A Network Telescope for Early Warning Intrusion Detection

Panos Chatziadam, Ioannis G. Askoxylakis, and Alexandros Fragkiadakis

FORTHcert, Institute of Computer Science Foundation for Research & Technology – Hellas (FORTH)
{panosc,asko,alfrag}@ics.forth.gr

Abstract. Proactive cyber-security tools provide basic protection as today's cyber-criminals utilize legitimate traffic to perform attacks and remain concealed quite often until it is too late. As critical resources, hidden behind layers of cyber-defenses, can still become compromised with potentially catastrophic consequences, it is of paramount significance to be able to identify cyber-attacks and prepare a proper defense as early as possible. In this paper we will go over the architecture, deployment and usefulness of a distributed network of honeypots that relies on darknets to obtain its data. As we have envisioned that such a system has the potential to detect large scale events as early as possible we have adopted the name Early Warning Intrusion System (EWIS).

Keywords: Human aspects of intelligence-driven cybersecurity.

1 Introduction

Honeypots (HP) can capture and identify not only known but also emerging cyber-attacks. As a HP is not generating any network activity, any traffic that is directed toward it, it is considered to be malicious. An active HP can interact with the attacking system and capture the data flow so that it can be analyzed at a later point. The attack can then be classified as a known attack with an established approach for resolution, or a new kind of attack that has to be analyzed in order to produce a way to act upon it. Honeypots that interact with the attacking system can be classified either as low-interaction or high-interaction. Low-interaction HPs will interact with the attacking system by realistically emulating specific Operating Systems (OS) and services thus allowing the full capture of the attack in progress. Beyond this emulation they will not allow further control to the attackers thus posing little security risk. High-interaction HP on the other hand, become fully exposed to Internet attacks. This method can provide a full view of the specifics of the attack as well as the technique the attacker has used to perform the attack but it's also quite risky as an attacker can establish control of them just like with any other real system. Often security analysts consider them unsafe to use as they pose a risk to real production systems should they be breached by an attacker [1].

T. Tryfonas and I. Askoxylakis (Eds.): HAS 2014, LNCS 8533, pp. 11–22, 2014.

While HPs can provide extensive information about an attack carried out to a specific system, utilizing specific methods and resources, they lack the capability of observing the big picture. Consequently, large scale events go unnoticed by HPs that can only focus on attacks carried out on them. If an attack is initiated towards an organization, unless it is directed to the HP itself first, there will be so little knowledge regarding the attack, that by the time it reaches the HP it may have affected many systems. A large deployment of HPs enhances the probability of the attacker hitting one of them; however, managing many HPs has been proven to be a prohibited affair [1].

Network telescopes (NT) allow the capture of attacking data on a wider scale. In principle, a NT allows the observation of large scale events by capturing traffic from dark (unutilized) Internet address spaces. The more dark address spaces a NT monitors, the higher its resolution is [1] thus making it capable of detecting a wider range of events. A NT is also safe to use as it passively captures traffic without interacting with it. A distributed NT [2] may consist of many devices (sensors) that reside on remote networks capturing traffic from dark address spaces and relaying it back to a central server so that it can be classified and analyzed. This is the approach we have adopted for our system which has been largely influenced, in regards to its architecture and implementation, by project NoAH [3]. This work will be presented as follows. Section 2 offers implementation insights whereas Section 3 goes through the components of the system. In Section 4 we present visualizations of the collected data, while Section 5 offers a discussion on EWIS' benefits and future potential. Lastly, we summarize in Section 6.

2 Concepts and Implementation

Our vision was to establish a system that would be cost effective to implement, easy to deploy and provide us with sufficient data to create an Early Warning System that could potentially detect large scale events on a global scale. A potential implementation of a NT was found to exceed our needs and expectations. As it operates on unused address spaces, all traffic reaching it can be classified as malicious thus avoiding unnecessary filtering of legitimate traffic. While there are two major types of NTs we have adopted our design to be of a passive NT as opposed to an active implementation. A passive NT can capture data from UDP and ICMP attacks making it possible to detect DoS attacks [4] and UDP worms. Nevertheless, it is incapable of detecting malicious traffic as a result of application exploitation of captured TCP traffic since it does not complete a TCP handshake [5].

As it is very difficult for an organization to dedicate a large number of IPv4 addresses to the NT, our design had to include a number of smaller NTs that would be combined to create a large NT. These smaller telescopes (sensors) would have to be easily deployable, fit easily within an organization's network infrastructure and be robust enough to continuously capture traffic and relay it back for analysis. As security is a major concern, the sensors would have to be built on a secured OS platform, run the bare minimum and be protected by a firewall. The fact that our

sensors utilize a passive approach to data collection is also a favorable point by an organization that certainly doesn't need any devices within their infrastructure to be talking back to a potential attacker. From a network topology point of view, the sensors should be positioned outside the organization's network on the Demilitarized Zone (DMZ) or even the public section of the network just outside the organization's firewall (Fig.1)

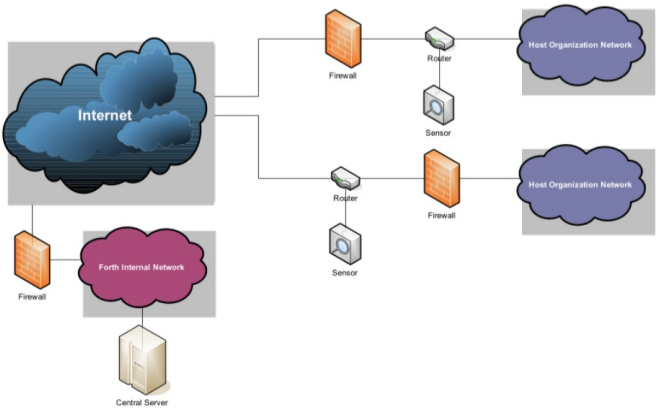


Fig. 1. Possible scenarios for sensor deployment

Isolating the sensor from the organization's Intranet and at the same time fully exposing it to the outside world is the best way for deployment. Should the sensor be compromised there would be no posed threat to the internal network of the host organization. In such an unlikely case, we can simply replace the sensor.

3 Architecture and Deployment

The previous sections have offered several hints as to how our NT is architected and implemented. We will look further into that in this section.

3.1 The Sensors

The initial sensor rollout consisted of small, inexpensive PCs preconfigured and ready to use (Fig.2). Recently, we are experimenting in utilizing very small computing platforms [6] as sensors, such as the Raspberry Pi [7]. A deployable bootable image is also under development. Software cost has also been minimized by utilizing in-house built applications and running them on a Linux-based open source OS.

Prior to deployment, each sensor is assigned a dark network space that could span from a few IPs to entire subnets. Assigning the dark space addresses to the network interface of the sensor was proven to be impractical and time consuming. Instead, we utilized a redirection on the ARP level [1]. When traffic arrives for a specific IP address the router broadcasts an ARP request for discovering the host. When the host

replies to the ARP request as the owner of the corresponding IP address, the router directs all traffic to this host. To utilize this methodology of traffic direction we used a daemon called *farpd*. For *farpd* to function, it has to be assigned the IP addresses it will respond for when receiving the gateway's ARP request. While this method is the most non-intrusive when deploying a sensor to an organization's network, it may prove to be inefficient and resource consuming when monitoring very large blocks of dark space networks [1]. In this case, a static redirection on the router level, for all traffic of the monitored dark addresses would be a more effective solution [1].

Fig. 2. Hardware used for initial versions of the sensor

Next, the *monitoring* daemon captures the received traffic and records a subset of the popular format NetFlow [8] to the database. This in-house developed daemon uses the *pcap* library to capture information such as the source and destination IP addresses, the source and destination ports, the flow payload size, the protocol type (number), the TCP flag (in case the flow is of TCP type) and the associated timestamp. As with *farpd* the *monitoring* daemon is also assigned the IP addresses that it should be recording traffic for, as well as the network interface in case the sensor has multiple interfaces (we would utilize multiple physical interfaces to place the sensor itself on a different network subnet than the dark space network it will monitor).

On a timely interval, the sensor will initiate a reverse SSH tunnel so that the central server will pull the collected data from the sensor's database. We utilize strong encryption and password-less shared key authentication for the tunnel. The reverse tunnel utilizes a predefined port and triggers a component on the server side that will perform the data collection. The sensor utilizes a number of PHP and BASH scripts for automating the process of daemon startup, tunnel opening, information transfer as well as database maintenance. SSH access is restricted to the IP of the central server as well to a specific IP of the host organization should they request remote access to the sensor. For this, as well as console operations in case of an emergency, the host organization is given access to a local user account on the sensor.

3.2 Central Server

The EWIS central server is where all data collected by the sensors is stored, processed, analyzed and presented. It is hosted at FORTH's main datacenter and it is maintained as a high priority system by FORTH's Systems and Networks team. The

hardware we are using for the central server is of server class with all possible redundancies for continuous operation and high availability. The server's PostgreSQL [9] database has been tuned to be able to cope with the volume of data downloaded from the sensors as well as to quickly respond to concurrent queries for the purpose of displaying the collected data. At the time that this document is written the *packets* table in the server's database has over 900 million records.

Every download from each of the sensors is logged and its timestamp is recorded so that downloads resume from that point forward. Each downloaded set consists of the captured traffic that has been recorded since the last download as well as the list of IPs the sensor monitors. This is useful to have in the database as it is used in filtering data when performing certain queries. Similarly to the sensors, PHP and BASH scripts are used for the automation of procedures such as the process of data transfer from the sensor, the parsing and commitment of data to the server's database, the monitoring of the sensors, the issue of alerts etc.

The sensors are monitored in regards to their ability to connect with the server and the integrity of the downloaded data set. If the sensor's communication is late, the state of the sensor is changed to potentially down while if the sensor fails to connect to the server the sensor's state is changed to down. If a sensor is flagged down, the support team will attempt to access the sensor to investigate the issue. If the sensor is not accessible from the server side, the issue will be communicated to a contact on the host organization so that the sensor's situation can be assessed from the console. Both the sensors and the server are monitored on a 24 hour basis so to provide an uninterrupted flow of data to the system.

4 Visualization Dashboard

The central server is utilizing a web interface for the users to interact with the data collected as well as perform some administrative functions. The web interface is comprised of a collection of PHP scripts that provide a functional and versatile environment. Access to this facility is given to the organizations that host a sensor as well as other individuals that may need access to the data for research purposes. Typically the host organization would only be allowed access to the data of their own sensor(s) while an admin type user will have access to the entire range of information stored.

The current implementation of this interface is rather minimalistic; however, a more comprehensive implementation is work in progress. There are several views that offer real-time queries to the statistics downloaded from the sensors.

4.1 Sum of Packets

The following three plots (Fig.3, Fig.4 and Fig.5) provide a trend of incoming traffic to the dark address spaces allocated to the sensor. Sudden changes observed from hour to hour (Fig.3) is an indication that an attack is likely to be taking place. The second and third graphs (Fig.4, Fig.5) display days of the week and weeks of the year respectfully and are meant more for statistical than alerting purposes.

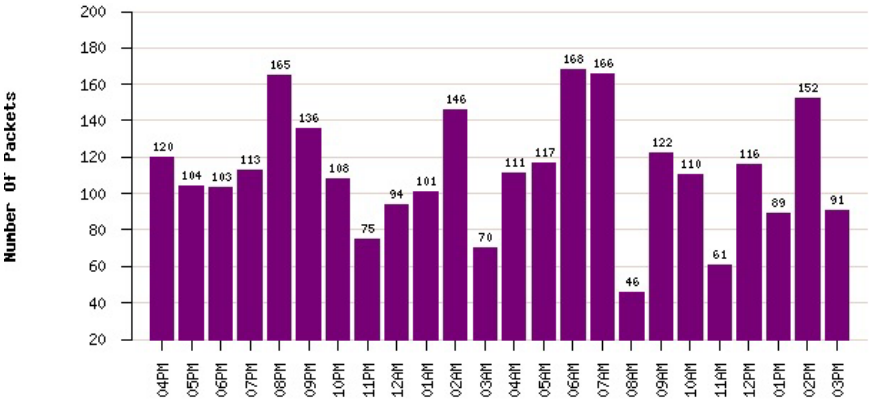


Fig. 3. The number of packets the sensor has received for each of the past 24 hours

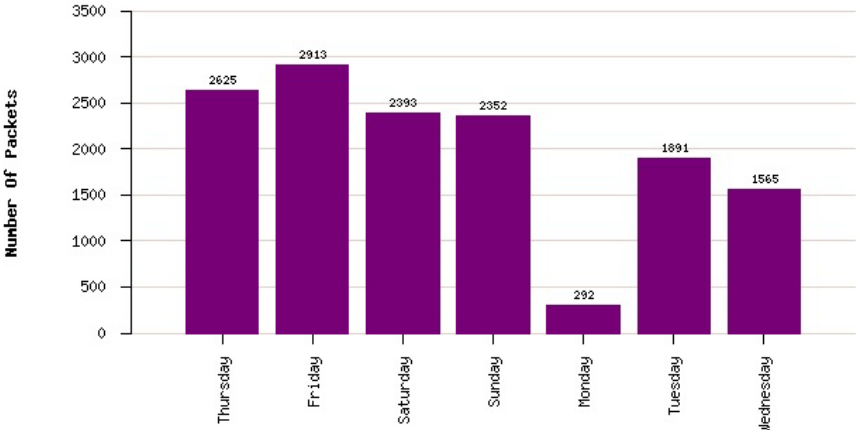


Fig. 4. The number of packets the sensor has received for each of the last 7 days

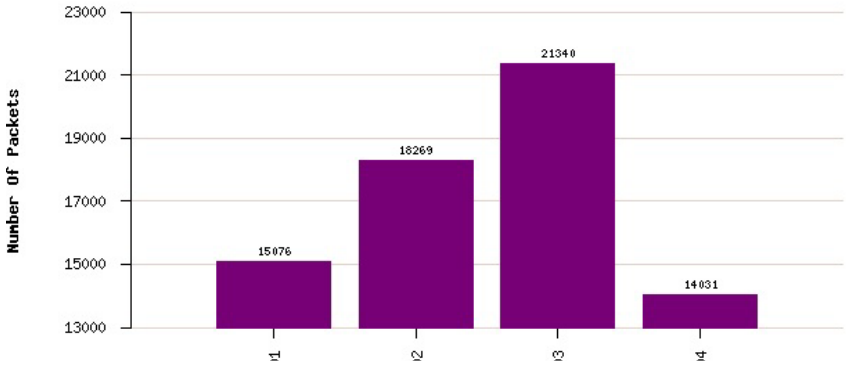
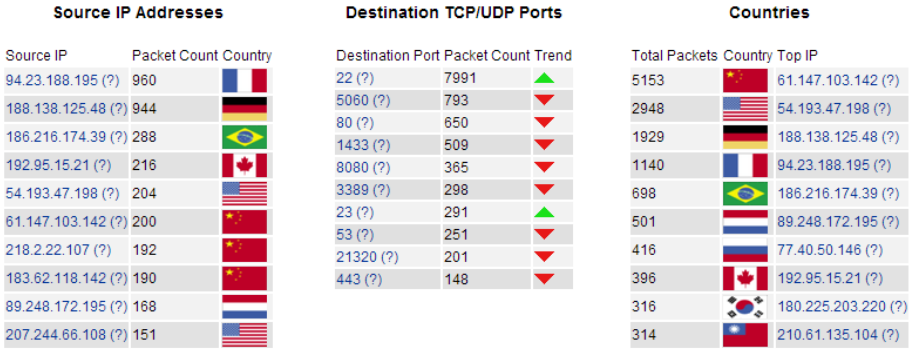


Fig. 5. The number of packets the sensor has received for each of the past 4 weeks

4.2 Top Statistics

This view has three tables that provide an assortment of top 10 statistics within a defined timeframe (Fig.6). For all three tables the user has the option to select the timeframe and top count option.



Source IP Addresses

Source IP	Packet Count	Country
94.23.188.195 (?)	960	
188.138.125.48 (?)	944	
186.216.174.39 (?)	288	
192.95.15.21 (?)	216	
54.193.47.198 (?)	204	
61.147.103.142 (?)	200	
218.2.22.107 (?)	192	
183.62.118.142 (?)	190	
89.248.172.195 (?)	168	
207.244.66.108 (?)	151	

Destination TCP/UDP Ports

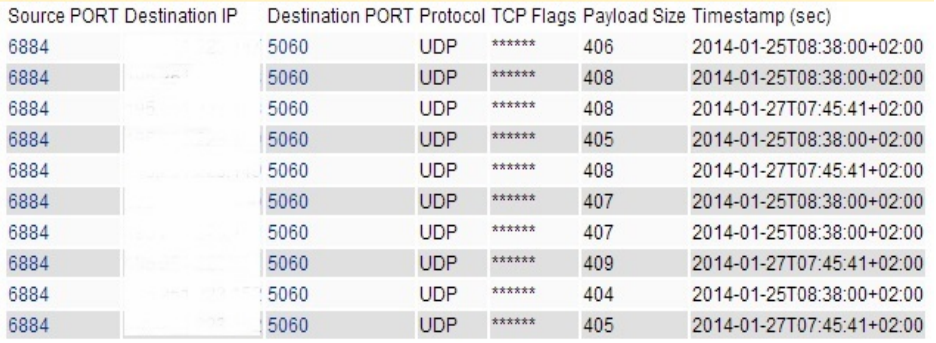
Destination Port	Packet Count	Trend
22 (?)	7991	▲
5060 (?)	793	▼
80 (?)	650	▼
1433 (?)	509	▼
8080 (?)	365	▼
3389 (?)	298	▼
23 (?)	291	▲
53 (?)	251	▼
21320 (?)	201	▼
443 (?)	148	▼

Countries

Total Packets	Country	Top IP
5153		61.147.103.142 (?)
2948		54.193.47.198 (?)
1929		188.138.125.48 (?)
1140		94.23.188.195 (?)
698		186.216.174.39 (?)
501		89.248.172.195 (?)
416		77.40.50.146 (?)
396		192.95.15.21 (?)
316		180.225.203.220 (?)
314		210.61.135.104 (?)

Fig. 6. Three tables providing statistics based on selected timeframe and top count

The left table, labeled *Source IP Addresses*, offers a list of the top 10 source IP addresses within the past 7 days. Next to each IP address is the accumulated number of packets the IP sent on the selected timeframe, as well as its country of origin. The flags displayed are the result of a match between the result of geoiplookup [10] and a local cache of flag icons. Selecting one of the source IP addresses the user will be presented with a list of all the IP flows received within the timeframe (Fig.7).



Source PORT	Destination IP	Destination PORT	Protocol	TCP Flags	Payload Size	Timestamp (sec)
6884		5060	UDP	******	406	2014-01-25T08:38:00+02:00
6884		5060	UDP	******	408	2014-01-25T08:38:00+02:00
6884		5060	UDP	******	408	2014-01-27T07:45:41+02:00
6884		5060	UDP	******	405	2014-01-25T08:38:00+02:00
6884		5060	UDP	******	408	2014-01-27T07:45:41+02:00
6884		5060	UDP	******	407	2014-01-25T08:38:00+02:00
6884		5060	UDP	******	407	2014-01-25T08:38:00+02:00
6884		5060	UDP	******	409	2014-01-27T07:45:41+02:00
6884		5060	UDP	******	404	2014-01-25T08:38:00+02:00
6884		5060	UDP	******	405	2014-01-27T07:45:41+02:00

Fig. 7. Truncated list of IP flows received within the past 7 day from the selected IP

Selecting the question mark next to the IP will perform a reverse DNS lookup and display the associated FQDN if one exists (Fig.8).

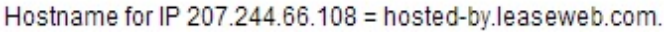


Fig. 8. FQDN name of selected IP

Selecting the source or destination port on the table of Fig.7 will result on a lookup of the port on an online resource database. If there are known applications that use the specific port, their names will be displayed (Fig.9).

sip	5060	TCP	IANA	SIP
sip	5060	UDP	IANA	SIP

Fig. 9. Destination port 5060 lookup

The table in the center of Fig.6, labeled *Destination TCP/UDP ports*, offers a list of the top 10 destination ports for the selected timeframe. Next to each port is the total packet count received for that timeframe and in the last column an indicator of the traffic trend. By selecting one of the ports, a list of all the IP flows received within the selected timeframe (Fig.10) is provided. The selection of the traffic trend indicator, offers a plot that illustrates the traffic trend of received packets for the selected destination port within the selected timeframe (Fig.11). This provides an explicit view of how incoming traffic changes for a specific timeframe while sudden rises suggest that perhaps a major event is taking place.

Source IP	Source PORT	Destination IP	Destination PORT	Protocol	Flags	Timestamp (sec)
85.25.199.95	5263	[illegible]	5060	UDP	0	2014-01-23T15:50:07+02:00
85.25.199.95	5263	[illegible]	5060	UDP	0	2014-01-23T15:50:07+02:00
85.25.199.95	5263	[illegible]	5060	UDP	0	2014-01-23T15:50:07+02:00
85.25.199.95	5263	[illegible]	5060	UDP	0	2014-01-23T15:50:07+02:00
85.25.199.95	5263	[illegible]	5060	UDP	0	2014-01-23T15:50:07+02:00
85.25.199.95	5263	[illegible]	5060	UDP	0	2014-01-23T15:50:07+02:00
85.25.199.95	5263	[illegible]	5060	UDP	0	2014-01-23T15:50:07+02:00
85.25.199.95	5263	[illegible]	5060	UDP	0	2014-01-23T15:50:07+02:00
85.25.199.95	5263	[illegible]	5060	UDP	0	2014-01-23T15:50:07+02:00
85.25.199.95	5263	[illegible]	5060	UDP	0	2014-01-23T15:50:07+02:00

Fig. 10. Truncated list of IP flows received within the past 7 days for the selected port

The rightmost table on Fig.6, labeled *Countries*, presents a top list of countries sorted by total packet number and the top source IP address from that country. This enables the viewer to quickly identify the top attacking country/IP address combination for a quick action against the attacking pair, such as submitting the information to the national CERT of the country in question. Selecting an IP address from the table will display a list of all the IP flows received from the selected source IP within the selected timeframe (Fig.7). As previously mentioned, selecting the question mark next to the IP will perform a reverse DNS lookup of the IP and display the associated FQDN if one exists (Fig.8).

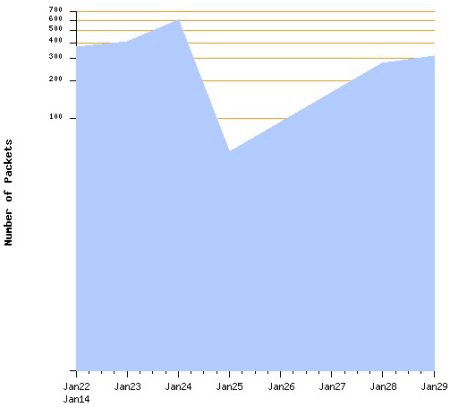


Fig. 11. 7 day traffic trend to destination port

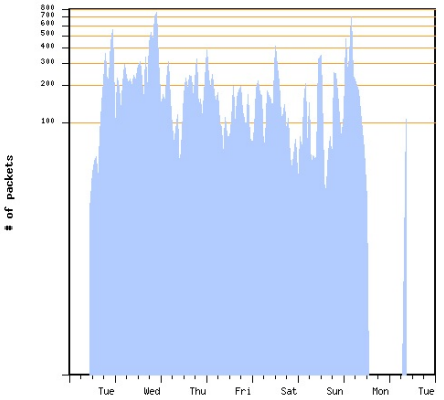


Fig. 12. 7 day trend of backscatter traffic

4.3 Protocol and Backscatter

The third view provides protocol breakdown statistics in three separate graphs (Fig.13). The left graph, labeled *Incoming TCP Traffic*, offers a view of the top ten destination ports and the total number of packets received for each port within the last 24 hours. In a similar manner, the right graph labeled *Incoming UDP Traffic*, offers a view of the top ten destination ports and the total number of packets received for each port within the last 24 hours. The bottom graph illustrates a breakdown of percentages of TCP, UDP and ICMP traffic received during the last 24 hours.

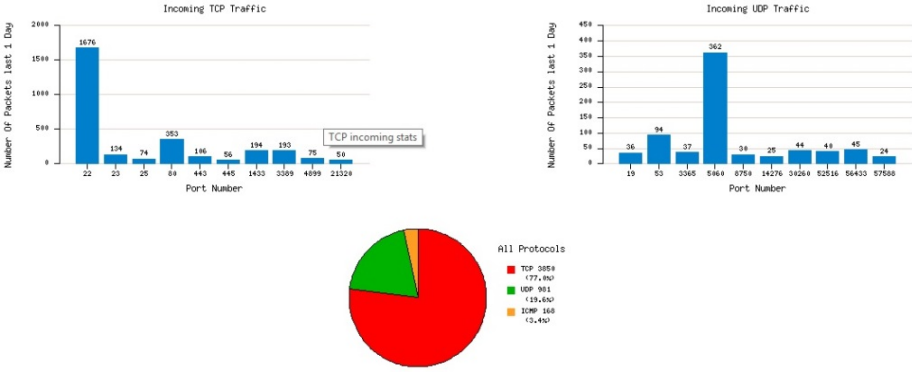


Fig. 13. Protocol Breakdown Statistics

By selecting the bar of a charted port on either the TCP or UDP graph the traffic trend of that port (Fig.11) is displayed. This provides a further understanding as to whether an attack is taking place or the activity of the port is normal as usual.

The sensors also receive responses from hosts that have received packets spoofed with sender's address to be one of the monitored dark space addresses. This unsolicited traffic, also known as Backscatter [11], is a known indicator or side-effect of a spoofed DoS attack. An ascending trend of Backscatter traffic is a well justified cause for alert. We provide a plot (Fig.12) of the Backscatter traffic each sensor receives over the period of one week. Associating the result of Backscatter traffic on a specific timeframe with spikes of traffic on specific ports we can begin to realize the detection of a DoS attack. Anomaly detection algorithms [12] can help automate the process making it possible to detect upcoming cyber-attacks before they reach their full potential.

4.4 Sensor Information

The Sensors view provides information on the sensor location and availability (Fig.14). This section is only available to admin users as it discloses privileged information regarding each sensor such as its IP address, geographical location and host organization. On the left side of Fig.14 there is a table showing the sensor names (hidden for privacy) and their availability status. As mentioned previously, should a sensor fail to contact the server an escalation to the EWIS support team takes place by utilizing email and SMS [13]. If needed, the host organization is contacted to provide hands on assistance to the hardware of the sensor.

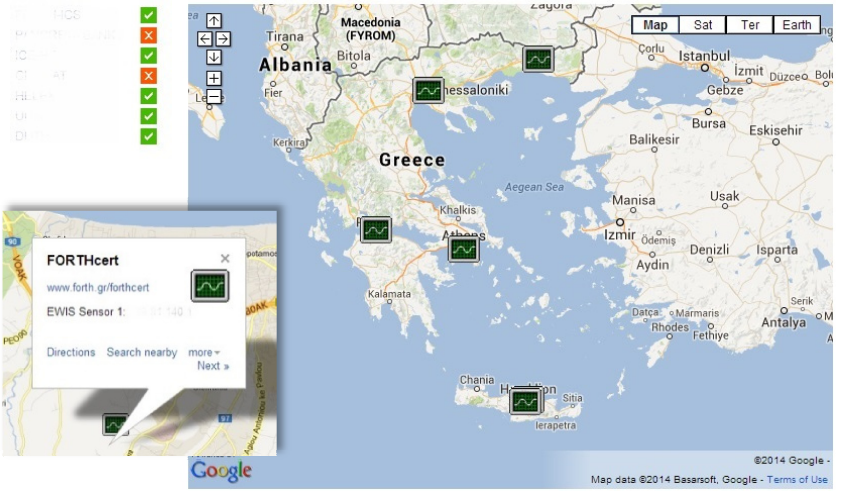


Fig. 14. EWIS Sensor Information and Availability

5 Discussion

EWIS provides a scalable platform for detecting emerging network threats on a large scale. By increasing the number of sensors the resolution of our NT also increases

providing more accurate results and less false positives when predicting upcoming attacks [1]. A large number of sensors will provide the resolution necessary to create an international Network Alert Weather Map.

Current means of visualization of the captured data provides a basic platform of analysis at a satisfactory level. We envision an automated procedure that will be able to correlate past and present data and provide results without human intervention. Algorithms of pattern recognition and anomaly detection [12] can further enhance the predictive capability and even provide an early warning alert of a large scale event at its early stages. An alerting system can subsequently inform our sensor hosts of an upcoming threat such as a Worm or Distributed DoS (DDoS) attack. DDoS attacks are considered a product of SYN Floods received from compromised hosts (zombies) [14] and can be detected by monitoring the backscatter traffic [11]. Worms can be detected by monitoring for excessively large portions of traffic. Known examples are Conficker [15], Blaster [16] and Nimda [16], all utilizing port 445/TCP used by CIFS [17].

At the time this document was compiled, a total of 7 sensors have been installed to an assortment of organizations ranging from small to large and from government operated to public sector. Although all functioning sensors are installed within Greece, a sensor was once hosted by the Computer Emergency Response Team of Austria (cert.at) [18] as part of a joined project with FORTHcert. To further expand the functionality of the EWIS NT we envision a merge with other NTs and Darknets [19] already functioning on the Internet such as the UCSD NT [20], the CCIED NT project[21], Oxford University's Darknet Mesh project [19] and Team Cymru's Darknet project[22]. We are also exploring the possibility of enhancing EWIS's capabilities by adding sensors that detect attacks on wireless networks[23].

6 Conclusion

The current deployment of EWIS has served well as an operational pilot and has provided an interesting assortment of data. Our main goal was to create a functional framework that would work as the basis of a distributed NT that could scale well beyond our national borders. The passive sensors deployed seamlessly on a host organization's infrastructure pave the road for a topologically diverse deployment. The security utilized for sensor to server transfers combined with tight software integration and a scalable database, makes EWIS a competent platform. The functional traffic analysis interface provides the means of basic data exploitation while the implementation of more advanced visualization tools is work in progress. The use of advanced backend data processing by utilizing anomaly detection algorithms [12] will assist in uncovering large scale malicious events such as DDoS attacks and worms.

By forming alliances with other organizations operating their own NTs we will be able to create a very large NT of global scale that could provide us and our partners the ability to have an aggregate view of Internet traffic across operational boundaries.

References

1. Irwin, B.: A framework for the application of network telescope sensors in a global IP network (January 2011)
2. Pouget, F., Dacier, M., Pham, V.: Vh: Leurre.com: on the advantages of deploying a large scale distributed honeypot platform. In: ECCE 2005, E-Crime and Computer Evidence, pp. 1–13 (2005)
3. Final Report - NoAH (NoAH: a European Network of Affined Honeypots) (2008)
4. Spyridopoulos, T., Karanikas, G., Tryfonas, T., Oikonomou, G.: A game theoretic defence framework against DoS/DDoS cyber attacks. Computers & Security 38, 39–50 (2013)
5. Cooke, E., Bailey, M., Watson, D., Jahanian, F., Nazario, J.: The Internet motion sensor: A distributed global scoped Internet threat monitoring system, 1–16 (2004)
6. Akram, R.N., Markantonakis, K., Mayes, K.: User centric security model for tamper-resistant devices. In: Proceedings - 2011 8th IEEE International Conference on e-Business Engineering, ICEBE 2011, pp. 168–177 (2011)
7. Raspberry Pi, `http://www.raspberry.org`
8. Bailey, M., Cooke, E., Jahanian, F., Myrick, A., Sinha, S.: Practical Darknet Measurement. In: 40th Annual Conference on Information Sciences and Systems (2006)
9. PostgreSQL, `http://www.postgresql.org`
10. Maxmind, `http://www.maxmind.com`
11. Moore, D., Shannon, C., Brown, D.: Inferring internet denial-of-service activity. ACM Transactions (2006)
12. Bhuyan, M.H., Bhattacharyya, D.K., Kalita, J.K.: Surveying Port Scans and Their Detection Methodologies. The Computer Journal 54(10), 1565–1581 (2011)
13. Akram, R., Markantonakis, K. (n.d.): Smart Cards: State-of-the-Art to Future Directions. crow.org.nz
14. Cooke, E., Jahanian, F., McPherson, D.: The zombie roundup: Understanding, detecting, and disrupting botnets. In: USENIX SRUTI Workshop (2005)
15. Symantec, W32.downadup, `http://www.symantec.com`
16. Cisco, Branch router QoS design, `http://www.cisco.com`
17. Internet file system, `http://www.snia.org`
18. Computer Emergency Response Team of Austria, cert.at
19. Oxford University, The Darknet Mesh Project, projects.oucs.ox.ac.uk
20. Caida, The UCSD Network Telescope, `http://www.caida.org/projects/network_telescope/`
21. ICSI, CCIED Network Telescope, `http://www.icir.org/vern/telescope.html`
22. Team Cymru, The Darknet Project, `http://www.team-cymru.org`
23. Fragkiadakis, A.G., Tragos, E.Z., Tryfonas, T., Askoxylakis, I.G.: Design and performance evaluation of a lightweight wireless early warning intrusion detection prototype. EURASIP Journal on Wireless Communications and Networking (1), 73 (2012)