Real-time Distributed Malicious Traffic Monitoring for Honeypots and Network Telescopes.

Samuel Oswald Hunter
SensePost
Pretoria, South Africa
Email: sam@sensepost.com

Barry Irwin
Dept. Computer Science
Rhodes University
Grahamstown, South Africa
Email: b.irwin@ru.ac.za

Etienne Stalmans
SensePost
Pretoria, South Africa
Email: etienne@sensepost.com

***Abstract*—Network telescopes and honeypots have been used with great success to record malicious network traffic for analysis, however, this is often done off-line well after the traffic was observed. This has left us with only a cursory understanding of malicious hosts and no knowledge of the software they run, uptime or other malicious activity they may have participated in. This work covers a messaging framework (rDSN) that was developed to allow for the real-time analysis of malicious traffic. This data was captured from multiple, distributed honeypots and network telescopes. Data was collected over a period of two months from these data sensors. Using this data new techniques for malicious host analysis and re-identification in dynamic IP address space were explored. An Automated Reconnaissance (AR) Framework was developed to aid the process of data collection, this framework was responsible for gathering information from malicious hosts through both passive and active fingerprinting techniques. From the analysis of this data; correlations between malicious hosts were identified based on characteristics such as Operating System, targeted service, location and services running on the malicious hosts. An initial investigation in Latency Based Multilateration (LBM), a novel technique to assist in host re-identification was tested and proved successful as a supporting metric for host re-identification.**

***Index Terms*—remote fingerprinting, data mining, situational awareness, monitoring**

I. Introduction

NUMEROUS techniques have been developed to combat malicious agents on the Internet and improve network defenses. These technologies include anti-virus, intrusion detection systems, intrusion prevention systems and firewalls to name a few. The techniques, however, are fallible and do not provide complete protection against malicious activity. On a daily bases many of the hosts connected to the Internet still endure reconnaissance probes and attacks from malicious agents.

These agents may be categorised as self propagating worms or hackers. In the cases were the source of malicious activity is human driven, it becomes much harder to defend against. These agents have different motivations such as monetary gain, political agenda, espionage or sabotage. It is important to note that the single underlying similarity between all of these agents is that their actions originate from a device or host that is connected to the Internet.

Existing literature explores the characteristics, actions and motives of these agents in thorough detail. However, a conclusive body of work that focuses on the logical and physical representation and associated characteristics of these hosts does not exist. There is interest from both academia and industry to better understand the representation of these hosts, the scale of penetration and locality of their presence with regards to critical infrastructure. If these hosts could be identified, analysed and tracked it could allow researchers to develop better countermeasures to protect critical infrastructure and model the risk posed by them.

For sufficient data collection to occur distributed data sensors would need to be used, this is needed as malicious agents have been found to, in some cases, be locally biased. To insure that data collected is relevant and could be used to gather further information from the malicious sources it would need to be available for real-time analysis. This is motivated by the dynamic and interconnected nature of the Internet, hosts are connected and reachable in one moment, may not be available at a later stage. With these requirements and constraints in mind, an application was implemented for use with honeypots and network telescopes. This application, named real-time Distributed Sensor Network (rDSN) made use of a RabbitMQ[1] messaging framework and was implemented in Python.

In order to increase situational awareness with regard to these malicious agents, the AR Framework was developed. The framework was based on an adapted multi-sensor data fusion model[1][2]. This model assisted in the aggregation and interpretation of multiple heterogeneous data sensors, reconnaissance modules and *priori* data. The AR Framework would make use of various remote fingerprinting techniques to enumerate characteristics of hosts detected by the rDSN application. Through the use of network telescopes and honeypots as data sensors, the AR Framework could collect data targeted specifically towards malicious hosts, in real-time.

A new method of fingerprinting was developed; Latency Based Multilateration(LBM). LBM was used to create a logical representation of the physical location of a host on network infrastructure. ICMP Ping requests would be sent from three geographically separate locations to a given host. By removing latency outliers and calculating the average times from each location, a 3-tuple value would be produced. This value could

[1]RabbitMQ - Enterprise messaging system based on the emerging AMQP standard (http://www.rabbitmq.com/)

be used as a supporting metric for host re-identification. LBM based comparisons between hosts were achieved by mapping the 3-tuple values to euclidean 3-space and calculating the distance between the two points in space. If this distance was smaller than a predetermined threshold, it could indicate that the previously observed host and current host are the same. Part of this research shows the calculation of optimal values for this threshold variable.

II. State of the Art

Capturing malicious traffic and more importantly for this research, identifying the sources of malicious activity required appropriate sensors to be deployed. Honeypots and network telescopes were used for this purpose. This section briefly outlines on the concept of network telescopes and honeypots and their use for capturing malicious traffic and monitoring unused IP address space.

A. Network Telescopes

The basic setup for a network telescope is to have a server to which traffic that would normally be destined for unused address space is forwarded. Different configurations exist for network telescopes, some respond to incoming traffic (Active Telescopes) such as the IMS which makes use of a lightweight responder[3] while others capture all traffic forwarded to them without any response (Passive Telescopes).

A network telescope's ability to capture traffic destined for unused address space results in it receiving very little, if any, legitimate traffic. The two exceptions here are traffic caused by mis-configured hardware and certain types of DDoS attacks[4]. The traffic captured by the network telescope is thus highly suitable for providing us with information regarding events such as denial of service attacks[4] and the automated propagation of Internet based worms and viruses[5]. When considering the traffic observed by a network telescope other researchers refer to the term "backscatter"[6][5][4] which implies residual traffic observed from other hosts that may have been the target of distributed denial of service attacks and are responding to spoofed source addresses. Other traffic such as network scanning from worms and malicious users also amount to backscatter while a very small portion of backscatter is the result of mis-configured hardware[6].

Past research in the field of unsolicited traffic analysis has proven successful in identification and tracking of Distributed Denial of Service(DDoS) attacks[7][4] , worm propagation[5][8] and network scanning[9]. None of this research was performed in real-time, all relying on data captured by network telescopes and off-line analysis.

Even though malicious traffic such as DDoS and worm propagation are globally scoped, data from the IMS indicate widely different trends between separate network telescopes[10]. These differences were noted across three dimensions namely; over all protocols and services, a specific protocol and port and lastly signatures of known worms. Another publication[11] shares this view that multiple points of monitoring are required coupled with a collective interpretation to provide a more comprehensive view of malicious network traffic. Thus observing traffic from only a single point would provide very little, if any information about the background activities[11] as a whole.

Contrasting this view, there are, however, still important information that can be learnt from a single vantage point,[5] showed how a small /24 telescope was used to identify and distinguish between port scans, host scans and DDoS attacks. While the results found in[5] might not correlate strongly with global traffic, their findings were still useful in understanding current threats.

B. Honeypots

Honeypots are very similar in nature to network telescopes, they are, however, at the other end of the spectrum with regards to capturing nefarious traffic. This is because honeypots normally emulate a vulnerable service or host in order to elicit malicious interaction. Thus acting as a decoy vulnerable system and soliciting bi-directional interaction with malicious agents. Honeypots have multiple uses ranging from capturing malware, vulnerability exploitation, active network defenses, credentials used in attacks and detecting the compromise of a system[12][13][14]. Honeypots may be implemented to emulate any services, some even have the ability to emulate unknown protocols, such as the Dionaea with the NFQ module enabled[15]. The Kippo honeypot emulates an SSH service with authentication credentials, capturing the keystrokes of attackers and any files they attempt to download. It has also been suggested that honeypots be used in an offensive manner to strike-back at attackers or provide false information that could compromise an attacker or lead them astray during their attack methodology[16][12].

Honeypots could be deployed using a single IP address[15] or function over a range of addresses, an example of multiple honeypots over a range of IP addresses is known as a honeynet and could be deployed using a tool such as honeyd[2]. Some of the differences between honeypots and a network telescopes include the increased resource requirements of honeypots and the methods of capturing malicious traffic. The active network telescope implemented by IMS starts to bridge the gab between network telescopes and honeypots, if only slightly through its ability to complete TCP handshakes. As such honeypots interact in greater detail with threats (such as a worms) and are thus more resource intensive and as a result need to be sufficiently hardened to avoid the compromise of the actual honeypot.

In conclusion, honeypots are able to characterise attacks against services in greater detail while also granting them the ability to capture malware and vulnerability exploitation that requires asynchronous communication as opposed to a malicious agent such as the MSSQL Slammer worm that propagates with a single UDP packet which contains both the exploit and payload[17].

III. Design and Implementation

Today the Internet consists of a massive and dynamically changing network of hosts. While many of these hosts are

[2]honeyd - Virtual honeypots (http://www.honeyd.org/)

on-line and reachable for extended periods of time, there is another large set of hosts that are not. These hosts may not spend the majority of their lifetime connected to the Internet, in other cases the hosts may not have static IP addresses and as such may only be reachable on a given IP address for a relatively short period of time. As this research is concerned with the analysis and data mining of malicious hosts connected to the Internet, it is of great importance that the information retrieval process is started as soon as a malicious host it detected. As the elapsed time increases before the reconnaissance is completed, the probability that a host may no longer be on-line or reachable on the originally detected IP address also increases.

A. rDSN Publisher

In order to address the need for real-time data exposure from the network telescopes and honeypots, the development of a rDSN publisher was undertaken.

The primary purpose of the rDSN publisher was to act as a packet sniffer on network telescopes and data relay service on honeypots. Once data had been detected it would be published in real-time. This was achieved by making use of AMQP and a RabbitMQ[3] broker. A publish/subscribe model was chosen as it allowed for scalable data exposure and could assist in distribution of work to analysis modules, if need be.

Packets captured on the listening interface would be inspected and constructed into objects. For Dionaea honeypots, the SQLite database used to store events was polled and new data extracted. After transformation the object data would be serialised using JSON encoding and published to the appropriate exchange on the configured RabbitMQ message broker. Publishing to different exchanges allows for more control over the dissemination of data, for instance access to data may be granted to third parties based on exchange name.

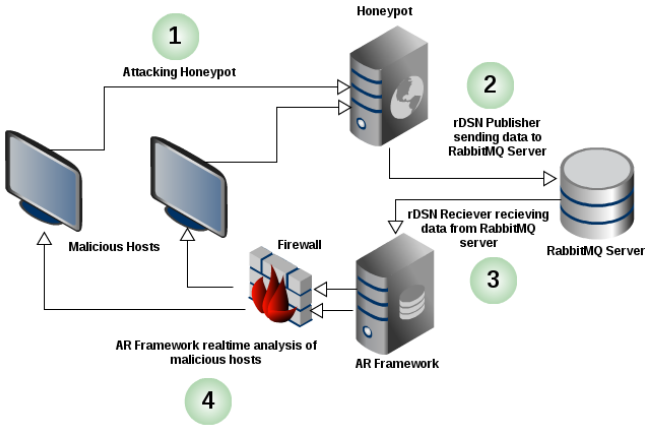

Figure 1. High level overview of rDSN deployment and integration with the AR Framework.

B. rDSN Receiver

The rDSN Receiver would subscribe to queues on the RabbitMQ broker that had been assigned to the various data sensors. As data is published to the RabbitMQ broker, it would be pushed to all rDSN Receivers; in real-time. Data from the network telescope consisted of packet data. This data was serialised with the help of JSON encoding before transmission and de-serialised by the DataSource Manager in the AR Framework. Data from honeypots were represented as Malicious events as apposed to packet data.

The rDSN Receiver was incorporated into the AR-Framework and formed part of the Collection phase shown in Figure 2. The integration of the rDSN Receiver into the AR Framework serves as an example of how the rDSN application could be easily deployed with other applications.

C. AR-Framework

During the design process of the AR Framework it became apparent that information flow was going to play an important role during the analysis of malicious hosts. Combined with the large number of responsibilities the AR Framework was tasked with it was decided to divide and group similar functionality according to task. This allowed for clearly defined inputs and outputs from each of the four components within the AR Framework as well as simplifying the process of extending the Framework as might be required at a later stage.

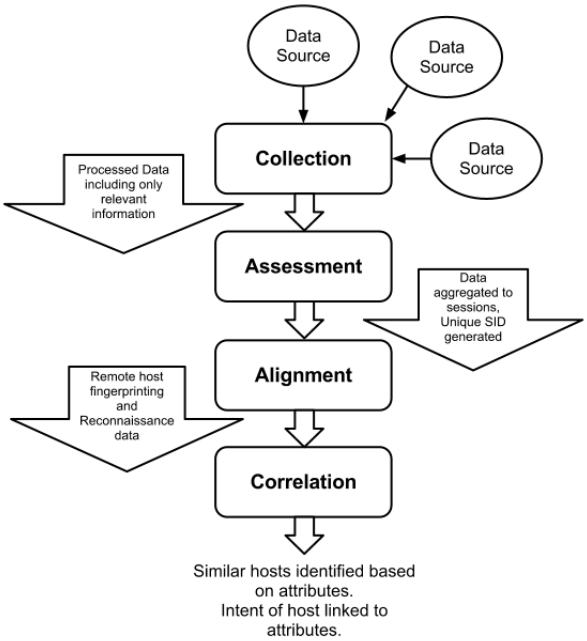

Figure 2. Overview of the core components within the AR Framework.

The first core component of the AR-Framework, namely the Collection phase, was responsible for subscribing to the appropriate message queues on the RabbitMQ message broker through the incorporation of a rDSN Receiver. In addition to managing the various data sources, the Collection Phase was also responsible for filtering the traffic and events it captured, ensuring that only valid and appropriate data from the various data sources was used by the AR Framework. The Traffic Filter module also attempted to identify the local service being targeted on a data sensor by correlating the targeted port with

[3]RabbitMQ - Messaging framework (http://www.rabbitmq.com/)

IANA[4] port numbers. This type of filtering forms part of Level 0 Data Refinement during the data fusion process[18][1]as defined by our adapted multi-sensor data fusion model[2].

The reconnaissance process was divided into two phases: Assessment and Alignment. Figure 2 illustrates the Assessment phase leading into the Alignment phase through the generation of unique Session Identifiers and aggregation of incoming traffic into appropriate sessions. The AR Framework session management maintained current traffic flows. After a new session had been created and an SID generated, that information would be passed to the Alignment phase. During this phase active fingerprinting would occur, this was achieved through a collection of reconnaissance modules, each with their own task. Tasks included operations such as Nmap scans, IP geolocation translations, *priori* data lookups and LBM fingerprinting. Lastly during the correlation phase data from the previous phases would be aggregated, stored and used to determine if other hosts previously identified and fingerprinted, represented similar characteristics to newly fingerprinted hosts.

D. LBM Fingerprinting

Latency Based Multilateration (LBM) requires that the hosts acting as *Base Stations* are geographically separate. This separation naturally translates to different locations on the physical infrastructure that represents the Internet. The difference in physical location implies that measurements from each of the *Base Stations* to some unique entity, would result in different ranges of round-trip (RTT) latency incurred. This is a result of the physical distance between the different *Base Stations* and a targeted host. Some of the *Base Stations* may be closer to the target and others further and under normal network congestion should maintain a constant latency. Packet-switched networks incur various delays while transferring data and LBM takes advantage of these delays in order to remotely fingerprint hosts. While the delays in packet-switched networks make it possible to measure a form of distance between hosts, it may also obscure results during times of abnormal load.

The process of latency multilateration is started when a set of ICMP type 8 (echo/ping requests) are sent to a host from the three *Base Stations.* When the ICMP type 0 (echo replies) are received by the *Base Stations* the RTT are recorded. Outliers are removed by extracting the values that fall between the 25th and 75th percentile in each respective set of RTTs. However in cases where 40% or more of the sets RTTs are represented by the mode, the mode is used. In statistics the mode is the value that occurs most frequently in a data set. The the resulting subset of values from the 25th to 75th percentile are then used to calculate three mean values; one for each of the three *Base Stations*. Once again if the mode value was selected in a particular set, it is used instead of calculating a mean value. This produces a 3-tuple representation of the average time taken by each set of ICMP ping requests. The process of comparing these results with others to determine the likelihood of two hosts being the same is a non-trivial problem, especially with large datasets. To overcome this the 3-tuple is mapped to euclidean 3-space, representing each of the three mean timings as a value on a x, y and z axis. This results in a series of points in space which allows for easier comparisons between hosts as the distance between two points in space is a trivial calculation as shown with Algorithm 1. A threshold value is then used to determine if the distance between two points is small enough to conclude that they may be the same host. The optimal value for this threshold variable and the results to support it will be shown in Section IV-D of this work.

Algorithm 1 Adapted Pythagoras for 3 dimensional Cartesian plane used to calculate the distance between two points in space.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

E. Data Collection

Datasets were collected over a period of 2 months. The data collected with the AR Framework was saved as two separate datasets, the first spanning the month of September in 2012 while the second was recorded over October in 2012. During analysis data was often grouped according to the different sensors that were used, these included two honeypots; one called EC2Honeypot and the other EliteHoneypot. Finally the network telescope used was called RUScope. These datasets were collected with some interruption due to connectivity problems with ISP that was used. However, it will be shown that the data collected from honeypots and network telescope with the aid of the AR Framework remained mostly consistent over time.

IV. Results

Data collected by the AR Framework provided insight into characteristics of malicious hosts as observed by the honeypots and network telescope used during this research. This data was analysed from different perspectives in order to provide a holistic representation of the data collected. Firstly the geographic locality of malicious hosts were determined, this data could be used, as an example, to construct black lists to stop potentially malicious traffic at the firewall level. The services targeted by malicious hosts were then determined and counted across the different types of data sensors used. This was done in order to determine the similarity of services being targeted across different IP ranges. It also provided insight into the most "attractive" targets for malicious agents, ie. what services they target above others. With these services in mind, networks could be better secured (as a first line of defensive) by ensuring that if they run any of these identified services, that they are maintained and not vulnerable to attack. The tables were then turned, using data collected by the AR Framework information regarding the malicious hosts were analysed, this included the Operating Systems (OS) of malicious hosts and the services they run (as determined by open ports). Furthermore, the most common port combinations were determined and correlated with the identified OS's. This information provided insight

[4]IANA - Service Name and Transport Protocol Port Number Registry (http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xml)

into the characteristics of malicious hosts, how they may have been compromised and what further dangers they may pose.

Lastly LBM fingerprint data was analysed to determine the optimal values for a threshold variable and the feasibility of using it as an additional metric for host fingerprinting such as host re-identification.

A. Geographic traffic analysis

A MaxMind[5] City database was used to the geographic location of IP addresses; translating them to latitude and longitude co-ordinates. Figure 3 Shows points on a map illustrating the locations of malicious hosts as determined by the EC2Honeypot (Green) and the EliteHoneypot (Blue). This kind of map gives a quick indication of were malicious hosts targeting infrastructure are located. There are not many points on this map as it only represents hosts detected by two honeypots, each on a single unique IP addresses as apposed to hosts identified by the RUScope network telescope (see Figure 5).

Figure 3. Malicious host locations as identified by the EC2Honeypot (green) and EliteHoneypot(blue).

For a more in-depth look at the data from Figure 3, the top 10 countries from which malicious hosts originated from were determined. These countries have been shown in Figure 4. It is clear that the highest proportion of malicious activity originated from China, the next highest, with only a quarter as many hosts identified, was the United States. These two countries represent the top contributors of malicious hosts that targeted the two honeypots over a two month period. These results become more interesting when compared to the data collected by the network telescope, this telescope operated over 254 Internet facing IP addresses, allowing it to capture a great deal more malicious traffic than the two single honeypots. Interestingly, the proportions of countries identified as the top 10 sources do not differ much between the honeypot dataset and network telescope. As such it contrasts the view of Shinoda[11] by indicating that similar data could be collected from a small source as-well as a larger one, at-least with regard to the geographic location of malicious hosts.

The top two countries (China and the United States) stay at the same positions, and another 3 of the countries may be found in both lists, thus in the top 10 countries identified between the honeypots and network telescopes, the datasets share 50% of the same countries.

[5]MaxMind - Determining geographic locations from IP Addresses (http://www.maxmind.com/en/home)

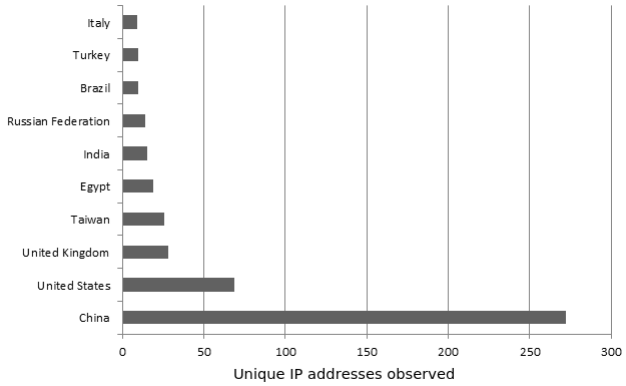

Figure 4. Top 10 countries from which malicious hosts were identified from honeypot data.

Figure 5. Malicious host locations as identified by the RUScope network telescope.

Figure 6 and Figure 7 show the countries of origin of malicious hosts targeting the RUScope network telescope for two successive months respectively. Figure 6 represents the September 2012 dataset while Figure7 shows data collected during October 2012. Between these two data sets, 90% of the countries are the same, the only countries that have changed between the two sets were the Republic of Korea (shown in Figure 7) and Italy (shown in Figure 6). Once again the top two countries were China and the United States, the countries in position 3 to 8 do not differ greatly nor does the number of hosts observed from each country between the two separate months.

These similarities speak to the predictable and reliable measurements from network telescopes and honeypots. These types of sensors provided relatively consistent data over the two month period and as such, could easily be used to observe new trends and sudden changes in the climate of malicious activity on the Internet. Examples of this could include the outbreak of a new self propagating worm or activity of new, significantly large botnets. To some extend, data such as this could also be used to measure the effectiveness of botnet take down operations.

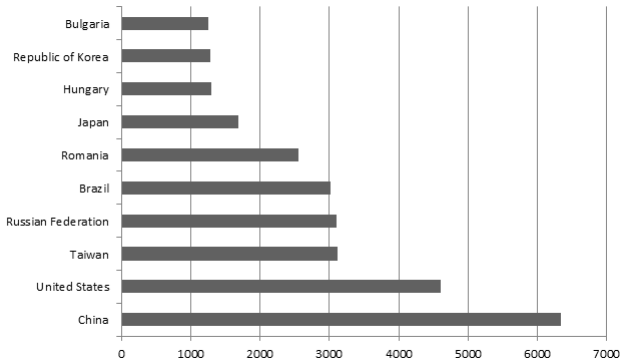

Figure 6. Top 10 countries from which malicious hosts were identified from network telescope data (September dataset).

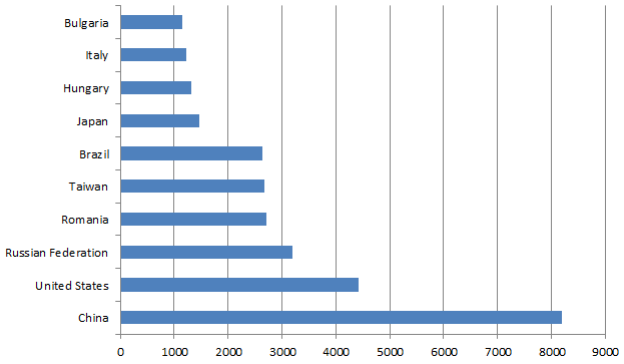

Figure 7. Top 10 countries from which malicious hosts were identified from network telescope data (October dataset).

B. Targeted Services

By investigating the ports targeted on the honeypots and network telescope, it is possible to identify not only the most targeted services on the Internet, but also the services that were likely used to compromise the malicious hosts in the first place. Figures (8 and 9) summarise the top 10 ports that were targeted on the honeypots and network telescope. Along with port numbers, the common services operating on each port, as taken from IANA numbering, has been listed. Figure 8 shows a relatively even distribution across the top 3 targeted services (Telnet 24.2%, MSSQL 24.04% and RDP 21.4%). Interestingly both Telnet and RDP are used for remote system management, while exploits exist to compromise implementations of both these services (if the version is old enough), it is far more common to find these services being attacked by credential brute-forcing or dictionary based attacks. The likely hood of success towards compromising a system by guessing credentials would be substantially higher as it does not depend on a specifically vulnerable version. While MSSQL is not used for system management like Telnet or Remote Desktop Protocol (RDP), it could be used to achieve complete compromise of a host if the server administrator (sa) account was present and not locked down. This often happens when credentials are guessed and would provide System level access to that host.

The last 7 services also have a relatively even distribution, ranging from 7.3% for SMB, 2.6% for VNC and an average

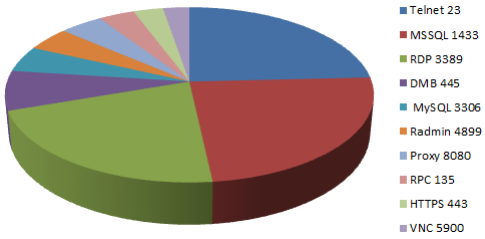

Figure 8. Top 10 services that were targeted on the EC2Honeypot for both September and October 2012.

of 4% for the remaining services. All of the services targeted here either provide some form of remote system administration (Radmin, RPC, VNC or provide functionality that could be abused or lead to compromise through exploitation of known vulnerabilities (in vulnerable versions). Figure 9 shows the top 10 targeted services on the RUScope network telescope, here SMB dominates with 75.8%. Next is RDP with 18.3% and the remaining 8 services range between 1.3% for Telnet to 0.43% for the unknown service running on port 12804. These two Figures seem to agree with Shinoda[11] stating that a single source for capturing malicious activity could not accurately represent the extrapolation of such activity. While they both share some of the same services, the distributions are significantly different. Indicating that different types of malicious agents are attacking different network ranges and that some sort of biased may exist. The large number of malicious traffic destined for port 445 (SMB) could be attributed to Conficker as it has been, to date, the most prolific worm to target the SMB service.

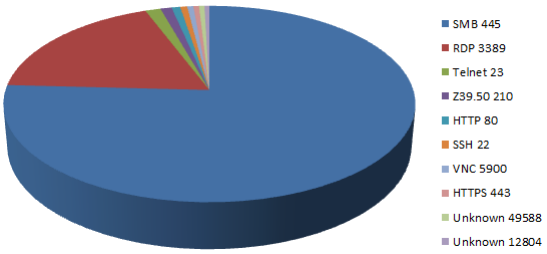

Figure 9. Top 10 services that were targeted on the RUScope network telescope for both September and October 2012.

C. Open Port Statistics of Malicious Host

While port scanning has been a well defined technique for many years, especially in the realm of information security, to our knowledge, it has never been used against malicious hosts for the purpose of research. By analysing port data from the AR-Framework it was possible to identify interesting correlations between open ports and the OS of malicious hosts. This data also varied significantly according to the sensor used to detect malicious interaction (Honeypot vs. Network telescope).

Figure 10 shows the top open ports found on malicious hosts as a proportion, grouped by the data sensor used to detect the hosts. From this graph it is clear that port 3389; commonly used for RDP, was the most common open port

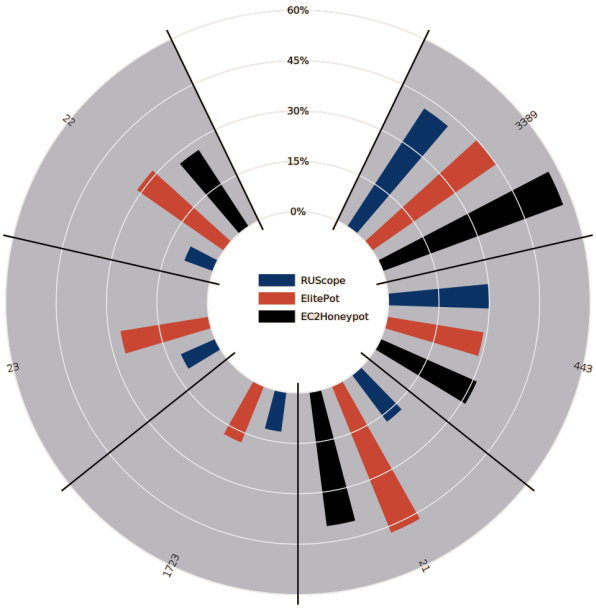

Figure 10. Most common open ports detected on malicious hosts, grouped by data sensor.

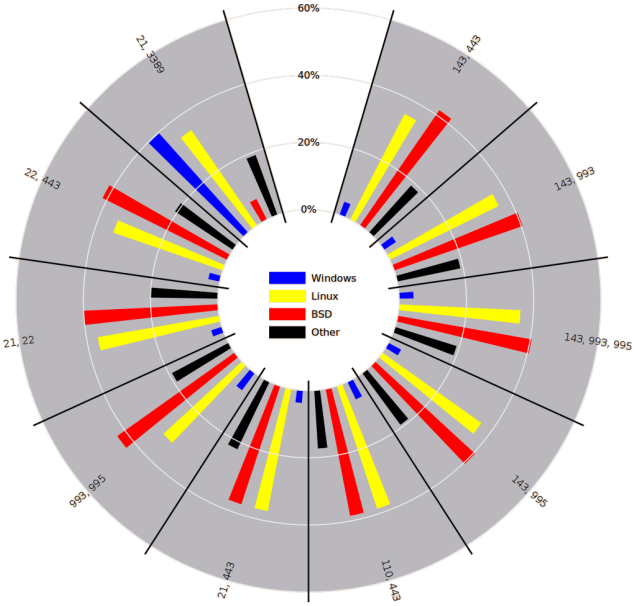

Figure 11. The top combinations of two or more open ports on malicious hosts along with the corresponding operating systems from the EliteHoneypot.

across detected malicious hosts. A close second but with far less open ports detected from network telescope data was port 21; used for FTP. The next port was 443, used for HTTPS with almost the exact same distribution across all sensors. When hosting malicious content such as malware it becomes advantages to do so over an encrypted channel such as SSL in order to circumvent network based IDS software. This simple IDS evasion technique could explain the way port 80 does not feature in the top 6 ports and rather 443. The last two notable ports were 22 and 23; used for ssh and telnet respectively. Both used for remote administration of a system. Interestingly 3 of the top open ports found across malicious hosts are used for interactive remote management, making them attractive targets due to the access normally granted by them.

For Figures 11 and 12 the top combinations of two or more open ports were taken from single sensors and grouped by operating system. The network telescopes grouped in Figure 3 and the EliteHoneypot in Figure11. The OS was determined through Nmap scanned and grouped according to major OS family types, namely Windows, Linux, BSD and all other OS types.

Figure 11 clearly shows Linux and BSD systems dominating all port combinations except for the port 21 and 3389 combination, as would be expected with RDP on Windows. Explanations for the large proportion of 3389 port instances found on Linux hosts could be as a result of Nmap miss-characterisation, port forwarding or other services running on 3389. As would expected one of the most common combinations of ports include 110, 143, 993 and 995 often also grouped in different permutations. These four ports are most often used for providing email services; 110 for POP3, 143 used for IMAP and their encrypted counterparts; 993 IMAP over SSL, with 995 being used for SSL over POP3. These compromised hosts are likely used for sending phishing emails; this theory is motivated more by the explicit combinations of port 110 and 443 as well as 143 and 443 were mail servers may be used to send phishing mails or spam, while simultaneously hosting content for the particular campaign on port 443 (HTTPS).

The contrast and suggestion of locally biased network scanning as suggested by [10][11] is evident when comparing the host operating systems and port combinations between the honeypot data in Figure 11 and the network telescopes data in Figure 12. The top port combinations from the network telescope data clearly favors Windows hosts, which is evident as well with RDP port 3389's presence in the majority of port combinations. A new port also features in the network telescope dataset, namely 1723 which is commonly used for the Point-to-Point Tunneling Protocol (PPTP). Port 5900; another remote administrative port is observed. Port 5900 often used for Virtual Network Computing (VNC) offers a similar service as RDP.

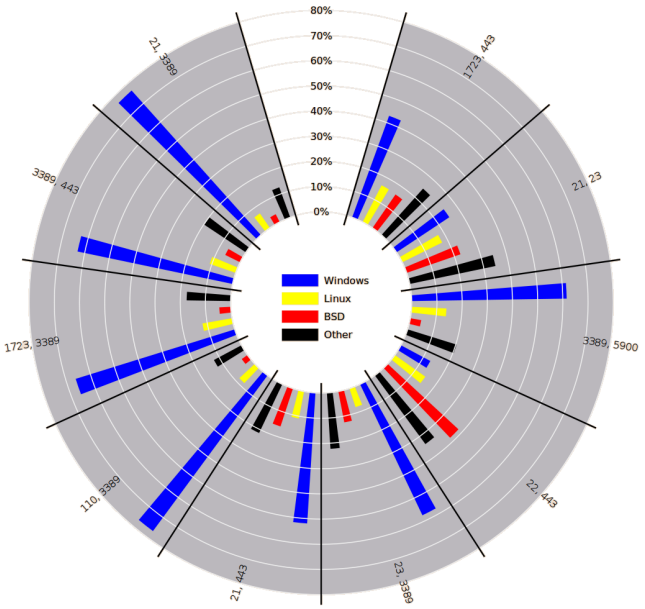

Figure 12. The top combinations of two or more open ports on malicious hosts along with the corresponding operating systems from the RUScope network telescope.

One of only two port combination from the network telescope dataset that was not dominated by Windows systems was the port 22 and 433. SSH (port 22) is considered the De-facto remote system administration tool for Linux and BSD based systems, which is re-affirmed from the lack of Windows representation in this combination. The second combination not dominated by Windows hosts was for ports 21 and 23; FTP and Telnet respectively. This category was dominated by "Other" operating systems that were not categorised directly under the 3 previous OS families. It is likely that these hosts represent devices such as routers and switches located on public Internet facing endpoints. These devices are often administered over SSH or Telnet.

D. LBM Analysis

A feasibility study for LBM fingerprinting revealed that hosts could be re-identified with an accuracy between of 94% and 96% using threshold values between 85ms and 125ms. The accuracy of host re-identification based on only LBM Fingerprints was measured using set containing both legitimate and malicious hosts. Two separate datasets each monitoring one legitimate host (A and B) that had been fingerprinted using LBM every hour over a period of three months was used to pick two LBM fingerprints at random, one of these fingerprints was then inserted into a set including all hosts monitored over September and October 2012. The difference or LBM distance between the second LBM fingerprint and all hosts in the set were then calculated. Hosts who had a LBM Difference that was equal to or smaller to a threshold were then considered to be the same, according to LBM and marked as a true positive if their IPs were also the same. This method was repeated for thresholds from 1ms to 350ms, during each iteration of threshold the tests were run a total of 200 times. For each set of tests the True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) were measured. These results were then used to calculate the accuracy and precision of LBM fingerprinting to re-identify a host in order to determine optimal values for a threshold variable. Figure 13 shows the re-identification of host A across multiple thresholds, while Figure 14 shows the same results for host B.

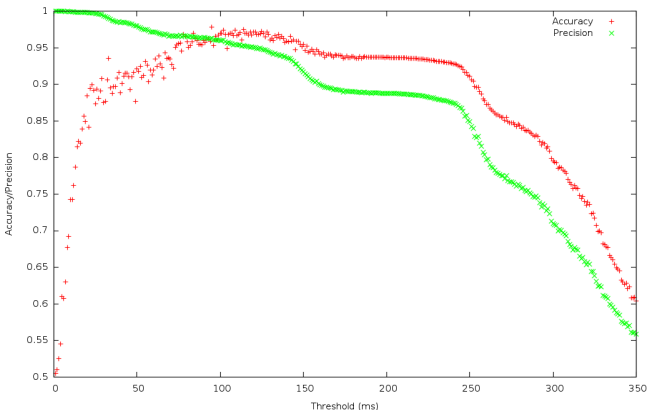

Figure 13. Accuracy and precision measurements for various threshold values for host A.

Figure 13 shows an accuracy and precision of 96% converging at the point were the threshold value was set to 94ms. While Figure 14 shows the same convergence at 95% with a threshold value of 109 ms. Ultimately for host re-identification a slightly higher threshold value should be used to insure a higher true positive rate as an increased false positive rate could be circumvented by progressive validation of other attributes of the observed host; such as open ports, ISP and service versions. This, however, does indicate the LBM fingerprinting for host re-identification should only be used as a supporting metric and not on its own.

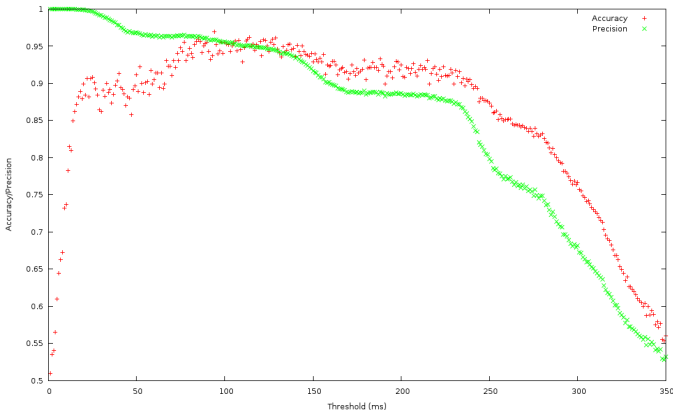

Figure 14. Accuracy and precision measurements for various threshold values for host B.

The accuracy of LBM fingerprinting could largely be effected by the geographic location of PingBase stations, if the stations are not sufficiently far enough from each other, similar latency measurements would exist. This would decrease the accuracy of re-identification and was observed by looking at all of the latency measurements across the three PingBases.

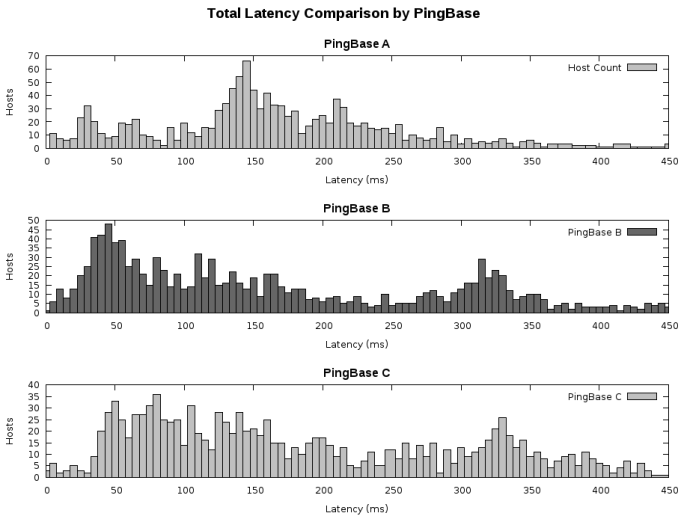

Figure 15. Latency comparisons between the three geographically separated PingBase stations.

Figure shows 15 histograms representing the measurements of LBM fingerprint latency of the three PingBase stations. It's clear that PingBase B and C have a similar distribution in measurements that is contrasting to PingBase A, this is likely as a result of the close geographic locality of A and B. Figure 16 confirms this assumption as PingBase B and C are much closer to each other, C situated in the Ukraine and B in the Czech Republic while A is located far from the other two in the United States of America. Ideally the PingBases should be located far from each other, such as one in North America, one in Europe and one in east or central Asia. However, as a proof of concept we believe these results prove the possible use of LBM fingerprinting for host re-identification.

Figure 16. Geographic location of the three ping bases.

V. Conclusion

Through the integration of the rDSN publishers with honeypots and network telescopes it was shown how real-time access to malicious data sensors could be provided. The utilisation of this access to data was illustrated through the integration of the rDSN receiver with the AR Framework and data was collected from malicious hosts. The data collected with through the integration of these tools provided, in real-time, situational awareness regarding nefarious hosts on the Internet. Giving researchers an insight into the characteristics of malicious hosts such as more accurate OS identification through Nmap scans. Further more, open ports on malicious hosts were determined, allowing for the inference of services that run on these compromised hosts. This research also showed how latency multilateration could be used as an supporting metric for host fingerprinting, with a high level of accuracy and also determined the optimal values for the threshold variable during LBM comparisons. It is hoped that through the deployment of applications introduced in this paper, an increased level of situational awareness could be generated to provide insight into the climate of malicious hosts on the Internet. Finally it is hoped that this data could support the development of defensive measures to protect future networks.

References

[1] T. Bass, "Multisensor data fusion for next generation distributed intrusion detection systems," in *In Proceedings of the IRIS National Symposium on Sensor and Data Fusion*, 1999, pp. 24–27.

[2] S. O. Hunter, E. Stalmans, B. Irwin, and J. Richter, "Remote fingerprinting and multisensor data fusion." in *ISSA*, H. S. Venter, M. Loock, and M. Coetzee, Eds. IEEE, 2012, pp. 1–8. [Online]. Available: http://dblp.uni-trier.de/db/conf/issa/issa2012.html#HunterSIR12

[3] M. Bailey, E. Cooke, F. Jahanian, J. Nazario, and D. Watson, "The internet motion sensor: A distributed blackhole monitoring system," in *In Proceedings of Network and Distributed System Security Symposium (NDSS âĂŹ05*, 2005, pp. 167–179.

[4] D. Moore, C. Shannon, D. J. Brown, G. M. Voelker, and S. Savage, "Inferring internet denial-of-service activity," *ACM Trans. Comput. Syst.*, vol. 24, no. 2, pp. 115–139, May 2006. [Online]. Available: http://doi.acm.org/10.1145/1132026.1132027

[5] U. Harder, M. W. Johnson, J. T. Bradley, and W. J. Knottenbelt, "Observing internet worm and virus attacks with a small network telescope," 2005.

[6] M. Bailey, E. Cooke, F. Jahanian, A. Myrick, and S. Sinha, "Practical darknet measurement."

[7] E. Aben. (2009) Conficker/conficker/downadup as seen from the ucsd network telescope. Http://www.caida.org/research/security/ms08-067/conficker.xml. [Online]. Available: http://www.caida.org/research/security/ms08-067/conficker.xml

[8] C. Shannon and D. Moore, "The spread of the witty worm," *IEEE Security and Privacy*, vol. 2, no. 4, pp. 46–50, Jul. 2004. [Online]. Available: http://dx.doi.org/10.1109/MSP.2004.59

[9] R. J. Barnett and B. Irwin, "Towards a taxonomy of network scanning techniques," in *Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology*, ser. SAICSIT '08. New York, NY, USA: ACM, 2008, pp. 1–7. [Online]. Available: http://doi.acm.org/10.1145/1456659.1456660

[10] U. Harder, M. W. Johnson, J. T. Bradley, and W. J. Knottenbelt, "Observing internet worm and virus attacks with a small network telescope," *Electron. Notes Theor. Comput. Sci.*, vol. 151, no. 3, pp. 47–59, Jun. 2006. [Online]. Available: http://dx.doi.org/10.1016/j.entcs.2006.03.011

[11] Y. Shinoda, K. Ikai, and M. Itoh, "Vulnerabilities of passive internet threat monitors," in *Proceedings of the 14th conference on USENIX Security Symposium - Volume 14*, ser. SSYM'05. Berkeley, CA, USA: USENIX Association, 2005, pp. 14–14. [Online]. Available: http://dl.acm.org/citation.cfm?id=1251398.1251412

[12] S. O. Hunter, E. Stalmans, B. Irwin, and J. Richter, "An exploratory framework for non-aggressive response to hostile network traffic," *Uses in Warfare and the Safeguarding of Peace 2012 (IWSP 2012)*, 2012.

[13] S. O. Hunter and B. Irwin, "Tartarus: A honeypot based malware tracking and mitigation framework." in *ISSA*, 2011.

[14] N. Provos and T. Holz, *Virtual honeypots: from botnet tracking to intrusion detection*, 1st ed. Addison-Wesley Professional, 2007.

[15] "Nepenthes - detecting malware," http://nepenthes.carnivore.it (last visited June 2010).

[16] C. van der Walt, "When the tables turn," in *Black Hat Asia*, 2004.

[17] Sans: Malware faq: Ms-sql slammer. [Online]. Available: http://www.sans.org/security-resources/malwarefaq/ms-sql-exploit.php

[18] T. Bass, "Intrusion detection systems & multisensor data fusion: Creating cyberspace situational awareness," 2000.