was used as the victim. As the IDS, Security Onion installation was used with two interfaces, one is for the span connection and other one as an interface with real world.
Configuration:

Attacker:	2GB RAM 64-bit Kali-Linux
Victim:	512MB RAM 64-bit Ubuntu Server 14.04
IDS:	8GB RAM 64-bit Security Onion

Chapter 4 - Implementation

4.1 Snort Rules creation

As the first part of the methodology target was to create IPv6 detectable rules, therefore existing rules by several developers were referred and new rules were created under those referred guidelines.

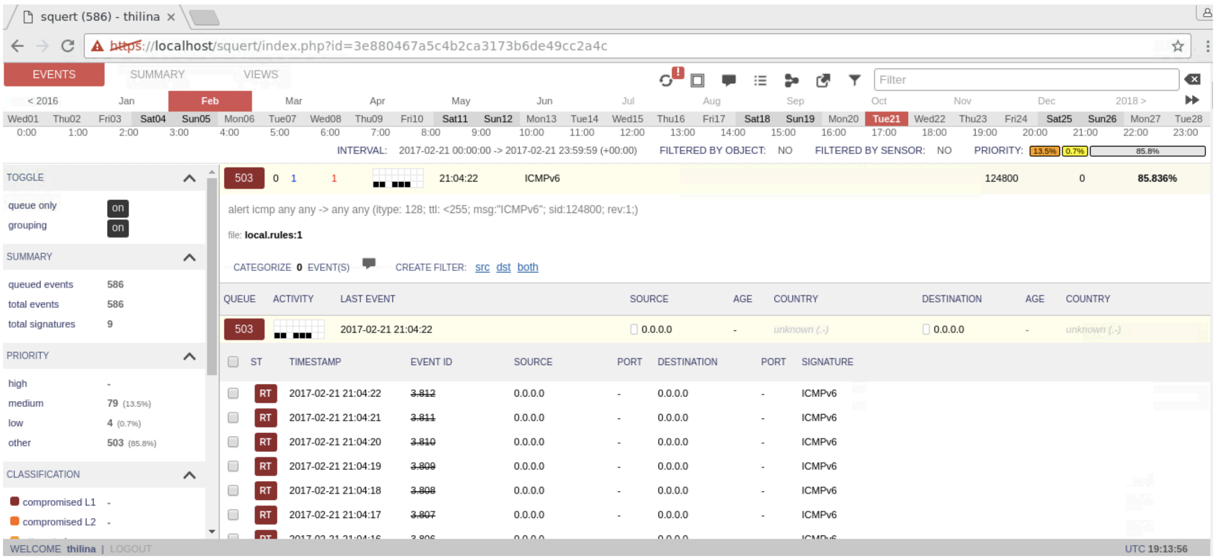
```
alert icmp any any -> any any (itype: 128; ttl: <255; msg:"SO-ELK ICMPv6"; threshold: type threshold, track by_dst,count 5, seconds 60; sid:222200; rev:1;)

alert icmp any any -> any any (itype: 130<>138; ttl: <255; msg:"SO-ELK ICMPv6 - Unusual Neighbor Discovery"; threshold: type threshold, track by_dst,count 5, seconds 60; sid:222201; rev:1;)

alert icmp any any -> any any (itype: 148<>149; ttl: 255; msg:"SO-ELK ICMPv6 - Unusual Neighbor Discovery SEND"; threshold: type threshold, track by_dst,count 5, seconds 60; sid:222202; rev:1;)

alert icmp any any -> any any (itype: 137; ttl: 255; msg:"SO-ELK ICMPv6 - Unusual Neighbor Discovery Redirect"; threshold: type threshold, track by_dst,count 5, seconds 60; sid:222203; rev:1;)

alert icmp any any -> any any (itype: 138; ttl: 255; msg:"SO-ELK ICMPv6 Unusual Neighbor Discovery Router Renumbering"; threshold: type threshold, track by_dst,count 5, seconds 60; sid:222204; rev:1;)

alert icmp any any -> any any (itype: 139<>140; msg:"SO-ELK ICMPv6 Unusual Neighbor Discovery - Node Information"; threshold: type threshold, track by_dst,count 5, seconds 60; sid:222205; rev:1;)

alert icmp any any -> any any (itype: 130<>132; ttl: >1; msg:"SO-ELK ICMPv6 Multicast Listener Discovery with invalid hop limit"; sid:222206; rev:1;)

alert icmp any any -> any any (itype: 143; ttl: >1; msg:"SO-ELK ICMPv6 - Multicast Listener Discovery v2 with invalid hop limit"; sid:222207; rev:1;)

alert icmp any any -> any any (itype: 134; msg:"SO-ELK ICMPv6 - Router Advertisement flooding"; threshold: type threshold, track by_dst, count 5, seconds 1;     sid:222208; rev:1;)

alert icmp any any -> any any (itype: 135; msg:"SO-ELK ICMPv6 - Neighbour Solicitation flooding";  threshold: type threshold, track by_dst, count 20, seconds 1;   sid:222209; rev:1;)

alert icmp any any -> any any (itype: 136;  msg:"SO-ELK ICMPv6 - Neighbour Advertisement flooding"; threshold: type threshold, track by_dst, count 20, seconds 1;    sid:222210; rev:1;)
```

Listing 1: Custom Made Snort Rules

Once the rules are created as Listing 1, they were copied to the local.rules file included in snort configuration of Security Onion. As Security Onion is subjected to auto-updates including rules base, the custom created rules must be included in local.rules otherwise all other will be updated to the latest version of the distribution revoking all changes made by the user.

4.2 Recompile Barnyard2

As the main correlation unit in Security onion is Barnyard2 and by default it does not support IPv6 and as barnyard2 setup didn't know what is IPv6 it sends IPv6 detected alerts to Sguil as unknown IP headers and as a result the IP addresses belongs in the alerts were masked as "0.0.0.0" by Sguil as shown in figure 11.



INTERVAL: 2017-02-21 00:00:00 -> 2017-02-21 23:59:59 (+00:00) FILTERED BY OBJECT: NO FILTERED BY SENSOR: NO PRIORITY: 13.5% 0.7% 85.8%

queued events	586
total events	586
total signatures	9

PRIORITY	
high	-
medium	79 (13.5%)
low	4 (0.7%)
other	503 (85.8%)

503 0 1 1 21:04:22 ICMPv6 124800 0 85.836%

alert icmp any any -> any any (itype: 128; ttl: <255; msg:"ICMPv6"; sid:124800; rev:1;)

file: **local.rules:1**

QUEUE	ACTIVITY	LAST EVENT	SOURCE	AGE	COUNTRY	DESTINATION	AGE	COUNTRY
503		2017-02-21 21:04:22	0.0.0.0	-	*unknown (.-)*	0.0.0.0	-	*unknown (.-)*

ST	TIMESTAMP	EVENT ID	SOURCE	PORT	DESTINATION	PORT	SIGNATURE
RT	2017-02-21 21:04:22	~~3.812~~	0.0.0.0	-	0.0.0.0	-	ICMPv6
RT	2017-02-21 21:04:21	~~3.811~~	0.0.0.0	-	0.0.0.0	-	ICMPv6
RT	2017-02-21 21:04:20	~~3.810~~	0.0.0.0	-	0.0.0.0	-	ICMPv6
RT	2017-02-21 21:04:19	~~3.809~~	0.0.0.0	-	0.0.0.0	-	ICMPv6
RT	2017-02-21 21:04:18	~~3.808~~	0.0.0.0	-	0.0.0.0	-	ICMPv6
RT	2017-02-21 21:04:17	~~3.807~~	0.0.0.0	-	0.0.0.0	-	ICMPv6
RT	2017-02-21 21:04:16	3.806	0.0.0.0		0.0.0.0		ICMPv6

Figure 11: IPv6 detection as 0.0.0.0

By issuing barnyard2 –V command we can inspect the version and capabilities of barnyard, when checked against the default installation it will not show IPv6 as a supporting flag as shown in Listing 2. Therefore, I had to recompile it to get the support. While recompiling there were several dependencies which had to satisfied.

```
thilina@thilina-VirtualBox:~$ barnyard2 -V

   ______     -*> Barnyard2 <*-
  / ,,_  \  Version 2.1.13 (Build 333) TCL
  |o"  )~|  By Ian Firns (SecurixLive): http://www.securixlive.com/
  + '''' + (C) Copyright 2008-2013 Ian Firns <firnsy@securixlive.com>
```

Listing 2: Security Onion Default Barnyard2 Installation

Before recompiling I had to install some dependencies according to the security onion community mailing group [31]. Installation steps are listed below as they were mostly self-research steps and not listed in any documentation online or with the developers,

First, we need to install the dependencies if any using Listing 3,

```
sudo apt-get install tcl8.4-dev libpcap-dev libmysqlclient-dev \
libtool autoconf
```

Listing 3: Install barnyard2 dependencies

After installing dependencies, downloaded the barnyard source code from GitHub as shown in Listing 4.

```
cd /usr/src
sudo git clone https://github.com/firnsy/barnyard2 barnyard2-master
cd barnyard2-master/
./autogen.sh
```

Listing 4: Download Barnyard2 Source

Before installing we may need to point out a library file and then reconfigure the library set as Listing 5.

```
sudo ln -s /usr/include/dumbnet.h /usr/include/dnet.h
sudo ldconfig
```

Listing 5: Library file updating

Once all dependencies and libraries are intact build the barnyard2 and compile it with ipv6 enable flag, here the previous compilation is backed up. This was done while other SO tools were online. Code is listed under Listing 6.

```
./configure --prefix=/usr --enable-ipv6 --with-tcl=/usr/lib/tcl8.4/ -
-with-libpcap-includes=/opt/pfring/lib/                --with-mysql-
libraries=/usr/lib/*-linux-gnu/                        --with-mysql-
includes=/usr/include/mysql/
make
sudo mv /usr/bin/barnyard2 /usr/bin/barnyard2.bak
sudo nsm_sensor_ps-stop --only-barnyard2
sudo make install
sudo nsm_sensor_ps-start --only-barnyard2
```

Listing 6: Recompile Barnyard2

Once barnyard is recompiled, a secondary output was configured to get the data out of Security Onion to be inserted on new analyzing system. This have to be configured according to the interface setup for barnyard in security onion and the settings were saved in a configuration file /etc/nsm/thilina-VirtualBox-eth1/barnyard2-1.conf here ‘thilina-VirtualBox-eth1’ is the name of the sniffing port of the Security Onion Setup.

Listing 7 shows the configuration and the line containing ‘output alert csv’ was manually added with the path for the csv file. According to snort configurations [24], the default csv pattern will be;
timestamp,sig_generator,sig_id,sig_rev,msg,proto,src,srcport,dst,dstport,ethsrc,ethdst,ethlen,tcpflags,tcpseq,tcpack,tcplen,tcpwindow,ttl,tos,id,dgmlen,iplen,icmptype,icmpcode,icmpid,icmpseq
As for the straightforwardness the defaults were used to link with new system.

```
thilina@thilina-VirtualBox:~$ cat /etc/nsm/thilina-VirtualBox-
eth1/barnyard2-1.conf
# barnyard2.conf: auto-generated by NSMnow Administration
config logdir: /nsm/sensor_data/thilina-VirtualBox-eth1
config classification_file: /etc/nsm/thilina-VirtualBox-
eth1/classification.config
config reference_file:      /etc/nsm/thilina-VirtualBox-
eth1/reference.config
config sid_file:            /etc/nsm/thilina-VirtualBox-eth1/sid-
msg.map
config gen_file:            /etc/nsm/thilina-VirtualBox-eth1/gen-
msg.map
config hostname: thilina-VirtualBox-eth1
config interface: eth1
input unified2
output sguil: sensor_name=thilina-VirtualBox-eth1-1 agent_port=8101
#output database: alert, mysql, user=root dbname=snorby
host=127.0.0.1
output alert_syslog: LOG_LOCAL6 LOG_ALERT
output alert_csv: /var/log/barnyard2logs.csv
thilina@thilina-VirtualBox:~$
```

Listing 7 Barnyard Configuration

```
thilina@thilina-VirtualBox:/var/log/sguild$ barnyard2 -V

  ______   -*> Barnyard2 <*-
 / ,,_  \  Version 2.1.14 (Build 337) IPv6 TCL
 |o"  )~|  By Ian Firns (SecurixLive): http://www.securixlive.com/
 + '''' +  (C) Copyright 2008-2013 Ian Firns <firnsy@securixlive.com>
```

Listing 8 After updating Barnyard2

Once the Barnyard IPv6 flag can be seen on the version output listed on Listing 8, now as barnyard is updated Squil rejects IPv6 related alerts as it does not support. This was confirmed by the log outputs of Barnyard. Therefore, the visualizer Squert now shows IPv4 related alerts only.
Anyhow the newly created csv file contains the captured alerts in comma separated format and it includes all IPv4 and IPv6 alerts.

4.3 Installation of ELK – Stack

As mentioned earlier ELK stack is a combination of software which works together to build meaningful analysis on data. This setup is highly depending on Java, therefore I had to install Java on Security Onion as Listing 9.

```
sudo add-apt-repository -y ppa:webupd8team/java
sudo apt-get update
sudo apt-get -y install oracle-java8-installer
```

Listing 9 Install Java on Security Onion

During the installation of Java an error occurred as shown in Listing 10,

```
Preparing to unpack .../oracle-java8-installer_8u121-1~webupd8~1_all.deb ...

oracle-license-v1-1 license could not be presented
try 'dpkg-reconfigure debconf' to select a frontend other than noninteractive

dpkg: error processing archive /var/cache/apt/archives/oracle-java8-
installer_8u121-1~webupd8~1_all.deb (--unpack):
 subprocess new pre-installation script returned error exit status 2
Processing triggers for man-db (2.6.7.1-1ubuntu1) ...
Processing triggers for doc-base (0.10.5) ...
Processing 2 added doc-base files...
Errors were encountered while processing:
 /var/cache/apt/archives/oracle-java8-installer_8u121-1~webupd8~1_all.deb
E: Sub-process /usr/bin/dpkg returned an error code (1)
```

Listing 10 dpkg configuration error

To bypass this error I had to reconfigure dpkg by issuing sudo dpkg-reconfigure debconf Once java is installed, according to Elasticsearch documentation I had to install Elasticsearch, Logstash then Filebeat and finally Kibana.

4.4 Installing Elasticsearch.

As Security Onion is a Ubuntu/debian based operating system, it is easy to install ELK with the help of apt-get. Therefore first I had to include the Elasticsearch repository [32]. Then did an apt update before installing elasticsearch as shown in Listing 11.

```
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo
apt-key add -
sudo apt-get update
sudo apt-get install elasticsearch
```

Listing 11 Install ElasticSearch

After installing, need to change the elasticsearch configuration to restrict outside access to the instance. By default, elasticsearch uses port 9200 and listen on all interfaces. By changing the host configuration on elasticsearch will provide the required security. Therefore edited the /etc/elasticsearch/elasticsearch.yml and changed the network.host to network.host: localhost
Full configuration is amended on Annex C.

After installing we need to add elasticsearch to the startup scripts. This was achieved by issuing command listed in Listing 12

```
sudo update-rc.d elasticsearch defaults 95 10
```

Listing 12: Startup configurtion – Elasticsearch

4.5 Installation of Logstash

Next is to install logstash. As we have already added elasticsearch repository installing logstash is very simple [33] as shown in Listing 13.

```
sudo apt-get install logstash
```

Listing 13: Install Logstash

After installing, configuration should be done to accept data from filebeat, process them and to send data to elasticsearch. These were done by creating three configuration files [34] on /etc/logstash/conf.d/
First file is to capture the input sent from filebeats, 02-beat-input.conf

```
input {

  beats {

    port => 5044

  }

}
```

Listing 14: 02-beats-input.conf

Listed commands in Listing 14 will tell the logstash to listen on port 5044 for any beats input.
Second is the file responsible in filtering the inputs as needed. Following configuration is saved under 10-log-filter.conf

The configuration in Listing 15 will filter any input tagged as ‘log’ by filebeats and it will try to match the mentioned grok pattern and populate the fields stated. Grok pattern stated on match rule for input message was custom made specially to match barnyard csv data. Pattern description as follows;

Grok patterns are the current perfect way of parsing log data to structured and quaryable way which elasticsearch uses to handle data [35] and they have a syntax of %{Syntax:Semantic}. Syntax is how data is represented in the log while Semantic is to tag that data will have.
There are many readymade patterns can be found on logstash developers tree on github (https://github.com/logstash-plugins/logstash-patterns-core/tree/master/patterns).
But to match Snort/barnyard2 logs, specific pattern had to build from scratch. Therefore, using standard patterns [36] and some specific custom patterns mentioned on /etc/logstash/patterns/snort, a new pattern set was created.

```
filter {
  if [type] == "log" {
    grok {
      patterns_dir => ["/etc/logstash/patterns"]
      match => { "message" => "%{TIMESTAMP:barnyard_time}
,%{INT:sig_generator},%{NUMBER:sig_id},%{NUMBER:sig_rev},%{QUOTED
STRING:msg},%{PROTOCOL:proto},%{IP:src_ip},%{INTGR:src_pp},%{IP:d
st_ip},%{INTGR:dst_pport},%{MACC:ethsrc},%{MACC:ethdst},%{BASE16N
UM:ethlen},%{TCPFLG:tcpflags},%{TCPWIN:tcpseq},%{TCPWIN:tcpack},%
{INTGR:tcplenb},%{TCPWIN:tcpwindoww},%{INT:ttl},%{INT:tos},%{INT:
id},?%{INTGR:dgmlen},?%{INT:iplen},%{INTGR:icmptype},%{INTGR:icmp
code},%{INTGR:icmpid},%{INTGR:icmpseq}" }
    }
    mutate {
           replace => [ "barnyard_time","%{barnyard_time} 2017"]
         }
    date {
      match => [ "barnyard_time" , "MM/dd-HH:mm:ss.SSSSSS
YYYY","MM/d-HH:mm:ss.SSSSSS YYYY"]
       }
    geoip {
      source => "src_ip"
      target => "geoip"
      database => "/etc/logstash/GeoLite2-City.mmdb"
      add_field => [ "[geoip][coordinates]",
"%{[geoip][longitude]}" ]
      add_field => [ "[geoip][coordinates]",
"%{[geoip][latitude]}"  ]
    }
    mutate {
      convert => [ "[geoip][coordinates]", "float"]
    }
  }
}
```

Listing 15: Logstash Filter

Contents of /etc/logstash/patterns/snort:

```
INTGR (?:%{POSINT}|\s*|0)
TIMESTAMP %{MONTHNUM2}/%{MONTHDAY}-%{HOUR}:%{MINUTE}:%{SECOND}
TCPFLG (?:%{DATA}|\s*)
MACC
%{BASE16NUM}:%{BASE16NUM}:%{BASE16NUM}:%{BASE16NUM}:%{BASE16NUM}:%{BASE1
6NUM}
TCPWIN (?:%{BASE16NUM}|\s*|0)
PROTOCOL (?:%{WORD}|\s*)
```

Listing 16 /etc/logstash/patterns/snort

In above listing INTGR means any positive number, a zero or a blank. TIMESTAMP refers to the Month number in two digits, date of the day, two digit Hour, Minute and Second with upto 3 decimal points. TCPFLG is some data with characters or none. MACC is the mac address and it had to be mentioned as 6 groups of 2 digit Base16 numbers including zero. There was a readymade pattern for mac addresses but it did not match the contents and the syntax of the mac addresses in log file. Next is TCPWIN which is again a Base16 number, zero or a blank and PROTOCOL is a word or a blank.
Using those custom syntax patterns, a match pattern for csv data was created as,

%{TIMESTAMP:barnyard_time} – according to the custom pattern
%{INT:sig_generator} – Integer Valued signature generator
%{NUMBER:sig_id}- Number to represent signature id
%{NUMBER:sig_rev} – Number to represent the signature revision
%{QUOTEDSTRING:msg} – This contains the Message of an alert
%{PROTOCOL:proto} – Protocol to distinguish traffic
%{IP:src_ip} – Source IP which will match to any IPv4 or IPv6 address
%{INTGR:src_pp} Source Port with type INTGR, errors occurred if INT was used
%{IP:dst_ip} – Destination IP same as src_ip
%{INTGR:dst_pport} – Destination Port same as src_pp
%{MACC:ethsrc} – Source Ethernet address
%{MACC:ethdst} – Destination Ethernet address
%{BASE16NUM:ethlen} - Ethernet frame length
%{TCPFLG:tcpflags} – TCP Flags with any of A,F,P,R,S,U,2,1,*,+,! values [24]
%{TCPWIN:tcpseq} – TCP sequence number
%{TCPWIN:tcpack} – TCP acknowledge number
%{INTGR:tcplenb} – TCP length
%{TCPWIN:tcpwindoww} – TCP window size
%{INT:ttl} – Time to live as of IP header
%{INT:tos} – Type of Service as of the IP header
%{INT:id} – ID of the packet
?%{INTGR:dgmlen} – length of the datagram
?%{INT:iplen} – IP length as in IP header
%{INTGR:icmptype} – Type of the ICMP as of IP header
%{INTGR:icmpcode} – ICMP code
%{INTGR:icmpid} – ICMP ID as of IP header
%{INTGR:icmpseq} – ICMP sequence number

Then the matched timestamp, barnyard_time was again synced with system time to keep the integrity of data in visualizer. While doing that, year of the evet had to be manually added as it was not supporting by the snort output.

After the filter a geoip match was created to match the source IP location and look for the country details [37]. Geoip database was downloaded from the internet for free from http://dev.maxmind.com/geoip/geoip2/geolite2/

Then the third file 30-elasticsearch-output.conf was made with the following commands on Listing 17, they will send the data to elasticsearch instance which is listening on port 9200.

```
output {
  elasticsearch {
    hosts => ["localhost:9200"]
    sniffing => true
    manage_template => false
    index => "%{[@metadata][beat]}-%{+YYYY.MM.dd}"
    document_type => "%{[@metadata][type]}"
  }
}
```

Listing 17 30-elasticsearch-output.conf

Once configurations are placed logstash instance was also marked as a startup script by entering

```
sudo update-rc.d logstash defaults 96 9
```

Listing 18: Auto Startup – logstash

4.6 Installation of Filebeat

As I have earlier added elasticsearch repository installing filebeat can be done as Listing 19 [38].

```
sudo apt-get install filebeat
```

Listing 19: Install Filebeat

After installing, configuration file for filebeat was edited to add the barnyard2 csv log file as a input and logstash as the output. Edited file located at /etc/filebeat/filebeat.yml as shown in Listing 20

```
- input_type: log

  # Paths that should be crawled and fetched. Glob based
paths.
  paths:
    - /var/log/barnyard2logs.csv


#----------------- Elasticsearch output ---------------------
#output.elasticsearch:
  # Array of hosts to connect to.
 # hosts: ["localhost:9200"]

  # Optional protocol and basic auth credentials.
  #protocol: "https"
  #username: "elastic"
  #password: "changeme"

#-------------------- Logstash output -----------------------
output.logstash:
  # The Logstash hosts
  hosts: ["localhost:5044"]
  bulk_max_size: 4096
```

Listing 20 filebeat.yml partial configuration

Full configuration is amended on Annex D.

On the top of the file, it is the prospectors section, which define prospectors that tells which log files should be stashed and how they should be processed. Prospectors are indicated by the - Minus character. As I am using a single file, modified the prospector to include the log file path. This will direct Filebeat to send data from barnyard2 csv to Logstash. Then changed the input type to log. This type should be the exact as mentioned in the logstash filter file.

Then, below the output section, commented out the line with elasticsearch, which indicates the Elasticsearch output upto the logstash section. On the Logstash output section line with output.logstash was uncommented by deleting the # and uncommented the hosts: ["localhost:5044"] line. As I am running logstash on the same computer I used localhost with its listening port 5044. Next, just after the host line added a line bulk_max_size: 4096 with the same indent to tell filebeat that it will be getting a large data to process. Defaults are 2048.

Then filebeat was also added to startup list as Listing 21

```
sudo update-rc.d filebeat defaults 97 9
```

Listing 21: Auto Startup – filebeat

4.7 Installation of Kibana

Visualizer of the ELK kibana was also installed using apt-get as shown in Listing 22.

```
sudo apt-get install kibana
```

Listing 22 Install Kibana

After installing, edited the configuration in /etc/kibana/kibana.yml to support the existing system. Therefore, uncommented the following to tell kibana to listen only to Localhost incoming requests using Listing 23. This will restrict others in the network to access kibana through browsers.

```
server.host: "localhost"
```

Listing 23: Kibana Host

Then kibana was also added to the startup scripts as Listing 24,

```
sudo update-rc.d kibana defaults 95 10
```

Listing 24: Auto Startup - Kibana

After Installing elasticsearch, Logstash, filebeats and kibana server was restarted. Then elasticsearch needed to be loaded with filebeat template. A custom made template can be found on github, therefore downloaded it (Listing 25) and uploaded it (Listing 26) to elasticsearch by

```
curl -O https://gist.githubusercontent.com/thisismitch/3429023
e8438cc25b86c/raw/d8c479e2a1adcea8b1fe86570e42abab0f10f364/filebeat-
index-template.json
```

Listing 25: Download Filebeat template

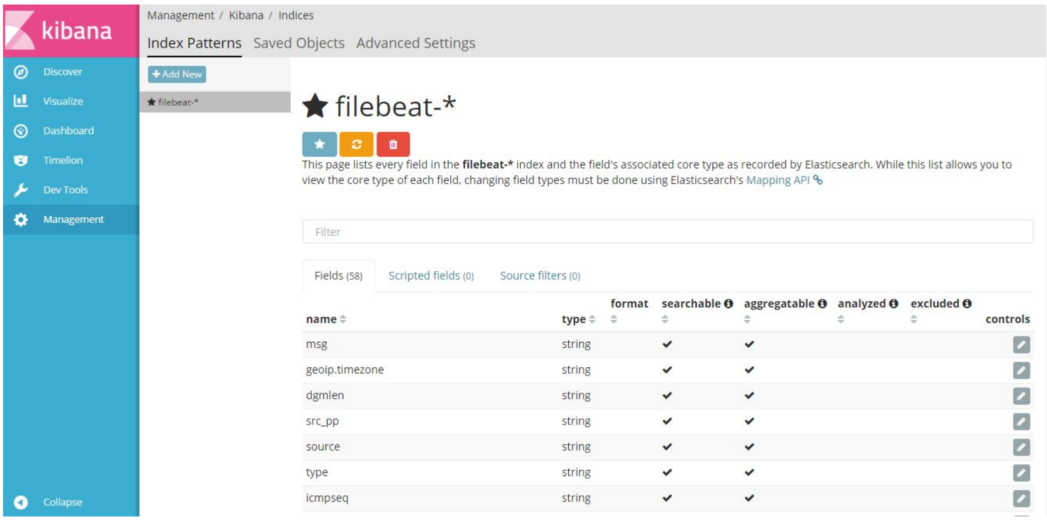
and

```
curl -XPUT 'http://localhost:9200/_template/filebeat?pretty' -
d@filebeat-index-template.json
```

Listing 26: Upload Filebeat template to ElasticSearch

Once its successfully updated, elasticsearch will send a message as acknowledged.
Next is to configure Kibana frontend to utilize filebeat template. This was done by accessing the kibana from default browser of the Security Onion as http://localhost:5601.

On the Management settings 'filebeat-*' was given as the index pattern and it automatically loads the configured pattern fields through logstash as the default index as shown in the figure 12.



★ filebeat-*

This page lists every field in the **filebeat-*** index and the field's associated core type as recorded by Elasticsearch. While this list allows you to view the core type of each field, changing field types must be done using Elasticsearch's Mapping API

name	type	format	searchable	aggregatable	analyzed	excluded	controls
msg	string		✓	✓			
geoip.timezone	string		✓	✓			
dgmlen	string		✓	✓			
src_pp	string		✓	✓			
source	string		✓	✓			
type	string		✓	✓			
icmpseq	string		✓	✓			

Figure 12: Kibana Default Index Patterns

Chapter 5 - Testing and Evaluation

5.1 Initial Testing

From the attacker machine as shown in figure 13, two continues pings were issued against the victim and the pings were based on IPv4 and IPv6 schemes respectively. Then the output of Security Onion Default visualizer 'Squert' was observed. Then the Kibana Discover was observed to check whether it receives the alerts.

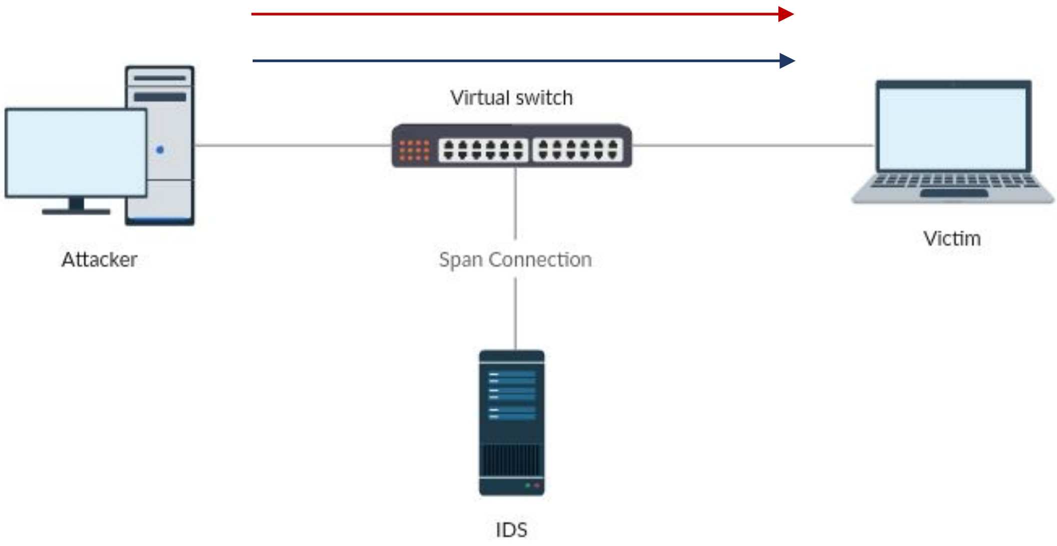


Figure 13: Testing

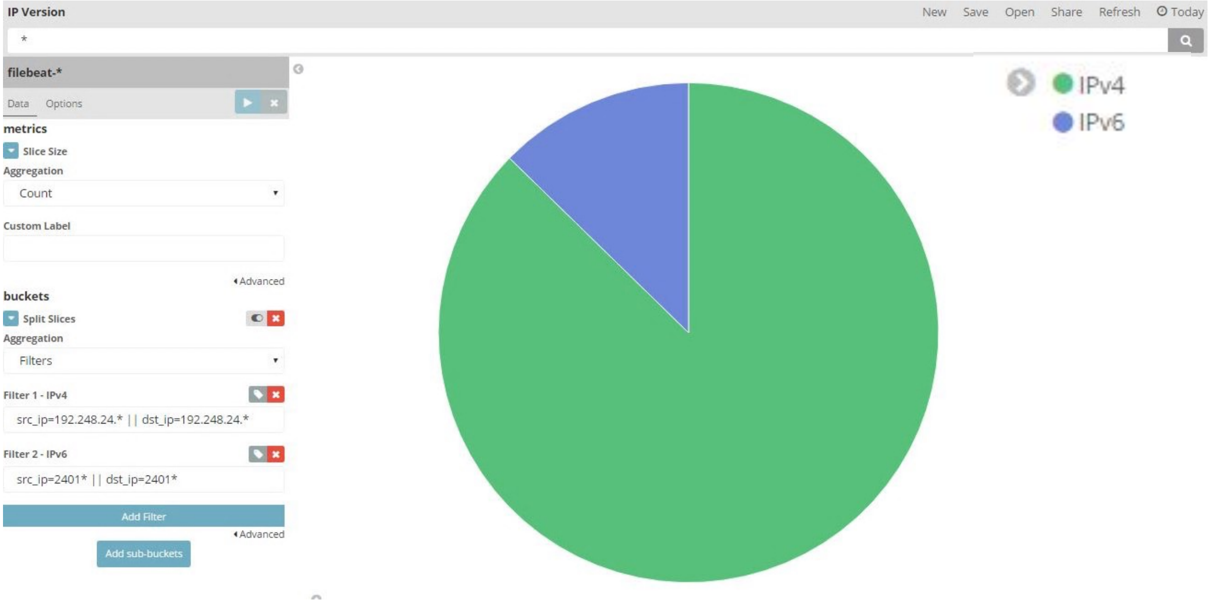
Initial testing is done by doing ping tests for both ICMP and ICMPv6 and NMAP sysc scans to test TCP and UDP open ports scans against the victim. The span connection to the IDS setup will send all traffic to the ELK-SO server where it should detect the pings and port scans.

5.2 Kibana Visualizations

Once confirmed that Kibana was showing alerts, on the visualize tab created the following graphs to show the alerts more graphically.

For this graph in Figure 14, two separate filters were used for IPv4 and IPv6 based on their src_ip and dst_ip. As these were fully customized charts values used to filter were completely dependent on the environment.

5.2.1 Graph 1: IP version Pie Chart



IP Version

*

filebeat-*

metrics

Slice Size

Aggregation: Count

Custom Label

buckets

Split Slices

Aggregation: Filters

Filter 1 - IPv4: src_ip=192.248.24.* || dst_ip=192.248.24.*

Filter 2 - IPv6: src_ip=2401* || dst_ip=2401*

Figure 14: IP version Pie Chart

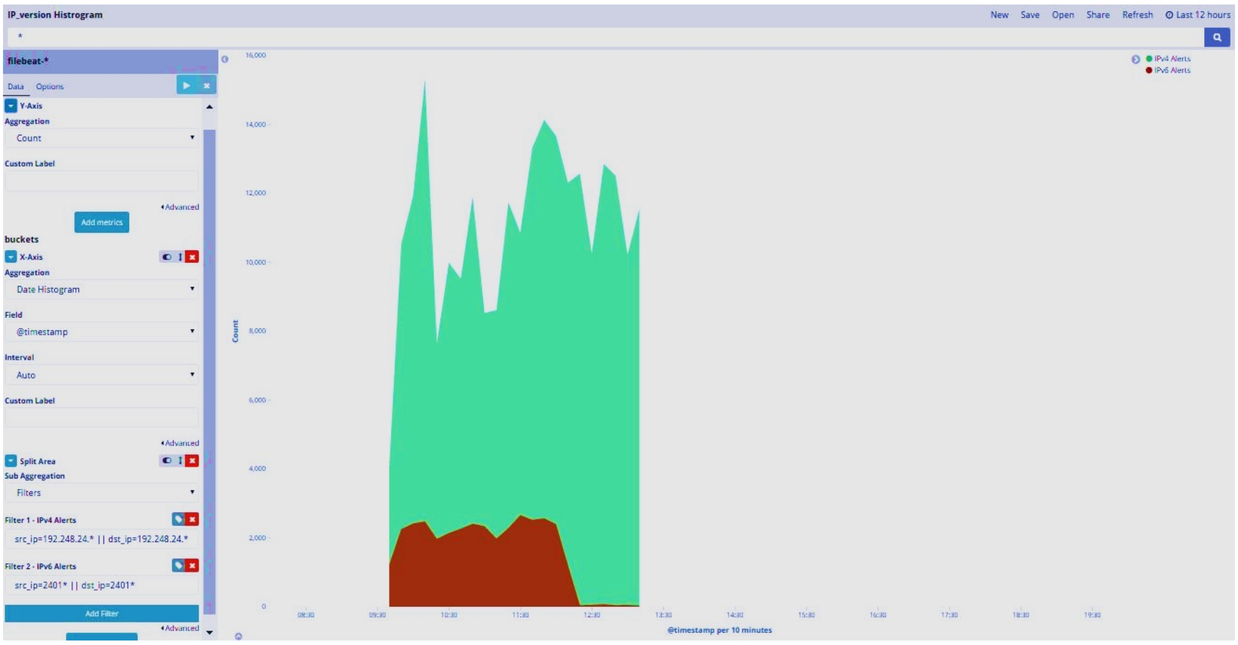
Metrics:

- Slice Size
  - Aggregation: count

Buckets:

- Split Slices
  - Aggregation: Filters
  - Filter 1: src_ip=192.248.24.* || dst_ip=192.248.24.*
  - Filter 1 Label: IPv4
  - Filter 2: src_ip=2401* || dst_ip=2401*
  - Filter 2 Label: IPv6

5.2.2 Graph 2: IP version Histogram



IP_version Histrogram

*

filebeat-*

Y-Axis

Aggregation: Count

Custom Label:

buckets

X-Axis

Aggregation: Date Histogram

Field: @timestamp

Interval: Auto

Custom Label:

Split Area

Sub Aggregation: Filters

Filter 1 - IPv4 Alerts: src_ip=192.248.24.* || dst_ip=192.248.24.*

Filter 2 - IPv6 Alerts: src_ip=2401* || dst_ip=2401*

Figure 15: IP version Histogram

Histogram in Figure 15 was created as a area chart with the same filters as the IP version pie chart but against the timestamp. Even though there is a Pie Chart representing the IP version divide, we need the histogram to distinguish traffic against time to analyze the usage along time where the pie chart only represents the whole number of version distribution.

Metrics:

Y-Axis

Aggregation: count

Buckets:

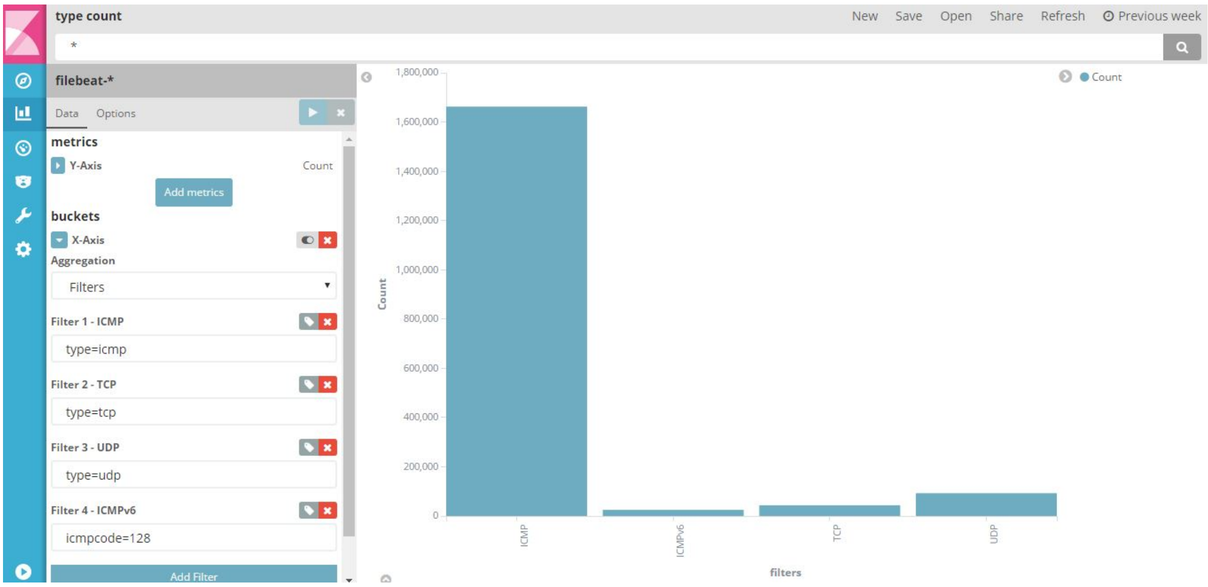
X-Axis

Aggregation: Date Histogram
Field: @timestamp
Interval: Auto
Order: Descending

Split Area

Sub Aggregation: Filters
Filter 1: src_ip=192.248.24.* || dst_ip=192.248.24.*
Filter 1 Label: IPv4 Alerts
Filter 2: src_ip=2401* || dst_ip=2401*
Filter 2 Label: IPv6 Alerts

5.2.3 Graph 3: Protocol Type Count



metrics

Y-Axis Count

buckets

X-Axis

Aggregation

Filters

Filter 1 - ICMP

type=icmp

Filter 2 - TCP

type=tcp

Filter 3 - UDP

type=udp

Filter 4 - ICMPv6

icmpcode=128

Figure 16: Type Count

To check how much alerts were received targeting ICMP, TCP, UDP and ICMPv6 was graphed against count as shown in Figure 16 and to check ICMPv6 a filter was created to extract packets with icmpcode 128 [39] and ignored any other icmp types related to IPv6.

Metrics:

Y-Axis

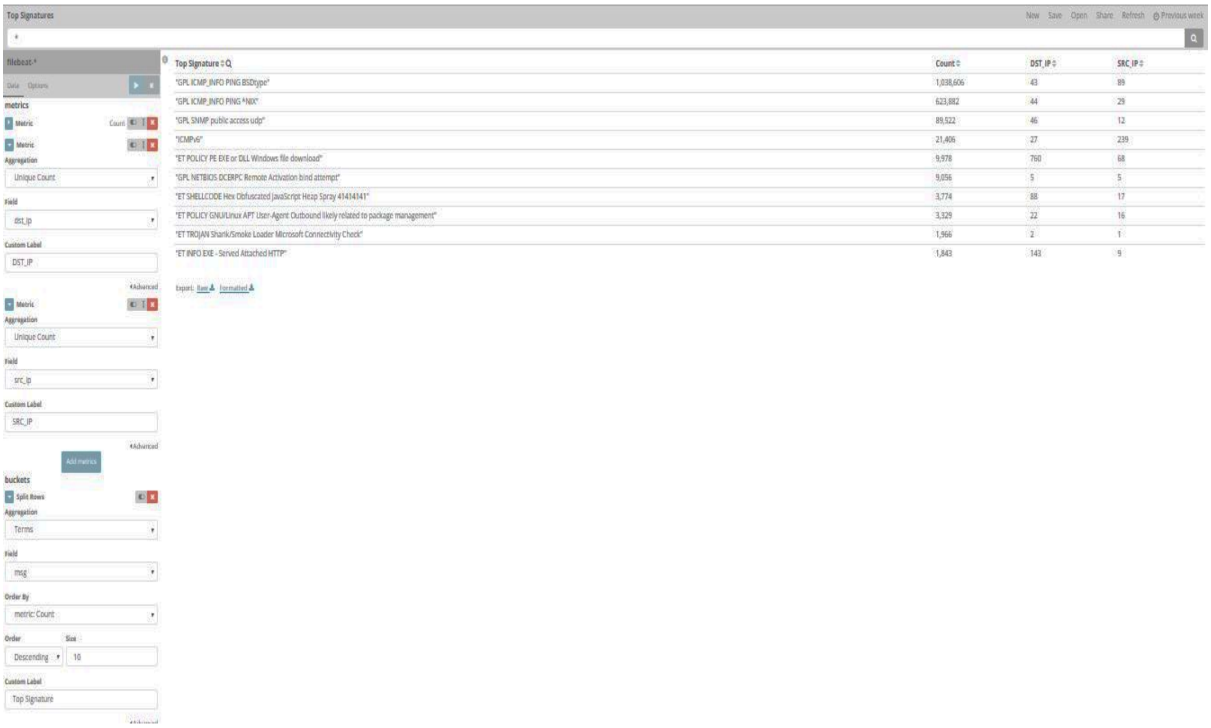
Aggregation: count

Buckets:

X-Axis

Aggregation: Filters
Filter 1: type=icmp, Label: ICMP
Filter 2: type=tcp, Label: TCP
Filter 3: type=udp, Label: UDP
Filter 4: icmpcode=128, Label: ICMPv6

5.2.4 Graph 4: Top Signatures



metrics
Metric: Count
Metric
Aggregation: Unique Count
Field: dst_ip
Custom Label: DST_IP
Metric
Aggregation: Unique Count
Field: src_ip
Custom Label: SRC_IP
buckets
Split Rows
Aggregation: Terms
Field: msg
Order By: metric: Count
Order: Descending
Size: 10
Custom Label: Top Signature

Top Signature	Count	DST_IP	SRC_IP
"GPL ICMP_INFO PING BSDtype"	1,038,606	43	89
"GPL ICMP_INFO PING *NIX"	623,882	44	29
"GPL SNMP public access udp"	89,522	46	12
"ICMPv6"	21,406	27	239
"ET POLICY PE EXE or DLL Windows file download"	9,978	760	68
"GPL NETBIOS DCERPC Remote Activation bind attempt"	9,056	5	5
"ET SHELLCODE Hex Obfuscated JavaScript Heap Spray 41414141"	3,774	88	17
"ET POLICY GNU/Linux APT User-Agent Outbound likely related to package management"	3,329	22	16
"ET TROJAN Shark/Smoke Loader Microsoft Connectivity Check"	1,966	2	1
"ET INFO EXE - Served Attached HTTP"	1,843	143	9

Figure 17: Top Signatures

Top 10 signatures were filtered along with the number of count and number of Destination IP's and Source IP's as shown in Figure 17. Here the data considered are in dual stack.

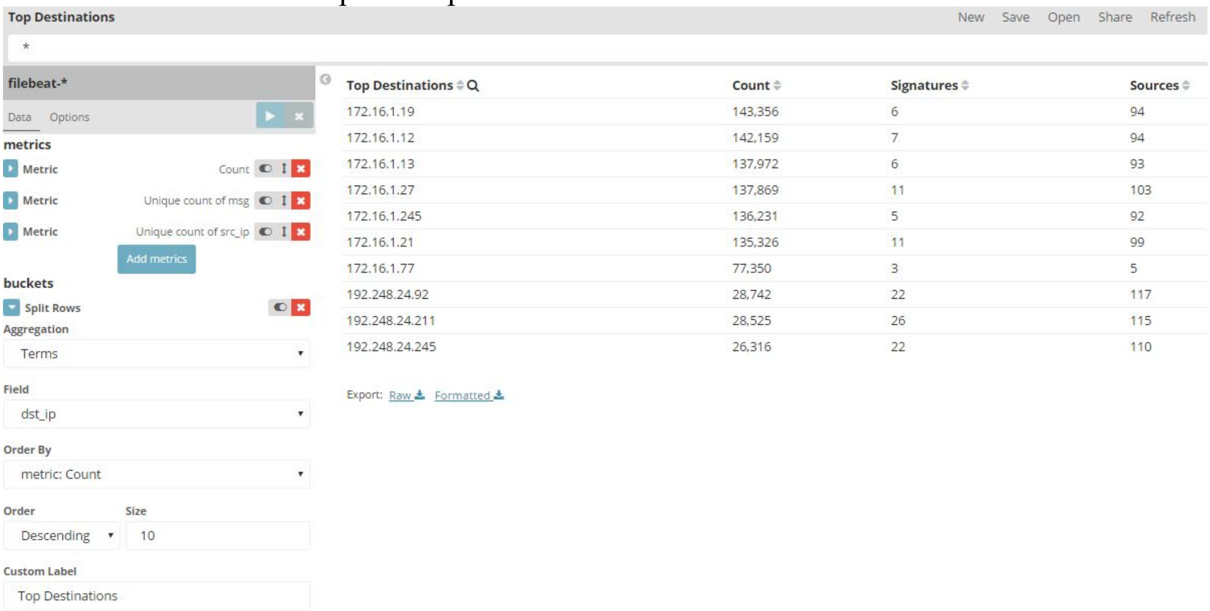
Metrics:

Metric 1
Aggregation: count
Metric 2
Aggregation: Unique Count
Field: dst_ip
Custom Label: DST_IP
Metric 3
Aggregation: Unique Count
Field: src_ip
Custom Label: SRC_IP

Buckets:

Split Rows 1
Aggregation: Terms
Field: msg
Order By: metric:count
Order: Descending
size: 10
Custom Label: Top Signatures

5.2.5 Graph 5: Top Destinations



metrics

Metric Count

Metric Unique count of msg

Metric Unique count of src_ip

buckets

Split Rows

Aggregation: Terms

Field: dst_ip

Order By: metric: Count

Order: Descending

Size: 10

Custom Label: Top Destinations

Top Destinations	Count	Signatures	Sources
172.16.1.19	143,356	6	94
172.16.1.12	142,159	7	94
172.16.1.13	137,972	6	93
172.16.1.27	137,869	11	103
172.16.1.245	136,231	5	92
172.16.1.21	135,326	11	99
172.16.1.77	77,350	3	5
192.248.24.92	28,742	22	117
192.248.24.211	28,525	26	115
192.248.24.245	26,316	22	110

Figure 18: Top Destinations

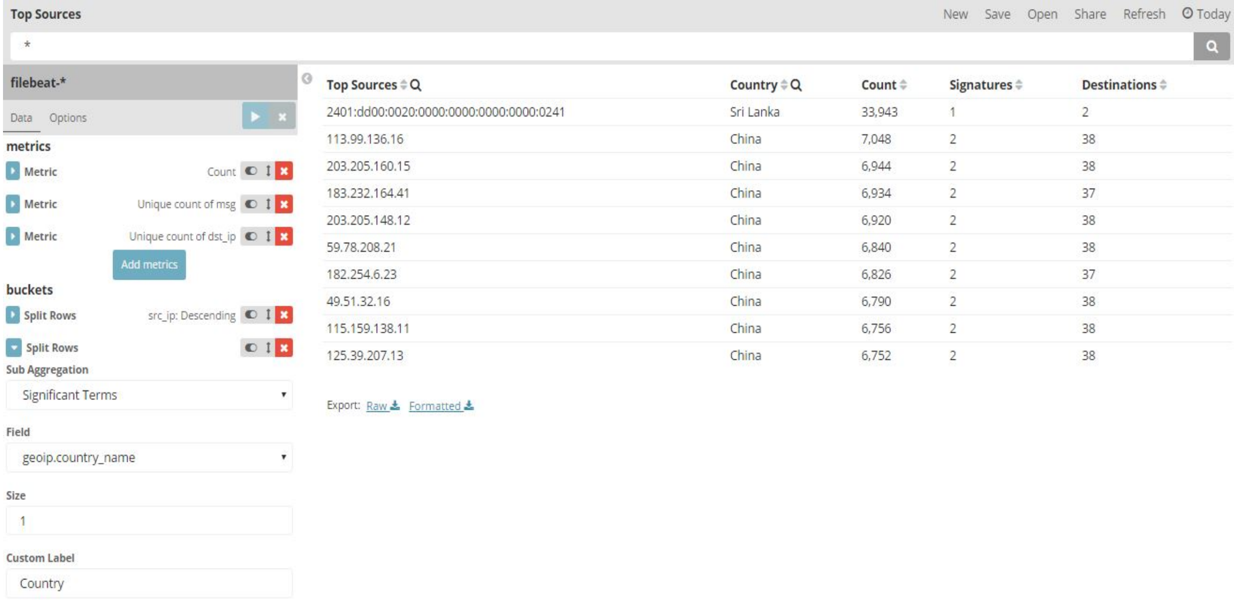
To illustrate top destinations as Figure 18 a table view was created with 10 top most hit destinations along with the numbers of hits and number of signatures matched. Number of sources affected were also calculated.

Metrics:

Metric 1
    Aggregation: count

Metric 2
    Aggregation: Unique Count
    Field: msg
    Custom Label: Signatures
Metric 3
    Aggregation: Unique Count
    Field: src_ip
    Custom Label: Sources
Buckets:
Split Rows 1
    Aggregation: Terms
    Field: dst_ip
    Order By: metric:count
    Order: Descending
    size: 10
    Custom Label: Top Destinations

5.2.6 Graph 6: Top Sources



Top Sources

*

filebeat-*

metrics

Metric Count

Metric Unique count of msg

Metric Unique count of dst_ip

buckets

Split Rows src_ip: Descending

Split Rows

Sub Aggregation: Significant Terms

Field: geoip.country_name

Size: 1

Custom Label: Country

Top Sources	Country	Count	Signatures	Destinations
2401:dd00:0020:0000:0000:0000:0000:0241	Sri Lanka	33,943	1	2
113.99.136.16	China	7,048	2	38
203.205.160.15	China	6,944	2	38
183.232.164.41	China	6,934	2	37
203.205.148.12	China	6,920	2	38
59.78.208.21	China	6,840	2	38
182.254.6.23	China	6,826	2	37
49.51.32.16	China	6,790	2	38
115.159.138.11	China	6,756	2	38
125.39.207.13	China	6,752	2	38

Export: Raw Formatted

Figure 19: Top Sources

Top sources as Figure 19 were tabulated with the hit count. Aggregated Number of Signature and Number of Destinations were also added. As an additional data country of origin for the source IP address was also listed.

Metrics:

Metric 1

Aggregation: count

Metric 2

Aggregation: Unique Count
Field: msg
Custom Label: Signatures

Metric 3

Aggregation: Unique Count
Field: dst_ip
Custom Label: Destinations

Buckets:

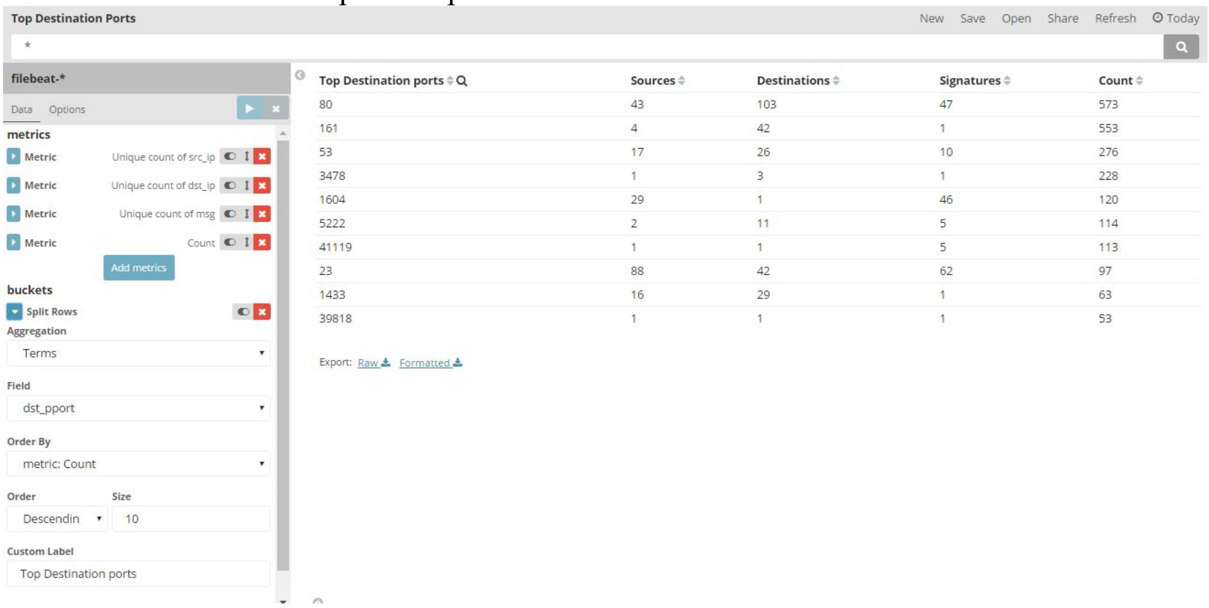
Split Rows 1

Aggregation: Terms
Field: src_ip
Order By: metric:count
Order: Descending
size: 10
Custom Label: Top Sources

Split Rows 2

Sub Aggregation: Significant Terms
Field: geoip.country_name
size: 1
Custom Label: Country

5.2.7 Graph 7: Top Destination Ports



Top Destination Ports

metrics
Metric Unique count of src_ip
Metric Unique count of dst_ip
Metric Unique count of msg
Metric Count

buckets
Split Rows
Aggregation: Terms
Field: dst_pport
Order By: metric: Count
Order: Descendin
Size: 10
Custom Label: Top Destination ports

Top Destination ports	Sources	Destinations	Signatures	Count
80	43	103	47	573
161	4	42	1	553
53	17	26	10	276
3478	1	3	1	228
1604	29	1	46	120
5222	2	11	5	114
41119	1	1	5	113
23	88	42	62	97
1433	16	29	1	63
39818	1	1	1	53

Figure 20 Top Destination Ports

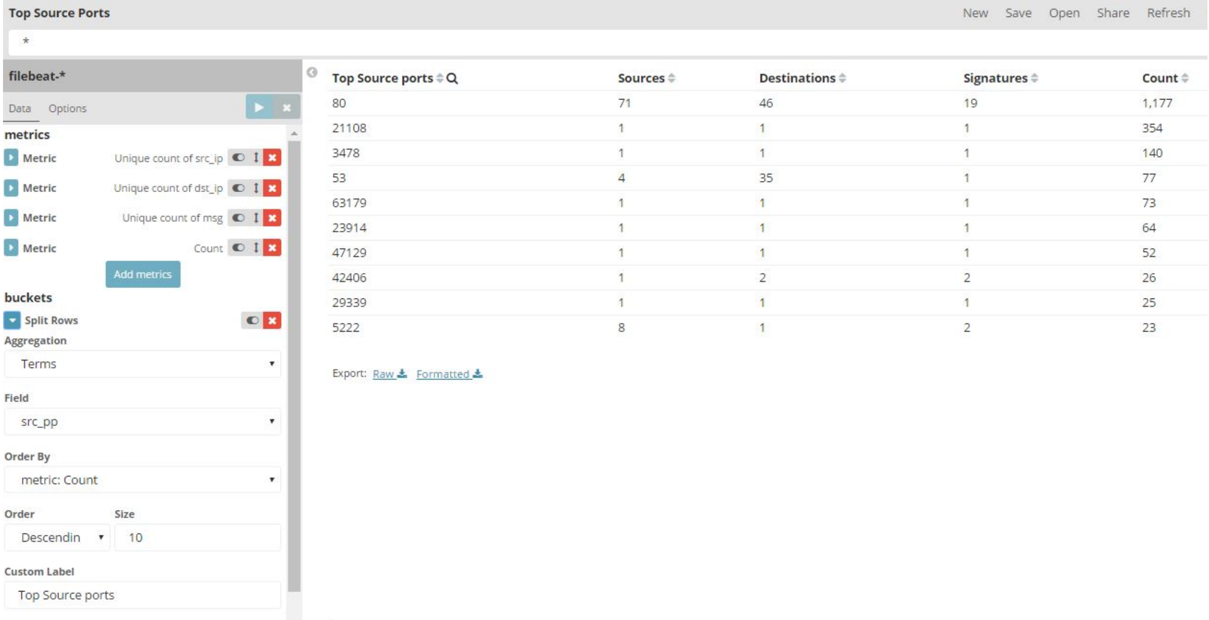
Ten top most hit destination ports were filtered along with the number or sources and destinations and the signatures and populated on a data table as in Figure 20. Using these graphs it can be shown that the most hit ports such as scan attacks.

Metrics:
Metric 1
Aggregation: count
Metric 2
Aggregation: Unique Count
Field: msg
Custom Label: Signatures
Metric 3
Aggregation: Unique Count
Field: dst_ip
Custom Label: Destinations
Metric 4
Aggregation: Unique Count
Field: src_ip
Custom Label: Sources

Buckets:
Split Rows
Aggregation: Terms
Field: dst_pport

Order By: metric:count
Order: Descending
size: 10
Custom Label: Top Destination Ports

5.2.8 Graph 8: Top Source Ports



*

filebeat-*

metrics
Metric Unique count of src_ip
Metric Unique count of dst_ip
Metric Unique count of msg
Metric Count

buckets
Split Rows
Aggregation: Terms
Field: src_pp
Order By: metric: Count
Order: Descendin
Size: 10
Custom Label: Top Source ports

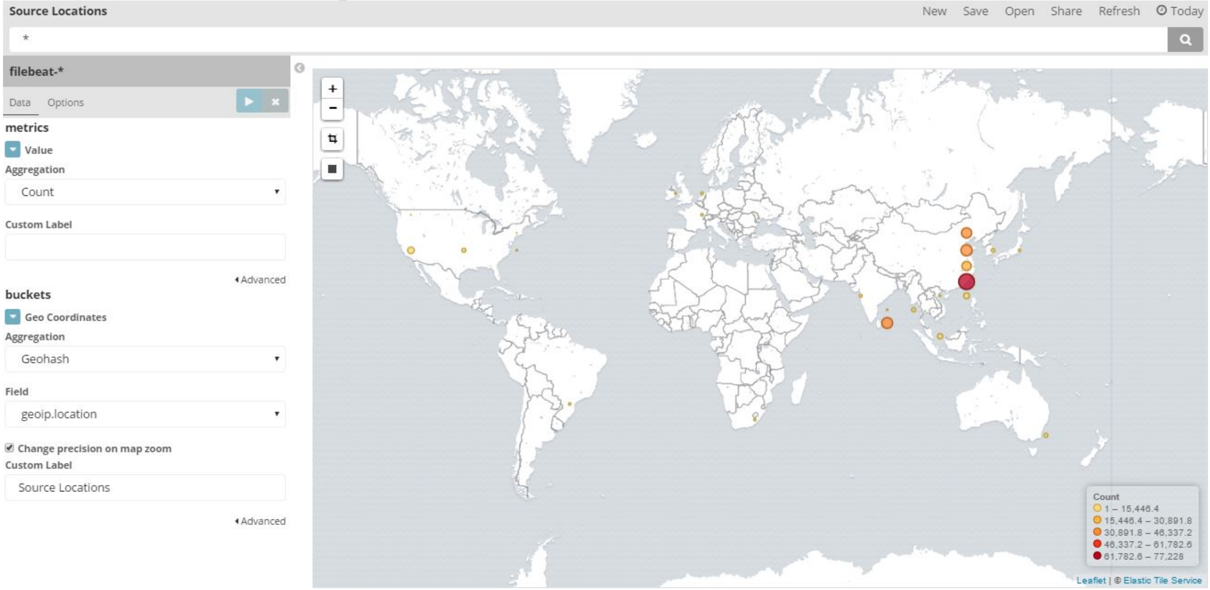
Top Source ports	Sources	Destinations	Signatures	Count
80	71	46	19	1,177
21108	1	1	1	354
3478	1	1	1	140
53	4	35	1	77
63179	1	1	1	73
23914	1	1	1	64
47129	1	1	1	52
42406	1	2	2	26
29339	1	1	1	25
5222	8	1	2	23

Figure 21 Top Source Ports

Highest hit source ports were also listed on a data table with the count and the numbers of unique destinations, sources and signatures as Figure 21.

Metrics:
    Metric 1
        Aggregation: count
    Metric 2
        Aggregation: Unique Count
        Field: msg
        Custom Label: Signatures
    Metric 3
        Aggregation: Unique Count
        Field: dst_ip
        Custom Label: Destinations
    Metric 4
        Aggregation: Unique Count
        Field: src_ip
        Custom Label: Sources
Buckets:
    Split Rows
        Aggregation: Terms
        Field: src_pp
        Order By: metric:count
        Order: Descending
        size: 10
        Custom Label: Top Source Ports

5.2.9 Graph 9: Source Locations



metrics

Value

Aggregation: Count

Custom Label

buckets

Geo Coordinates

Aggregation: Geohash

Field: geoip.location

☑ Change precision on map zoom

Custom Label: Source Locations

Count
- ● 1 – 15,446.4
- ● 15,446.4 – 30,891.8
- ● 30,891.8 – 46,337.2
- ● 46,337.2 – 61,782.6
- ● 61,782.6 – 77,228

Figure 22 Source Locations

Using the geo ip feature, an interactive tile map was created with all alert locations as Figure 22.

Metrics:

Value

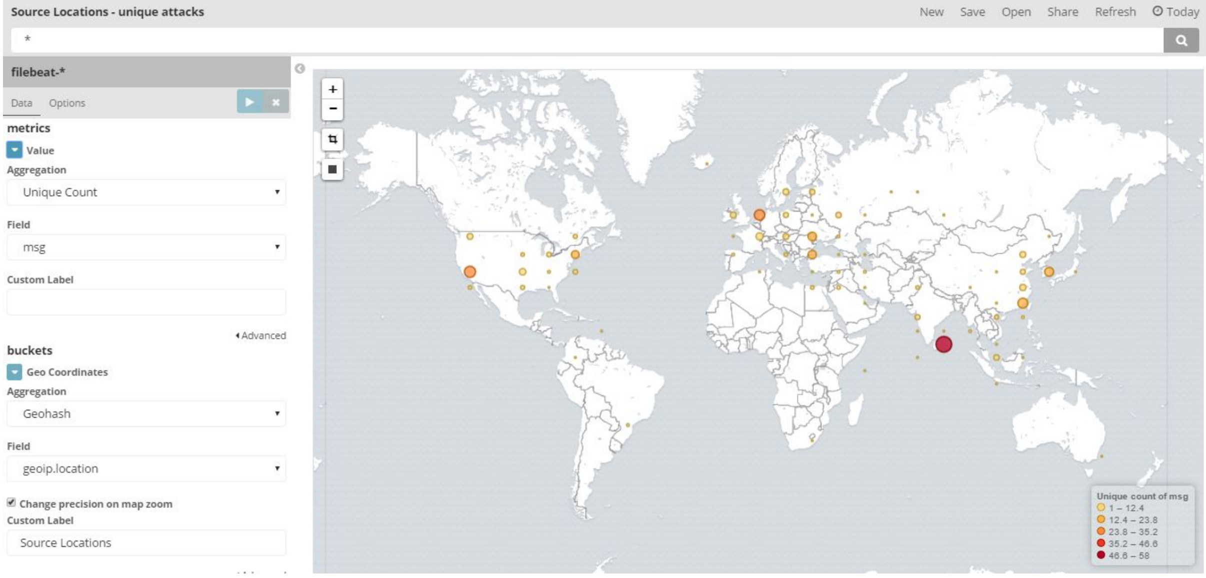
Aggregation: Count

Buckets:

Geo Coordinates

Aggregation: Geohash
Field: geoip.location
Change precision on map zoom: ticked
Custom Label: Source Locations

5.2.10 Graph 10: Source Locations - Unique Attacks



metrics

Value

Aggregation: Unique Count

Field: msg

Custom Label

buckets

Geo Coordinates

Aggregation: Geohash

Field: geoip.location

☑ Change precision on map zoom

Custom Label: Source Locations

Unique count of msg
- ● 1 – 12.4
- ● 12.4 – 23.8
- ● 23.8 – 35.2
- ● 35.2 – 46.6
- ● 46.6 – 58

Figure 23 Source Locations - Unique Attacks

Depending on the unique signatures, source location of those signatures were populated on a tile map aggregating location values as shown in Figure 23.

Metrics:

Value

Aggregation: Unique Count

Field: msg

Buckets:

Geo Coordinates

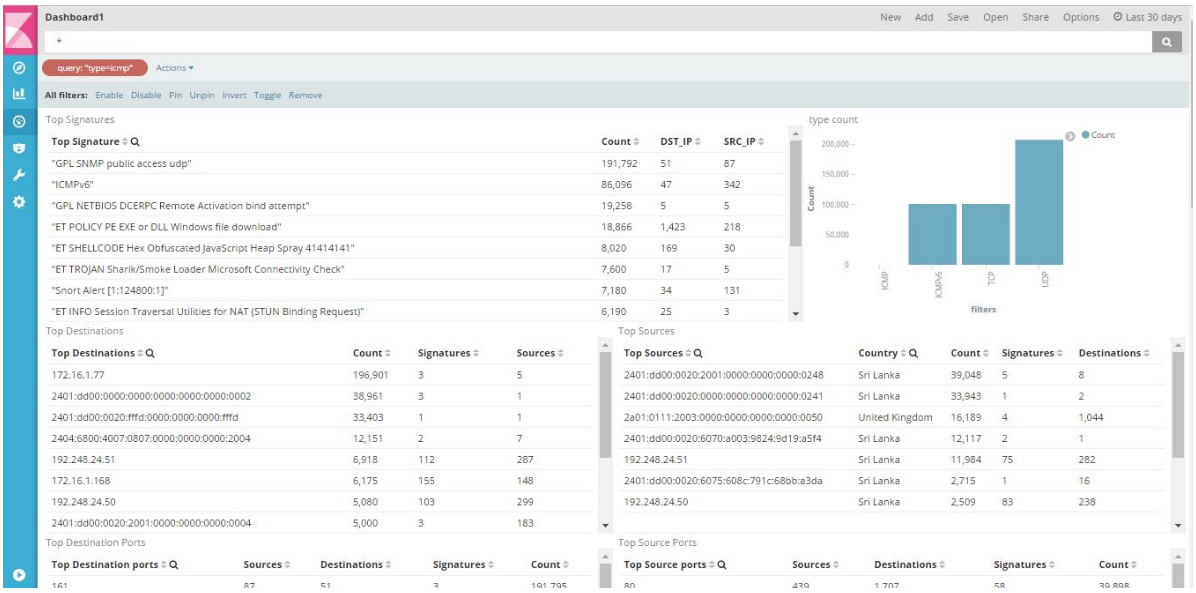
Aggregation: Geohash

Field: geoip.location

Change precision on map zoom: ticked

Custom Label: Source Locations

Next is to create a Dashboard in Kibana. This will be the front end to a user. Using the above created graphs a dashboard was created as shown in Figure 24.



Top Signatures

Top Signature	Count	DST_IP	SRC_IP
"GPL SNMP public access udp"	191,792	51	87
"ICMPv6"	86,096	47	342
"GPL NETBIOS DCERPC Remote Activation bind attempt"	19,258	5	5
"ET POLICY PE EXE or DLL Windows file download"	18,866	1,423	218
"ET SHELLCODE Hex Obfuscated JavaScript Heap Spray 41414141"	8,020	169	30
"ET TROJAN Sharik/Smoke Loader Microsoft Connectivity Check"	7,600	17	5
"Snort Alert [1:124800:1]"	7,180	34	131
"ET INFO Session Traversal Utilities for NAT (STUN Binding Request)"	6,190	25	3

Top Destinations

Top Destinations	Count	Signatures	Sources
172.16.1.77	196,901	3	5
2401:dd00:0000:0000:0000:0000:0000:0002	38,961	3	1
2401:dd00:0020:fffd:0000:0000:0000:fffd	33,403	1	1
2404:6800:4007:0807:0000:0000:0000:2004	12,151	2	7
192.248.24.51	6,918	112	287
172.16.1.168	6,175	155	148
192.248.24.50	5,080	103	299
2401:dd00:0020:2001:0000:0000:0000:0004	5,000	3	183

Top Sources

Top Sources	Country	Count	Signatures	Destinations
2401:dd00:0020:2001:0000:0000:0000:0248	Sri Lanka	39,048	5	8
2401:dd00:0020:0000:0000:0000:0000:0241	Sri Lanka	33,943	1	2
2a01:0111:2003:0000:0000:0000:0000:0050	United Kingdom	16,189	4	1,044
2401:dd00:0020:6070:a003:9824:9d19:a5f4	Sri Lanka	12,117	2	1
192.248.24.51	Sri Lanka	11,984	75	282
2401:dd00:0020:6075:608c:791c:68bb:a3da	Sri Lanka	2,715	1	16
192.248.24.50	Sri Lanka	2,509	83	238

Top Destination Ports

Top Destination ports	Sources	Destinations	Signatures	Count
161	87	51	3	191,795

Top Source Ports

Top Source ports	Sources	Destinations	Signatures	Count
80	439	1,707	58	39,898

Figure 24 Kibana custom dashboard

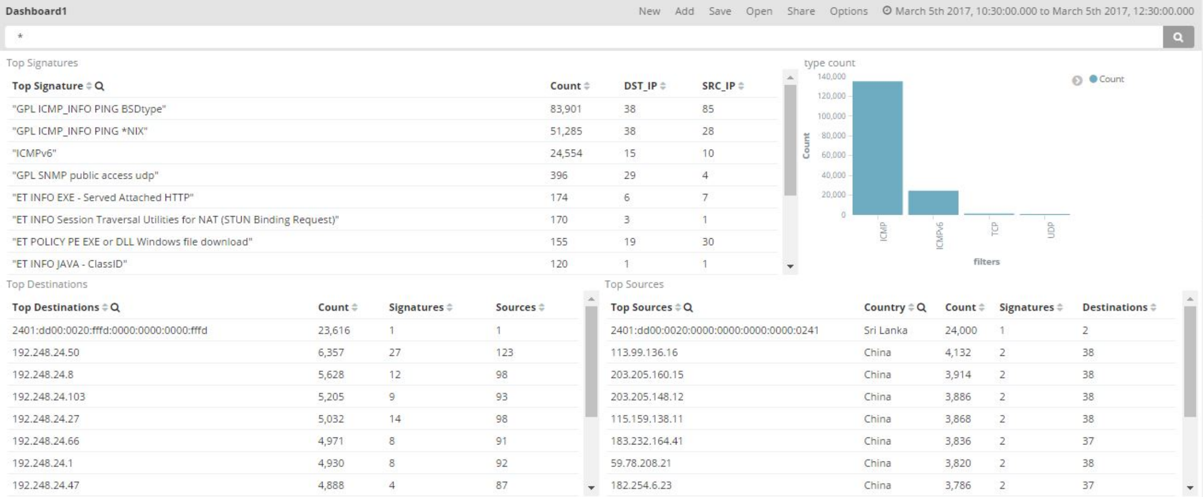
5.3 Evaluation

Modified version of Security Onion was installed in University of Kelaniya and it was connected to the network through a span connection which placed the server between the router and commercial firewall. Server was facing a 800Mbps Internet link, therefore had to increase the memory to 16GB for smooth actions.

Compared ELK results with Commercial Firewall and traditional squert. Samples were taken on 2017 March 05th from 10.30am – 12.30pm. In Security Onion/Squert time is logged in UTC and therefore the equivalent is 5.00 – 7.00 am.

Chapter 6 - Evaluation & Results

On the real environment following are the results shown in Kibana dashboard. Output is divided into 4 parts for the ease of description. Also note that the intention of the results are to indicate the IPv6 capability of the new system and not to describe the specific alerts itself.



Top Signatures

Top Signature	Count	DST_IP	SRC_IP
"GPL ICMP_INFO PING BSDtype"	83,901	38	85
"GPL ICMP_INFO PING *NIX"	51,285	38	28
"ICMPv6"	24,554	15	10
"GPL SNMP public access udp"	396	29	4
"ET INFO EXE - Served Attached HTTP"	174	6	7
"ET INFO Session Traversal Utilities for NAT (STUN Binding Request)"	170	3	1
"ET POLICY PE EXE or DLL Windows file download"	155	19	30
"ET INFO JAVA - ClassID"	120	1	1

Top Destinations

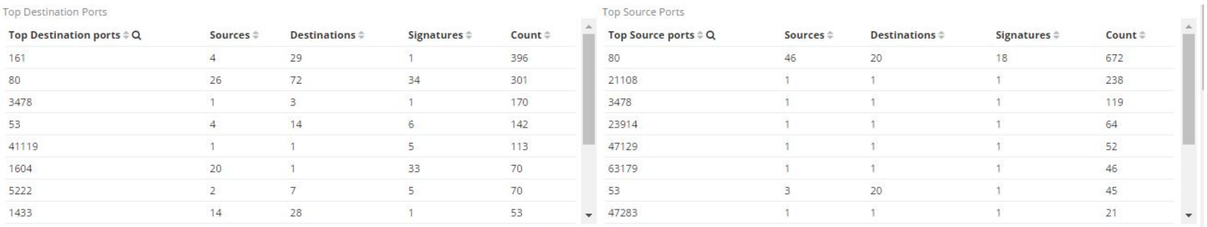
Top Destinations	Count	Signatures	Sources
2401:dd00:0020:fffd:0000:0000:0000:fffd	23,616	1	1
192.248.24.50	6,357	27	123
192.248.24.8	5,628	12	98
192.248.24.103	5,205	9	93
192.248.24.27	5,032	14	98
192.248.24.66	4,971	8	91
192.248.24.1	4,930	8	92
192.248.24.47	4,888	4	87

Top Sources

Top Sources	Country	Count	Signatures	Destinations
2401:dd00:0020:0000:0000:0000:0000:0241	Sri Lanka	24,000	1	2
113.99.136.16	China	4,132	2	38
203.205.160.15	China	3,914	2	38
203.205.148.12	China	3,886	2	38
115.159.138.11	China	3,868	2	38
183.232.164.41	China	3,836	2	37
59.78.208.21	China	3,820	2	38
182.254.6.23	China	3,786	2	37

Figure 25 Results - Kibana Dashboard1

As of the first dashboard result shown in Figure 25, it populates the Top Signatures, Top Destinations, Top Sources and Type Count. IPv6 capability can be seen on all four graphs. From the top sources it can be seen that most of the attacks were originated from China and they are IPv4.



Top Destination Ports

Top Destination ports	Sources	Destinations	Signatures	Count
161	4	29	1	396
80	26	72	34	301
3478	1	3	1	170
53	4	14	6	142
41119	1	1	5	113
1604	20	1	33	70
5222	2	7	5	70
1433	14	28	1	53

Top Source Ports

Top Source ports	Sources	Destinations	Signatures	Count
80	46	20	18	672
21108	1	1	1	238
3478	1	1	1	119
23914	1	1	1	64
47129	1	1	1	52
63179	1	1	1	46
53	3	20	1	45
47283	1	1	1	21

Figure 26 Results - Kibana Dashboard2

Second or the next output in Figure 26 shows the Top Destination Ports and Source Ports which are aggregations of TCP and UDP data in both dual stacks.

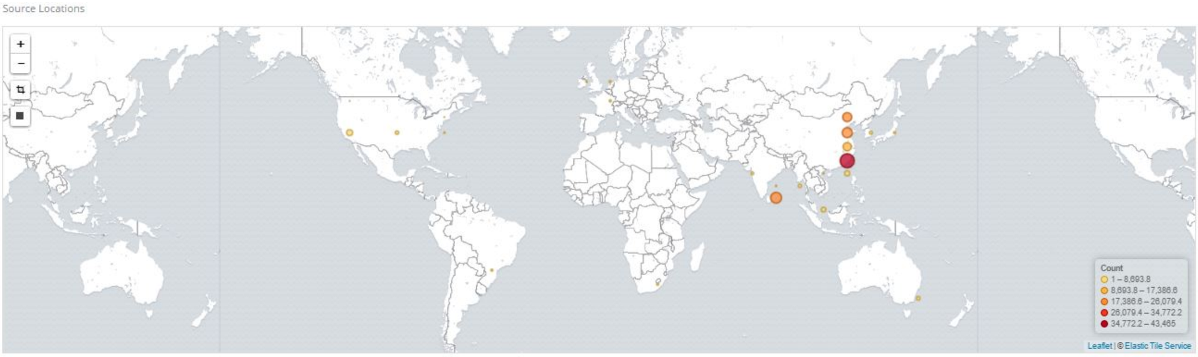


Figure 27 Results - Kibana Dashboard3

In the third part or the Figure 27, shows the aggregated count of activities based on the country of origin. Here it is visible that most of the alerts were happened due to traffic from China as seen in first result.

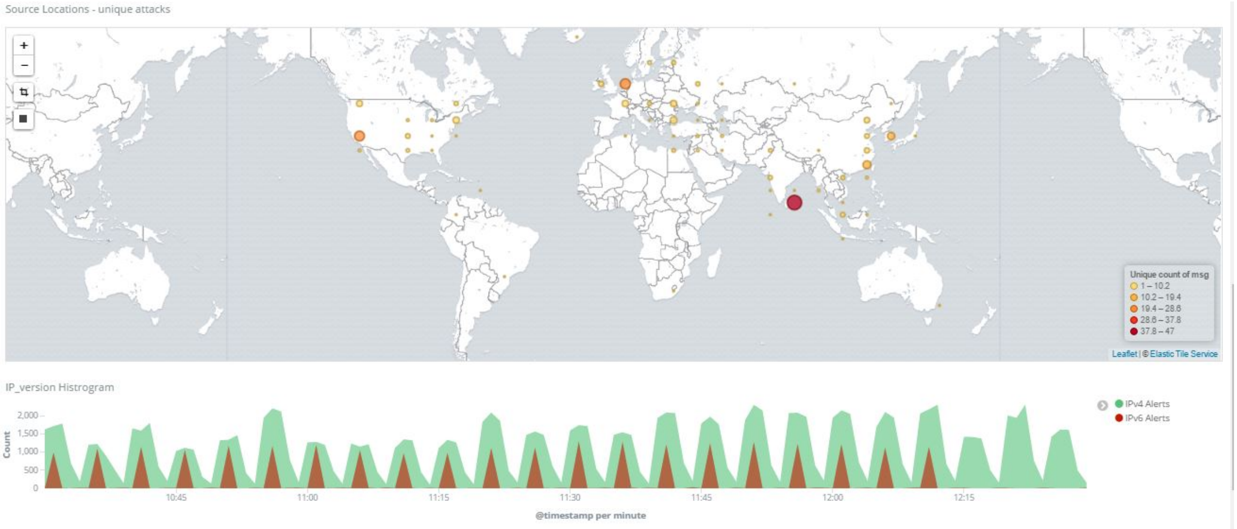
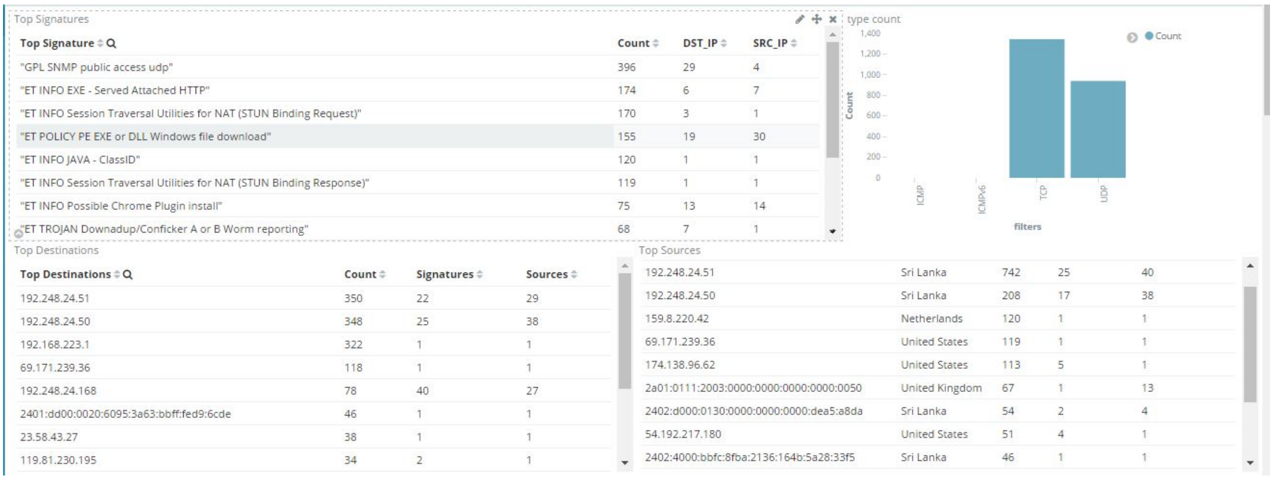


Figure 28 Results - Kibana Dashboard4

This result in Figure 28 indicates the sources that unique attacks were originated and from the results it can be seen that many countries not seen in Figure 27 can be seen. The Histogram of the IPv4 and IPv6 alert distribution is also listed below the map to show the distribution of alerts.

To illustrate IPv6 detection capability, another result was taken by filtering only the TCP and UDP data.



Top Signatures

Top Signature	Count	DST_IP	SRC_IP
"GPL SNMP public access udp"	396	29	4
"ET INFO EXE - Served Attached HTTP"	174	6	7
"ET INFO Session Traversal Utilities for NAT (STUN Binding Request)"	170	3	1
"ET POLICY PE EXE or DLL Windows file download"	155	19	30
"ET INFO JAVA - ClassID"	120	1	1
"ET INFO Session Traversal Utilities for NAT (STUN Binding Response)"	119	1	1
"ET INFO Possible Chrome Plugin install"	75	13	14
"ET TROJAN Downadup/Conficker A or B Worm reporting"	68	7	1

Top Destinations

Top Destinations	Count	Signatures	Sources
192.248.24.51	350	22	29
192.248.24.50	348	25	38
192.168.223.1	322	1	1
69.171.239.36	118	1	1
192.248.24.168	78	40	27
2401:dd00:0020:6095:3a63:bbff:fed9:6cde	46	1	1
23.58.43.27	38	1	1
119.81.230.195	34	2	1

Top Sources

192.248.24.51	Sri Lanka	742	25	40
192.248.24.50	Sri Lanka	208	17	38
159.8.220.42	Netherlands	120	1	1
69.171.239.36	United States	119	1	1
174.138.96.62	United States	113	5	1
2a01:0111:2003:0000:0000:0000:0000:0050	United Kingdom	67	1	13
2402:d000:0130:0000:0000:0000:dea5:a8da	Sri Lanka	54	2	4
54.192.217.180	United States	51	4	1
2402:4000:bbfc:8fba:2136:164b:5a28:33f5	Sri Lanka	46	1	1

Figure 29 Results - Kibana Dashboard5

This output Figure 29 shows multiple IPv6 Source addresses that confirms the capability of the system to detect IPv6. Figure 30 also shows the division of versions.

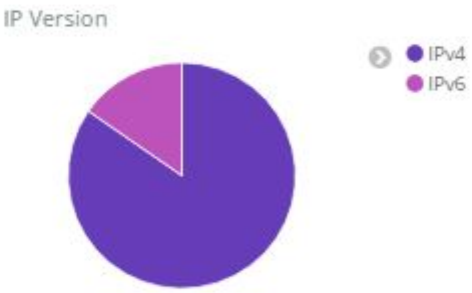
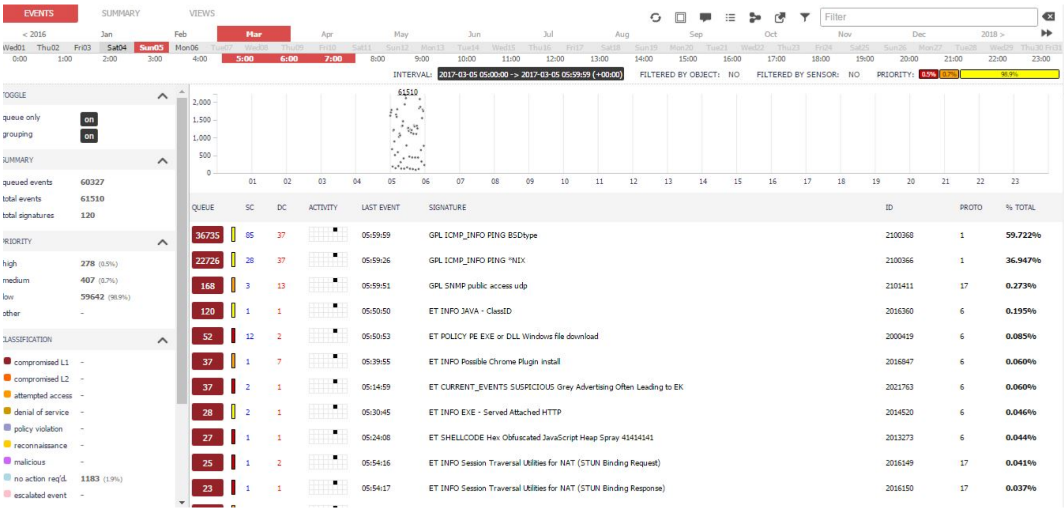


Figure 30: Results - Kibana Dashboard 6

As for the comparison squert output was taken,



INTERVAL: 2017-03-05 05:00:00 -> 2017-03-05 05:59:59 (+00:00) FILTERED BY OBJECT: NO FILTERED BY SENSOR: NO PRIORITY: 0.5% 0.7% 98.9%

TOGGLE

queue only	on
grouping	on

SUMMARY

queued events	60327
total events	61510
total signatures	120

PRIORITY

high	278 (0.5%)
medium	407 (0.7%)
low	59642 (98.9%)
other	-

CLASSIFICATION

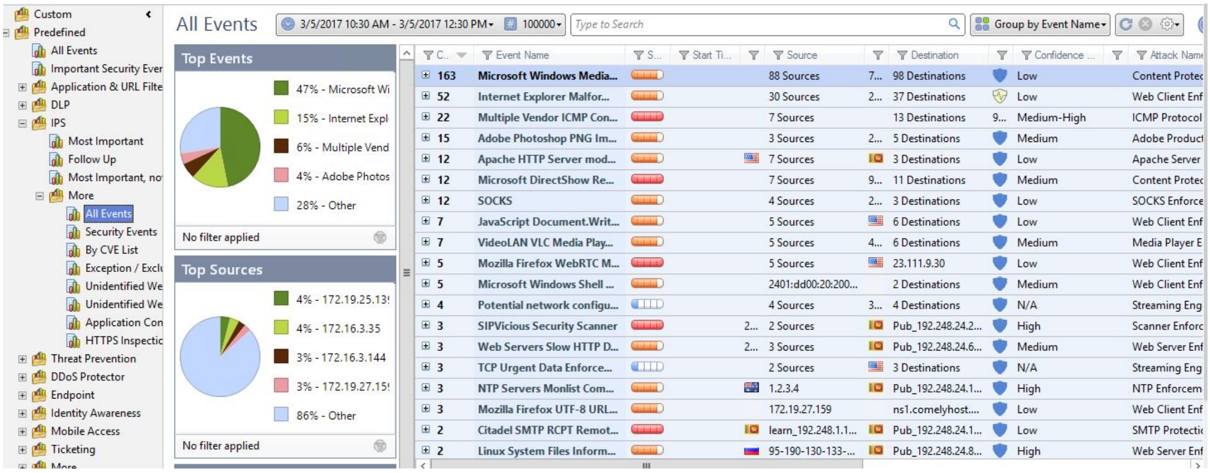
compromised L1	-
compromised L2	-
attempted access	-
denial of service	-
policy violation	-
reconnaissance	-
malicious	-
no action req'd.	1183 (1.9%)
escalated event	-

QUEUE	SC	DC	ACTIVITY	LAST EVENT	SIGNATURE	ID	PROTO	% TOTAL
36735	85	37		05:59:59	GPL ICMP_INFO PING BSDtype	2100368	1	59.722%
22726	28	37		05:59:26	GPL ICMP_INFO PING *NIX	2100366	1	36.947%
168	3	13		05:59:51	GPL SNMP public access udp	2101411	17	0.273%
120	1	1		05:50:50	ET INFO JAVA - ClassID	2016360	6	0.195%
52	12	2		05:50:53	ET POLICY PE EXE or DLL Windows file download	2000419	6	0.085%
37	1	7		05:39:55	ET INFO Possible Chrome Plugin install	2016847	6	0.060%
37	2	1		05:14:59	ET CURRENT_EVENTS SUSPICIOUS Grey Advertising Often Leading to EK	2021763	6	0.060%
28	2	1		05:30:45	ET INFO EXE - Served Attached HTTP	2014520	6	0.046%
27	1	1		05:24:08	ET SHELLCODE Hex Obfuscated JavaScript Heap Spray 41414141	2013273	6	0.044%
25	1	2		05:54:16	ET INFO Session Traversal Utilities for NAT (STUN Binding Request)	2016149	17	0.041%
23	1	1		05:54:17	ET INFO Session Traversal Utilities for NAT (STUN Binding Response)	2016150	17	0.037%

Figure 31 Squert Output

Figure 31 shows all the alerts except for the IPv6 related ones.

As for the comparison with the commercial firewall, following results were extracted from Checkpoint Gaia 77.30 [40] Smart Event software which is one of the event reporting tools in Checkpoint UTM devices. But the results are not equal as the two systems are using different rule bases. Most of the ICMP and ICMPv6 detected by snort have been ignored by the checkpoint.

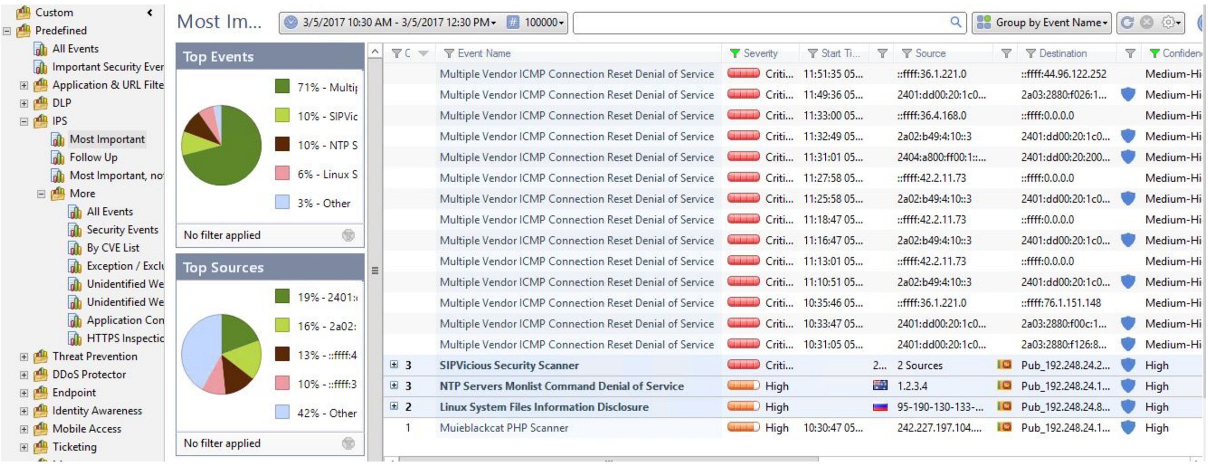


3/5/2017 10:30 AM - 3/5/2017 12:30 PM 100000 Type to Search Group by Event Name

C..	Event Name	S...	Start Ti...		Source		Destination		Confidence ...		Attack Name
163	Microsoft Windows Media...				88 Sources	7...	98 Destinations		Low		Content Protec
52	Internet Explorer Malfor...				30 Sources	2...	37 Destinations		Low		Web Client Enf
22	Multiple Vendor ICMP Con...				7 Sources		13 Destinations	9...	Medium-High		ICMP Protocol
15	Adobe Photoshop PNG Im...				3 Sources	2...	5 Destinations		Medium		Adobe Product
12	Apache HTTP Server mod...				7 Sources		3 Destinations		Low		Apache Server
12	Microsoft DirectShow Re...				7 Sources	9...	11 Destinations		Medium		Content Protec
12	SOCKS				4 Sources	2...	3 Destinations		Low		SOCKS Enforce
7	JavaScript Document.Writ...				5 Sources		6 Destinations		Low		Web Client Enf
7	VideoLAN VLC Media Play...				5 Sources	4...	6 Destinations		Medium		Media Player E
5	Mozilla Firefox WebRTC M...				5 Sources		23.111.9.30		Low		Web Client Enf
5	Microsoft Windows Shell ...				2401:dd00:20:200...		2 Destinations		Medium		Web Client Enf
4	Potential network configu...				4 Sources	3...	4 Destinations		N/A		Streaming Eng
3	SIPVicious Security Scanner			2...	2 Sources		Pub_192.248.24.2...		High		Scanner Enforc
3	Web Servers Slow HTTP D...			2...	3 Sources		Pub_192.248.24.6...		Medium		Web Server Enf
3	TCP Urgent Data Enforce...				2 Sources		3 Destinations		N/A		Streaming Eng
3	NTP Servers Monlist Com...				1.2.3.4		Pub_192.248.24.1...		High		NTP Enforcem
3	Mozilla Firefox UTF-8 URL...				172.19.27.159		ns1.comelyhost....		Low		Web Client Enf
2	Citadel SMTP RCPT Remot...				learn_192.248.1.1...		Pub_192.248.24.1...		Low		SMTP Protecti
2	Linux System Files Inform...				95-190-130-133-...		Pub_192.248.24.8...		High		Web Server Enf

Figure 32 Checkpoint IPS/IDS Activity 1

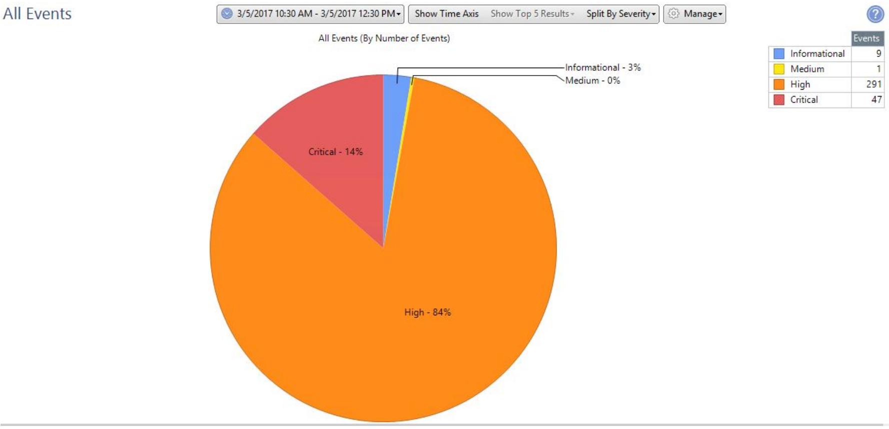
Above Figure 32 is the all IDS/IPS detected events grouped under same signature. Rather than security onion showing all internet traffic this also shows the internal traffic on LAN before getting NAT on the device itself. Here IPv6 alerts/events can also be seen.



C	Event Name	Severity	Start Ti...		Source	Destination	Confiden
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	11:51:35 05...		::ffff:36.1.221.0	::ffff:44.96.122.252	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	11:49:36 05...		2401:dd00:20:1c0...	2a03:2880:f026:1...	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	11:33:00 05...		::ffff:36.4.168.0	::ffff:0.0.0.0	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	11:32:49 05...		2a02:b49:4:10::3	2401:dd00:20:1c0...	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	11:31:01 05...		2404:a800:ff00:1::...	2401:dd00:20:200...	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	11:27:58 05...		::ffff:42.2.11.73	::ffff:0.0.0.0	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	11:25:58 05...		2a02:b49:4:10::3	2401:dd00:20:1c0...	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	11:18:47 05...		::ffff:42.2.11.73	::ffff:0.0.0.0	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	11:16:47 05...		2a02:b49:4:10::3	2401:dd00:20:1c0...	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	11:13:01 05...		::ffff:42.2.11.73	::ffff:0.0.0.0	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	11:10:51 05...		2a02:b49:4:10::3	2401:dd00:20:1c0...	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	10:35:46 05...		::ffff:36.1.221.0	::ffff:76.1.151.148	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	10:33:47 05...		2401:dd00:20:1c0...	2a03:2880:f00c:1...	Medium-Hi
	Multiple Vendor ICMP Connection Reset Denial of Service	Criti...	10:31:05 05...		2401:dd00:20:1c0...	2a03:2880:f126:8...	Medium-Hi
⊞ 3	**SIPVicious Security Scanner**	Criti...		2...	2 Sources	Pub_192.248.24.2...	High
⊞ 3	**NTP Servers Monlist Command Denial of Service**	High			1.2.3.4	Pub_192.248.24.1...	High
⊞ 2	**Linux System Files Information Disclosure**	High			95-190-130-133-...	Pub_192.248.24.8...	High
1	Muieblackcat PHP Scanner	High	10:30:47 05...		242.227.197.104....	Pub_192.248.24.1...	High

Figure 33 Checkpoint IPS/IDS Activity 2

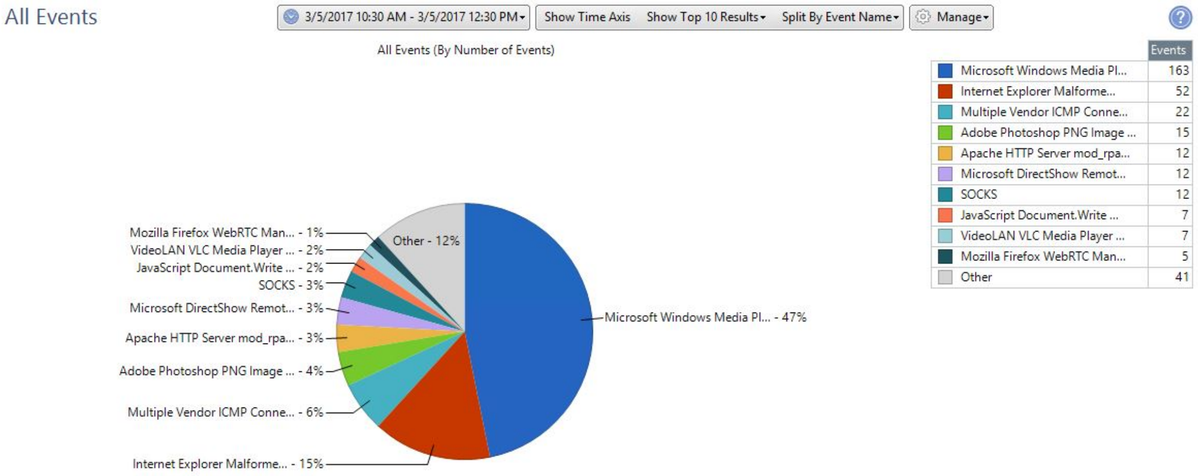
On the Figure 33, the second checkpoint image it shows the critical IPS events occurred and from this it can be seen that ICMPv6 based traffic at large. With comparison to snort, here it is visible that ICMP events had occurred.



	Events
🟦 Informational	9
🟨 Medium	1
🟧 High	291
🟥 Critical	47

Figure 34 Checkpoint IPS/IDS Activity 3

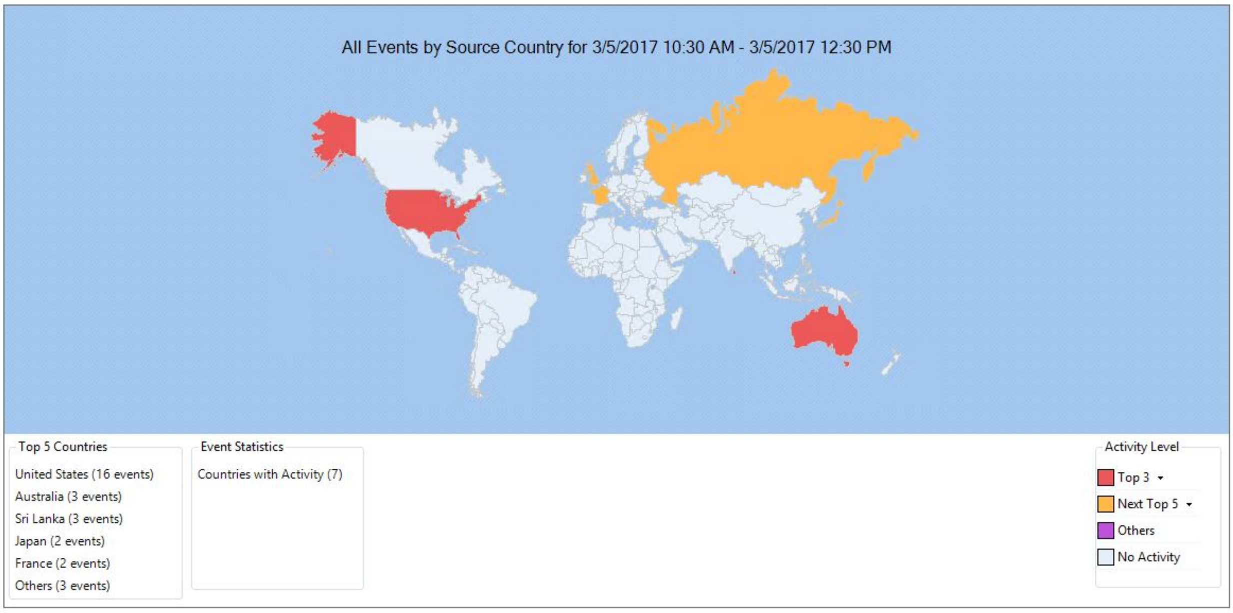
From the Figure 34 the checkpoints 3rd image it shows the breakdown of events depending on their severity level. A large number of High Severity alerts can be observed.



	Events
Microsoft Windows Media Pl...	163
Internet Explorer Malforme...	52
Multiple Vendor ICMP Conne...	22
Adobe Photoshop PNG Image ...	15
Apache HTTP Server mod_rpa...	12
Microsoft DirectShow Remot...	12
SOCKS	12
JavaScript Document.Write ...	7
VideoLAN VLC Media Player ...	7
Mozilla Firefox WebRTC Man...	5
Other	41

Figure 35 Checkpoint IPS/IDS Activity 4

Forth checkpoint image Figure 35, shows the event distribution divided according to the type of events. According to checkpoint highest attacks were Microsoft Windows Media Player PNG chunk code execution.


All Events by Source Country for 3/5/2017 10:30 AM - 3/5/2017 12:30 PM
Activity Level
Top 3
Next Top 5
Others
No Activity

Top 5 Countries

United States (16 events)
Australia (3 events)
Sri Lanka (3 events)
Japan (2 events)
France (2 events)
Others (3 events)

Event Statistics

Countries with Activity (7)

Figure 36 Checkpoint IPS/IDS Activity 5

Next, Figure 36 is the Source distribution according to country. As seen in ELK we cannot seen China as checkpoint is having a separate rule set in detecting threats. In this map, USA and Australia are the high source originating countries. Again China can be seen as a third highest originator.

As a summery to the comparison, it can be stated that commercial IDS uses a separate rule set other than the community build rules of snort. Therefore, the detection of some attacks may not visible in one IDS. The high number of ICMP detections were not detected by the commercial IDS due to the lack of rules on ICMP, this may eventually result in a massively distributed denial of service attack.

From Figure 25 to 30 it clearly shows that the new setup detects IPv6 and not like the previous original state, now it can alert in both IP versions.

Chapter 7 - Conclusion

Objective of the thesis, enabling open source intrusion detection / prevention system to detect IPv6 was accomplished by installing ELK stack on Security Onion. With the involvement of snort detection engine and barnyard the open source spooler which interpreted snort’s unified binary data into various other formats have effectively detected and outputted the IPv6 traffic and with the newly integrated ELK stack, alerts can be quickly visualize to the user. This was also possible because of the snort rules made to detect ICMPv6 threats. The default rules are basically still targeted on IPv4 but there are few IPv6 related rules.

Custom made GROK patterns are very handy in use as they efficiently filtered fields in input data. This made the tabularizing, sorting and analyzing easy as all data were like in a separate database system. Kibana visualizing were very effective when filtering and aggregating data into charts and because of that a clear view made on alerting IPv6 threats and also in dual stack environment it can be easy used to check both IP versions because of the graphing power of Kibana.

With comparison to commercial systems, the mentioned system may again detect more threats as the rule set used here developed by a huge community base and as they are rapidly updated. But efficiency may depend on how the rules are enabled on Snort engine. Therefore, the effectiveness of snort engine may greatly vary on how the administrators enable or disable signature rules.

As the final note it can be concluded that with the upgrade of Barnyard2 and with the integration of Elasticsearch, Logstash and Kibana into Security Onion distribution can be used as a IDS/IPS tool for IPv6 traffic. This states that the open source IPS/IDS solution which was selected is capable in detecting IPv6 and reporting once the mentioned changes are done. Also the effectiveness of the rules can be vary depending on the environment it is used and as these are all open source products users are allowed to do their modifications.

7.1 Future Work

Next step of the above work is to distribute the developed system with named as “SO-ELK” to the Security onion community and check how the community responds. As sguil is not going to be developed again as for the developer, the SO-ELK will be a good alternative in detecting IPv6 related alerts.

If the Security Onion community accepts the development of SO-ELK, it can be commit to the official distribution on GitHub.

Also, the developed GROK patterns are to be committed to Logstash tree on GitHub as it is custom made to detect snort csv format.

Then creation of new snort rules for IPv6 specific attacks have to be done as we can expect more IPv6 traffic flowing as most of the main Content Delivery Networks, Google, Facebook have upgraded their core networks to run on native IPv6 at the end of 2016.

More graphical charts have to create to visualize more traffic patterns in Kibana. This can be done with collaboration of other parties who use more IPv6 traffic.

Creation of new TCP and UDP based rules which are more specific for IPv6 traffic have to be created and verified with the community in detecting various anomalies.

As the last part, this system can be implemented on Universities with Sri Lankan context as most of the universities are still not using commercial firewall or UTM devices. This will be an excellent low cost solution as all the universities are facing the attacks from around the world.

Chapter 8 - Bibliography

[1] L. Smith and I. Lipner, "Free Pool of IPv4 Address Space Depleted," Number Resource Organization, 11 February 2011. [Online]. Available: https://www.nro.net/ipv4-free-pool-depleted/. [Accessed 25 April 2016].

[2] Internet Assigned Numbers Authority, "IANA IPv4 Address Space Registry," 12 January 2016. [Online]. Available: https://www.iana.org/assignments/ipv4-address-space/ipv4-address-space.txt. [Accessed 25 April 2016].

[3] S. Deering and R. Hinden, "“Internet Protocol, Version 6 (IPv6) Specification”," December 1998. [Online]. Available: http://www.ietf.org/rfc/rfc2460.txt.

[4] ICANN, "ICANN Research," [Online]. Available: http://stats.research.icann.org/rir/#v6_alloc_rir. [Accessed 20 February 2017].

[5] Cisco Inc., "IPv6 IOS Firewall," 22 03 2012. [Online]. Available: http://www.cisco.com/en/US/docs/ios-xml/ios/sec_data_cbac_fw/configuration/15-2mt/ip6-firewall.html. [Accessed 3 8 2016].

[6] checkpoint, "IPv6 Support FAQ," 19 4 2009. [Online]. Available: https://supportcenter.checkpoint.com/supportcenter/portal?eventSubmit_doGoviewsolutiondetails=&solutionid=sk39374. [Accessed 3 8 2016].

[7] paloalto, "Palo Alto Networks — Delivering Network Security for IPv6 Networks," 04 03 2015. [Online]. Available: https://www.paloaltonetworks.com/resources/whitepapers/network-security-ipv6-networks. [Accessed 3 8 2016].

[8] Check Point Software Technologies Ltd, "Check Point R77 Known Limitations," 02 November 2016. [Online]. Available: https://supportcenter.checkpoint.com/supportcenter/portal?eventSubmit_doGoviewsolutiondetails=&solutionid=sk92967. [Accessed 15 December 2016].

[9] Security Onion Solutions, "Security Onion," 8 August 2016. [Online]. Available: https://securityonion.net/.

[10] Cisco Inc., "Snort," 2016. [Online]. Available: https://www.snort.org/. [Accessed 31 8 2016].

[11] A. Conta, S. Deering and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification," March 2006. [Online]. Available: https://tools.ietf.org/html/rfc4443. [Accessed 12 August 2016].

[12] IPv6Test, "IPv6 in Sri Lanka," 20 8 2016. [Online]. Available: http://ipv6-test.com/stats/country/LK.

[13] S. Convery and D. Miller, "IPv6 and IPv4 Threat Comparison and Best-Practice Evaluation," March 2004. [Online]. Available: http://www.seanconvery.com/v6-v4-threats.pdf.

[14] F. Gont and W. Liu, "Security Implications of IPv6 on IPv4 Networks," February 2014. [Online]. Available: https://www.ietf.org/rfc/rfc7123.txt.

[15] H. A. Dawood, "IPv6 Security Vulnerabilities," *International Journal of Information Security Science,* vol. 1, no. 4, pp. 100-105, December 2012.

[16] S. Sotillo, "IPv6 Security Issues," East Carolina University, East Carolina, 2006.

[17] C. E. Caicedo, J. B. Joshi and S. R. Tuladhar, "IPv6 Security Challenges," *Computer,* vol. 42, no. 2, pp. 36-42, 2009.

[18] E. Durdağı and A. Buldu, "IPV4/IPV6 security and threat comparisons," *Procedia - Social and Behavioral Sciences,* vol. 2, no. 2, pp. 5285-5291, 2010.

[19] D. Zagar and K. Grgic, "IPv6 security threats and possible solutions," in *World Automation Congress, 2006 IEEE*, Budapest, Hungary, 2006.

[20] J. M. Chasser, "Security Concerns in IPv6 and Transition Networks," *Information Security Journal: A Global Perspective,* no. 19, pp. 282-293, 2010.

[21] E. Davies and J. Mohacsi, "Recommendations for Filtering ICMPv6 Messages in Firewalls. Network Working Group," 2007. [Online]. Available: http://tools.ietf.org/html/rfc4890.

[22] J. M. Allen, "IPv6 IDS," 2015. [Online]. Available: https://www.sans.org/reading-room/whitepapers/detection/ipv6-open-source-ids-35957.

[23] B. Visscher, "Sguil: The Analyst Console for Network Security Monitoring," 2014. [Online]. Available: http://bammv.github.io/sguil/index.html. [Accessed 10 May 2016].

[24] Cisco Inc., "Snort User Manual," Cisco Inc., [Online]. Available: http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node27.html. [Accessed 29 Augest 2016].

[25] I. Firns, "firnsy/barnyard2," 2013. [Online]. Available: firnsy/barnyard2. [Accessed 16 July 2016].

[26] ElasticSearch BV, "Elastic | Home Page," [Online]. Available: https://www.elastic.co/. [Accessed 15 September 2016].

[27] ElasticSearch BV, "Elasticsearch," [Online]. Available: https://www.elastic.co/products/elasticsearch. [Accessed 12 November 2016].

[28] ElasticSearch BV, "Logstash," [Online]. Available: https://www.elastic.co/products/logstash. [Accessed 15 November 2016].

[29] ElasticSearch BV, "Filebeat," [Online]. Available: https://www.elastic.co/products/beats/filebeat. [Accessed 15 November 2016].

[30] ElasticSearch, "Kibana," [Online]. Available: https://www.elastic.co/products/kibana. [Accessed 15 November 2016].

[31] K. Branch, "Security Onion Mailling Group," 02 May 2014. [Online]. Available: https://groups.google.com/forum/#!topic/security-onion/1rYOnxZ2Irs/discussion. [Accessed 14 December 2016].

[32] ElasticSearch BV, "Install Elasticsearch with Debian Package," [Online]. Available: https://www.elastic.co/guide/en/elasticsearch/reference/5.2/deb.html. [Accessed 20 December 2016].

[33] Elasticsearch BV, "Installing Logstash," [Online]. Available: https://www.elastic.co/guide/en/logstash/5.2/installing-logstash.html. [Accessed 20 December 2016].

[34] M. Anicas, "How To Install Elasticsearch, Logstash, and Kibana (ELK Stack) on Ubuntu 14.04," 10 March 2015. [Online]. Available: https://www.digitalocean.com/community/tutorials/how-to-install-elasticsearch-logstash-and-kibana-elk-stack-on-ubuntu-14-04. [Accessed 23 December 2016].

[35] Elasticsearch BV, "grok," [Online]. Available: https://www.elastic.co/guide/en/logstash/current/plugins-filters-grok.html. [Accessed 23 December 2016].

[36] colinsurprenant, "elastic/logstash/patterns/grok-patterns," [Online]. Available: https://github.com/elastic/logstash/blob/v1.4.2/patterns/grok-patterns. [Accessed 23 December 2016].

[37] Elasticsearch BV, "Geoip," [Online]. Available: https://www.elastic.co/guide/en/logstash/current/plugins-filters-geoip.html. [Accessed 26 December 2016].

[38] Elasticsearch BV, "Installing Filebeat," [Online]. Available: https://www.elastic.co/guide/en/beats/filebeat/5.2/filebeat-installation.html. [Accessed 26 December 2016].

[39] M. Crawford, "Internet Control Message Protocol version 6 (ICMPv6) Parameters," 27 January 2017. [Online]. Available: http://www.iana.org/assignments/icmpv6-parameters/icmpv6-parameters.xhtml. [Accessed 10 February 2017].

[40] Checkpoint Software Technologies Ltd, "Check Point R77.30," 19 May 2015. [Online]. Available: https://supportcenter.checkpoint.com/supportcenter/portal?eventSubmit_doGoviewsolutiondetails=&solutionid=sk104859. [Accessed 12 January 2017].

[41] Open Information Security Fundation, "index," 2016. [Online]. Available: https://suricata-ids.org. [Accessed 31 8 2016].

[42] J. Postel, "“Internet Protocol, DARPA Internet Program Protocol Specification”," September 1981. [Online]. Available: http://www.ietf.org/rfc/rfc0791.txt.

Annexure A – Email From Security Onion Community

To understand the background of Security Onion support on IPv6 mails were used to communicate with community. As a result main contributors replied that most of the tools were not supporting IPv6 except for snort barnyard bro etc.

Email Received from Security-Onion Developing Community:

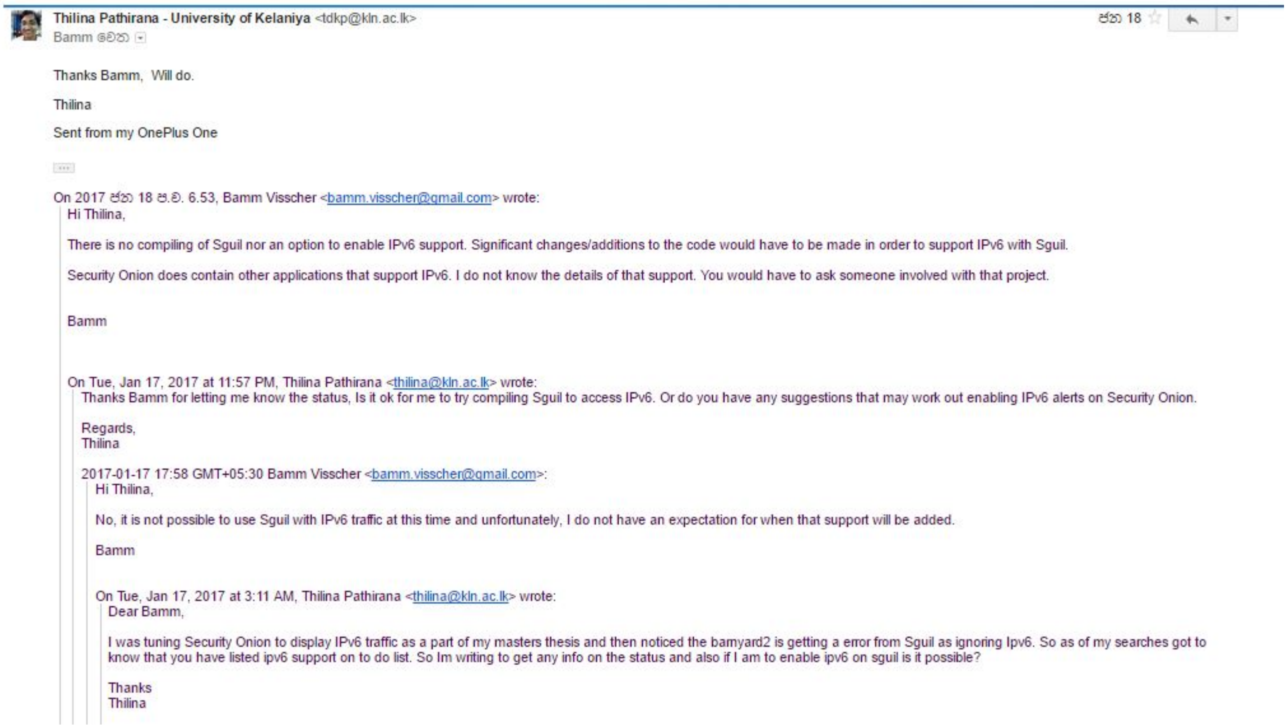
```
MIME-Version: 1.0
Received: by 10.194.88.38 with HTTP; Mon, 23 May 2016 08:13:39 -0700 (PDT)
In-Reply-To:
<CAHjBB6HvwfHPf2vD+6=BYaLo4c7NYoW3Ox7RVyL7V4_d+GHozg@mail.gmail.com
>
References:
<CACF2C_wyftRHgzkmKmWOb208vxqBZN0a9XgPposK=nE5KaCK5Q@mail.gmail.com
>
<CAHjBB6HvwfHPf2vD+6=BYaLo4c7NYoW3Ox7RVyL7V4_d+GHozg@mail.gmail.com
>
Date: Mon, 23 May 2016 20:43:39 +0530
Delivered-To: tdkp123@gmail.com
Message-ID: <CACF2C_xGuaDHy_X2v99ihm-Eb8k0A8UFZ3-
nfuRO=VcjsSLRGg@mail.gmail.com>
Subject: Re: [security-onion] SO IPV6 Alerts
From: Thilina Pathirana <tdkp123@gmail.com>
To: security-onion@googlegroups.com
Content-Type: multipart/alternative; boundary=001a1130cf30a7e076053383e2ff

--001a1130cf30a7e076053383e2ff
Content-Type: text/plain; charset=UTF-8
Content-Transfer-Encoding: quoted-printable

Thanks Wes, then I will try my work with snort alert logs.

rgds,
Thilina

On Mon, May 23, 2016 at 8:41 PM, Wes Lambert <wlambertts@gmail.com> wrote:

> As far as I understand, the main sniffing processes (Snort, Suricata, Bro=
)
> support interpretation of IPV6 traffic, however, I don't believe the
> database schema(s) for the alert interfaces (Squert, Sguil, ELSA) has/hav=
e
> been updated to support IPV6--this means such alerts would not be viewabl=
e
> in the interface(s).
>
> Thanks,
> Wes
> On May 23, 2016 9:39 AM, "Thilina Pathirana" <tdkp123@gmail.com> wrote:
>
>> Dear all,
```

>> I was looking to get alerts based on IPv6 traffic on squert or elsa but
>> did not see any. After some search on previous threads I assumed that
>> squert or elsa is still not capable on showing IPv6 alerts.
>>
>> Am I correct?
>>
>> I can read IPv6 data on packets captured and I am in a process of findin=
g
>> ways to migrate snort IPv4 rules to IPv6 compatible mode as a masters
>> research. So is there any help that I can get on setting up squert or el=
sa
>> to show IPv6 alerts.
>>
>> thanks
>>
>> Best Regards,
>> Thilina Pathirana
>> --
>>
>> --
>> *Thilina Pathirana | Assistant Network Manager*
>> *University of Kelaniya** - Sri Lanka*.
>> T. +94112903424 | F. +94112910163 | M. +94716246331
>> *Program Committee Chairman | EXCO ISOC-LK*
>> *Vice Leader CDU-SLSM*
>> *www.google.com/+ThilinaPathirana
>> <http://www.google.com/+ThilinaPathirana>*
>>
>> *www.facebook.com/t.d.k.pathirana
>> <http://www.facebook.com/t.d.k.pathirana>*
>>
>> *lk.linkedin.com/in/thilinapathirana/
>> <http://lk.linkedin.com/in/thilinapathirana/>*
>> *skype: tdkp123*
>>
>> - =E2=80=93 - =E2=80=93 - =E2=80=93 =E2=80=93 - =E2=80=93 =E2=80=93 - =
=E2=80=93 =E2=80=93 - =E2=80=93 =E2=80=93 - =E2=80=93 =E2=80=93 - =E2=80=93=
 =E2=80=93 - =E2=80=93 =E2=80=93 - =E2=80=93 =E2=80=93 - =E2=80=93 =E2=80=
=93 - =E2=80=93 =E2=80=93 - --
>>
>> =E2=80=9CSave a Tree=E2=80=9D =E2=80=93 Please consider the environment =
before printing this
>> email.
>>
>>
>> --
>> Follow Security Onion on Twitter!
>> https://twitter.com/securityonion
>> ---
>> You received this message because you are subscribed to the Google Group=
s
>> "security-onion" group.

```
>> To unsubscribe from this group and stop receiving emails from it, send a=
n
>> email to security-onion+unsubscribe@googlegroups.com.
>> To post to this group, send email to security-onion@googlegroups.com.
>> Visit this group at https://groups.google.com/group/security-onion.
>> For more options, visit https://groups.google.com/d/optout.
>>
> --
> Follow Security Onion on Twitter!
> https://twitter.com/securityonion
> ---
> You received this message because you are subscribed to the Google Groups
> "security-onion" group.
> To unsubscribe from this group and stop receiving emails from it, send an
> email to security-onion+unsubscribe@googlegroups.com.
> To post to this group, send email to security-onion@googlegroups.com.
> Visit this group at https://groups.google.com/group/security-onion.
> For more options, visit https://groups.google.com/d/optout.
>



--=20
Find me on:
www.google.com/+ThilinaPathirana
www.facebook.com/t.d.k.pathirana
lk.linkedin.com/in/thilinapathirana/
skype:  tdkp123

!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%%#tdkp#%%
$$$$$$$$$$$$
```

Annexure B – Email Communication with Sguil Developer

Email Communication with Sguil Developer Bamm Visscher (bamm.visscher@gmail.com) was done as figure 37, to understand the ability of enabling IPv6 on Sguil. According to Bamm upgrading sguil was not possible.



Thilina Pathirana - University of Kelaniya <tdkp@kln.ac.lk>
Bamm වෙත

ජන 18

Thanks Bamm, Will do.

Thilina

Sent from my OnePlus One

On 2017 ජන 18 ප.ව. 6.53, Bamm Visscher <bamm.visscher@gmail.com> wrote:

> Hi Thilina,
>
> There is no compiling of Sguil nor an option to enable IPv6 support. Significant changes/additions to the code would have to be made in order to support IPv6 with Sguil.
>
> Security Onion does contain other applications that support IPv6. I do not know the details of that support. You would have to ask someone involved with that project.
>
> Bamm
>
> On Tue, Jan 17, 2017 at 11:57 PM, Thilina Pathirana <thilina@kln.ac.lk> wrote:
>
>> Thanks Bamm for letting me know the status, Is it ok for me to try compiling Sguil to access IPv6. Or do you have any suggestions that may work out enabling IPv6 alerts on Security Onion.
>>
>> Regards,
>> Thilina
>>
>> 2017-01-17 17:58 GMT+05:30 Bamm Visscher <bamm.visscher@gmail.com>:
>>
>>> Hi Thilina,
>>>
>>> No, it is not possible to use Sguil with IPv6 traffic at this time and unfortunately, I do not have an expectation for when that support will be added.
>>>
>>> Bamm
>>>
>>> On Tue, Jan 17, 2017 at 3:11 AM, Thilina Pathirana <thilina@kln.ac.lk> wrote:
>>>
>>>> Dear Bamm,
>>>>
>>>> I was tuning Security Onion to display IPv6 traffic as a part of my masters thesis and then noticed the barnyard2 is getting a error from Sguil as ignoring Ipv6. So as of my searches got to know that you have listed ipv6 support on to do list. So Im writing to get any info on the status and also if I am to enable ipv6 on sguil is it possible?
>>>>
>>>> Thanks
>>>> Thilina

Figure 37 Email Communication with Sguil Developer

Thilina Pathirana - University of Kelaniya <tdkp@kln.ac.lk>

⍰⍰ 18

Bamm ⍰⍰⍰

Thanks Bamm, Will do.

Thilina

Sent from my OnePlus One

On 2017 ⍰⍰ 18 ⍰.⍰. 6.53, Bamm Visscher <bamm.visscher@gmail.com> wrote:

Hi Thilina,

There is no compiling of Sguil nor an option to enable IPv6 support. Significant changes/additions to the code would have to be made in order to support IPv6 with Sguil.

Security Onion does contain other applications that support IPv6. I do not know the details of that support. You would have to ask someone involved with that project.

Bamm

On Tue, Jan 17, 2017 at 11:57 PM, Thilina Pathirana <thilina@kln.ac.lk> wrote:

Thanks Bamm for letting me know the status, Is it ok for me to try compiling Sguil to access IPv6. Or do you have any suggestions that may work out enabling IPv6 alerts on Security Onion.

Regards,

Thilina

2017-01-17 17:58 GMT+05:30 Bamm Visscher <bamm.visscher@gmail.com>:

Hi Thilina,

No, it is not possible to use Sguil with IPv6 traffic at this time and unfortunately, I do not have an expectation for when that support will be added.

Bamm

On Tue, Jan 17, 2017 at 3:11 AM, Thilina Pathirana <thilina@kln.ac.lk> wrote:

Dear Bamm,

I was tuning Security Onion to display IPv6 traffic as a part of my masters thesis and then noticed the barnyard2 is getting a error from Sguil as ignoring Ipv6. So as of my searches got to know that you have listed ipv6 support on to do list. So Im writing to get any info on the status and also if I am to enable ipv6 on sguil is it possible?

Thanks

Thilina

--

Thilina Pathirana | Assistant Network Manager

University of Kelaniya - Sri Lanka.

T. +94112903424 | F. +94112910163 | M. +94716246331

Program Committee Chairman | EXCO ISOC-LK

Vice Leader CDU-SLSM

www.google.com/+ThilinaPathirana

www.facebook.com/t.d.k.pathirana

lk.linkedin.com/in/thilinapathirana/

skype: tdkp123

- – --

“Save a Tree” – Please consider the environment before printing this email.

+++

University of Kelaniya Sri Lanka, accepts no liability for the content of this email, or for the consequences of any actions taken on the basis of the information provided, unless that information is subsequently confirmed in writing. If you are not the intended recipient, this email and/or any information it contains should not be copied, disclosed, retained or used by you or any other party and the email and all its contents should be promptly deleted fully from our system and the sender informed.

E-mail transmission cannot be guaranteed to be secure or error-free as information could be intercepted, corrupted, lost, destroyed, arrive late or incomplete.

++

--

sguil - The Analyst Console for NSM

http://www.sguil.net

--

Thilina Pathirana | Assistant Network Manager

University of Kelaniya - Sri Lanka.

T. +94112903424 | F. +94112910163 | M. +94716246331

Program Committee Chairman | EXCO ISOC-LK

Vice Leader CDU-SLSM

www.google.com/+ThilinaPathirana

www.facebook.com/t.d.k.pathirana

lk.linkedin.com/in/thilinapathirana/

skype: tdkp123

- – --

“Save a Tree” – Please consider the environment before printing this email.

+++

University of Kelaniya Sri Lanka, accepts no liability for the content of this email, or for the consequences of any actions taken on the basis of the information provided, unless that information is subsequently confirmed in writing. If you are not the intended recipient, this email and/or any information it contains should not be copied, disclosed, retained or used by you or any other party and the email and all its contents should be promptly deleted fully from our system and the sender informed.

E-mail transmission cannot be guaranteed to be secure or error-free as information could be intercepted, corrupted, lost, destroyed, arrive late or incomplete.

++

--

sguil - The Analyst Console for NSM

http://www.sguil.net

Annexure C – Elasticsearch Configuration

Elasticsearch full configuration which is saved under /etc/elasticsearch/elasticsearch.yml. This configuration includes lot of commented lines which describes the default values that are in use. When fine tuning these values have to be changed.

```
# ==================== Elasticsearch Configuration ========================
#
# NOTE: Elasticsearch comes with reasonable defaults for most settings.
#       Before you set out to tweak and tune the configuration, make sure you
#       understand what are you trying to accomplish and the consequences.
#
# The primary way of configuring a node is via this file. This template lists
# the most important settings you may want to configure for a production cluster.
#
# Please consult the documentation for further information on configuration options:
# https://www.elastic.co/guide/en/elasticsearch/reference/index.html
#
# ---------------------------------- Cluster -----------------------------------
#
# Use a descriptive name for your cluster:
#
#cluster.name: my-application
#
# ------------------------------------ Node ------------------------------------
#
# Use a descriptive name for the node:
#
#node.name: node-1
#
# Add custom attributes to the node:
#
#node.attr.rack: r1
#
```

```
# ---------------------------------- Paths -----------------------------------
#
# Path to directory where to store the data (separate multiple locations by comma):
#
#path.data: /path/to/data
#
# Path to log files:
#
#path.logs: /path/to/logs
#
# ---------------------------------- Memory -----------------------------------
#
# Lock the memory on startup:
#
#bootstrap.memory_lock: true
#
# Make sure that the heap size is set to about half the memory available
# on the system and that the owner of the process is allowed to use this
# limit.
#
# Elasticsearch performs poorly when the system is swapping the memory.
#
# --------------------------------- Network ----------------------------------
#
# Set the bind address to a specific IP (IPv4 or IPv6):
#
#network.host: 192.168.0.1
 network.host: localhost
# Set a custom port for HTTP:
#
#http.port: 9200
#
# For more information, consult the network module documentation.
```

```
#
# -------------------------------- Discovery ---------------------------------
#
# Pass an initial list of hosts to perform discovery when new node is started:
# The default list of hosts is ["127.0.0.1", "[::1]"]
#
#discovery.zen.ping.unicast.hosts: ["host1", "host2"]
#
# Prevent the "split brain" by configuring the majority of nodes (total number of master-eligible nodes / 2 + 1):
#
#discovery.zen.minimum_master_nodes: 3
#
# For more information, consult the zen discovery module documentation.
#
# --------------------------------- Gateway ----------------------------------
#
# Block initial recovery after a full cluster restart until N nodes are started:
#
#gateway.recover_after_nodes: 3
#
# For more information, consult the gateway module documentation.
#
# --------------------------------- Various ----------------------------------
#
# Require explicit names when deleting indices:
#
#action.destructive_requires_name: true
```

Annexure D – Filebeat Configuration

Filebeat full configuration includes what are the types of input data and the location of input data. Also it specifies the output. Here the default output elasticsearch is commented and Logstash is enabled. This was done because the input data have to be filtered with Logstash before entering ElasticSearch.

```
###################### Filebeat Configuration #########################
#============== Filebeat prospectors ========================
filebeat.prospectors:
- input_type: log
  paths:
    - /var/log/barnyard2logs.csv
#==================== Outputs =============================
#-------------------------- Elasticsearch output ------------------------------
#output.elasticsearch:
# hosts: ["localhost:9200"]
  # Optional protocol and basic auth credentials.
  #protocol: "https"
  #username: "elastic"
  #password: "changeme"
#----------------------------- Logstash output --------------------------------
output.logstash:
  # The Logstash hosts
  hosts: ["localhost:5044"]
  bulk_max_size: 4096
```