Detecting Pandemic and Endemic Incidents through Network Telescopes: Security Analysis

Author: Fotis Gagadis
Supervisor: Dr. Stephen Wolthusen

Technical Report

RHUL-MA-2008-12

22 January 2008

Department of Mathematics
Royal Holloway, University of London
Egham, Surrey TW20 0EX, England
http://www.rhul.ac.uk/mathematics/techreports

Abstract

Moore *et al.*, from the *C*ooperative *A*ssociation for *I*nternet *D*ata *A*nalysis (CAIDA), proposed in recent years another measurement and monitoring method for the network and Internet. Network Telescopes are used to detect malicious traffic events generated from Denial of Service attacks, worm infected hosts and misconfiguration. This report is focused on endemic and pandemic incidents (DoS, Worm) and how these incidents observed through different Darknet topologies and statistical models. Furthermore, network telescopes effectiveness will be examined for broader understanding and evaluation.

Acknowledgments

The MSc Thesis is dedicated to my mother, who without her help, support and great efforts I would not succeed up till now.

Moreover, I would like to thank one more person who is very important to my life (right now at least :-( ), Haru Yoshimoto. Thank you very much for your kindness, support, love, affection and efforts all these months. I hope your dreams become true. This project would not become true without you. Thank you.

Furthermore, I would like to thank all my friends and relatives for their support Dimitris Koukas (or else the Boss), Spiros Pappas, Kostas Ntasiotis (or else Remali), Gaven Watson (the Dr.), Chris McLaughlin (thanks for everything mate), Akis Telis (my cousin), Alex Makropoulos, Eftichis Tsikouras, Panos Elliopoulos and to all the friends that I forgot(sorry is 5 a.m. and my brain is not working at all). Special thanks to all my tutors, Dr. Alex Dent(A valuable Advisor and Professor) and my supervisor Dr Stephen Wolthusen(Thank you very much for everything! For the guidance, advises and time.) at Royal Holloway, because they press my limits and I learn a lot this year.

And special thanks to all of you who supported me and I forgot.

"There is a thin line among reality and dream. The man who succeeds to pass this line, will make his dreams come true and his life like a dream"

Preface

For the purpose of this report network telescopes will be examined and how these systems will analyze, characterize and categorize pandemic and endemic incidents. There will be no further analysis on pandemic and endemic incidents, from the point of view on how they work, how generate attacks or even the underlying mechanisms of these potential threats. Moreover, various researchers opinions will be analyzed and how researchers observed and monitored these attacks and only from this point of view. Additionally, this report will evaluate topologies and architectures of network telescopes. Furthermore, at the end of this report, the reader will examine the effectiveness of network telescopes.

This report is conducted into six different chapters with different topics and continuity. The first chapter is a general introduction for the topics covered in the report. Also, there will be an extended analysis of what is a network telescope. Furthermore, familiarization with few of the topics will be presented to the reader. On Chapter two the concept of IP ranges and CIDR will be covered. There will be notation of prefixes for convenience. Furthermore, Ch2 will examine the need for large fractions of IP addresses which a network telescope must possess to operate correctly. Moreover, the concepts of targets and events will be presented. At the end of Ch2, the reader will be able to examine the single packet algorithms and how network telescopes characterize the times and duration attacks.

The third chapter, examines the pandemic and endemic incidents observed by a network telescope. As it was mentioned before, these phenomena will not be analyzed by itself, but from the point of view of the researchers. Furthermore, the reader will understand backscatter analysis and the propagation monitoring tools network telescopes make use of. Ch4 examines different topologies of network telescopes. There will be an analysis on passive, active, distributed, anycast, transit, honeyframs or hybrid systems and greynets which are mixtures of dark IPs and active ones. Ch5 is the evaluation and analysis of network telescopes. This chapter must be considered carefully because of the contents. Actually, this chapter refers to the advantages and disadvantages of network telescopes as systems. If it is read carefully, the reader can understand that this chapter is not only examines the deficiencies of telescopes, but also examines the concerns of the researchers and how these problems can be solved. The last chapter examines the observations resulted from the present research and proposals for problematic characteristics of network telescopes.

In conclusion, it must be referred that if the reader needs further information while examines this reports, before the bibliography section, can find glossary and abbreviations. Glossary and abbreviations are combined together, in order the reader to have the opportunity to comprehend and review the various ideas. Furthermore, it must be mentioned that in certain chapters the concepts are not fully developed. This is because of the lack of resources found from academia and educational networks. Unfortunately, network telescopes are new concepts and are not fully developed, concluding from the research done so far. Additionally, the bibliography at the end will help the reader to extend his/her knowledge on this particular subject with further information.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Why we analyze Network Telescopes

Network telescopes ,in recent years, have emerged as a solution for monitoring misconfiguration, pandemic and endemic incidents. Its usefulness extends to academia, enterprises,organizations and businesses for experimental reasons, until recently. CAIDA organization with the help of institutions, organizations and enterprises created a distributed network telescope the IMS(Internet Monitor Sensor). Its ability and effectiveness observed with the extended monitoring of pandemic and endemic incidents. Pandemic and endemic incidents are DoS and Internet Worms respectively.

Network telescopes have many features which create efficient systems. Moreover, the architecture of network telescopes and how this traffic is monitored and characterized is very interesting. Specialized algorithms, health community mathematics, sensors, Virtual Machines and more features describing an innovative system. Moreover, topological models are completely different and designed to offer efficiency in a network. Furthermore, network telescopes instead of monitoring assigned addresses such as Intrusion Detection/Prevention Systems, telescopes monitor unused resources. The use of dark(unused) addresses help a network telescope to monitor efficiently, because traffic in unused addresses means misconfiguration, unsolicited traffic or propagation of malware.

In conclusion, there will be an extended analysis of network telescopes through the chapters that help the reader comprehend and understand various features and characteristics of network telescopes. In case that the reader wants to extend his/her research, at the bibliography section can find an extended research of technical reports, in proceedings, proceedings, in books and many more references. For further research it is recommended for the reader to search keywords such as “network telescopes”.

1.2 What is a Network Telescope

At the end of previous millennium, beginning of the new era, Moore *et al.* proposed Network Telescopes as an alternative to network/security monitoring and measurement systems. Network Telescopes ,today, can be used mostly for academic purposes and gathering data as distributed topologies for Internet observance; analysis of this subject is not of our concern at the moment. Our foremost priority is to explain what a network telescope is.

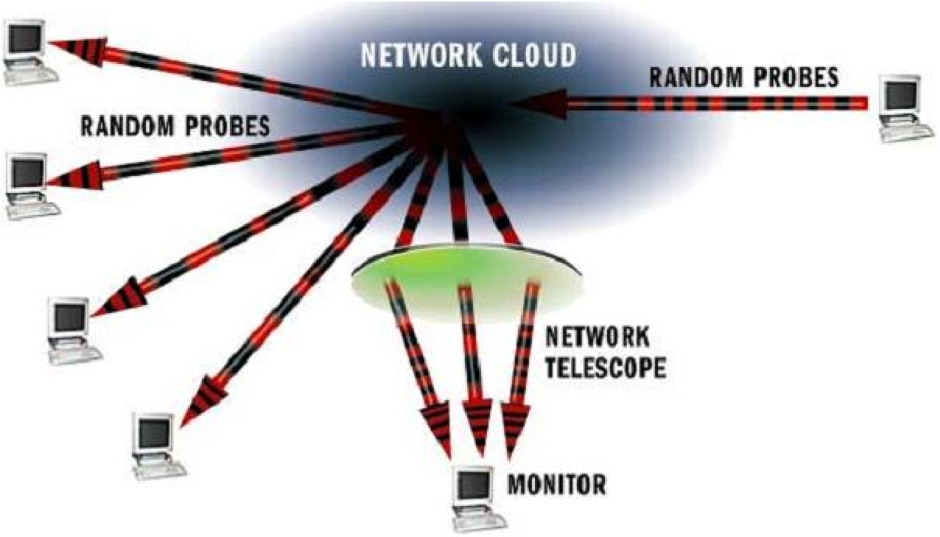


Figure 1.1: Network Telescope[56]

A Network Telescope considered to be a portion of routed IP addresses, with no or little legitimate traffic exists. Monitors remote security events which are the results of flooding DoS attacks, misconfiguration, type of scanning and worm propagation [64]. Therefore ,the observed traffic in an unused address space must be the result of misconfiguration, worms scanning or backscatter traffic from spoofed addresses since there are no hosts or devices[52].

Otherwise, a telescope can be considered as a tool or Darknet/BlackHole/Internet Sink/Telescope [2] which observes specific remote security events such as DoS/DDoS attacks, worm scanning from infected hosts, port/host scanning and misconfiguration[59]. Furthermore, if a host sends packets to randomly selected IP addresses there would be a probability to observe these packets , if the address space monitored is enough for safe conclusions resulting from statistical methodologies and the amount of incidents observed [56]. Therefore, analyzing and disassociating the unwanted traffic could be manageable[2]. Additionally, the prefix of the subnetwork observed characterizes the Network Telescope and determines the amount of data collected. The majority of Network Telescopes passively monitor the traffic, but there are also advanced telescopes that could perform active monitoring[31]. On the other hand, a telescope could be used to monitor a section of Internet space and measure the traffic of a wide range of IP addresses for backscatter activity, worm scanning and host scanning[69].

For further understanding, analyzing figure 1.1 could be beneficial. The infected host, in our figure, sends packets to all the hosts and IP addresses in our network, even to addresses that no traffic exists. Therefore, the network telescope receives the packets sent and by measurement techniques can analyze the purpose of this activity.

1.3 Endemic and Pandemic Incidents

Endemic and Pandemic incidents are the (Distributed)Denial-of-Service and Internet Worms, respectively. In DoS attacks the attacker spoofs IP addresses randomly and floods the targets with requests. The target responds believing that the sender is a legitimate user[59]. On the other hand, worm is a self-replicating/propagating program which exploits vulnerabilities without human intervention in order to infect hosts. Hence, the infectees will continue the propagation and the spread of the worm[56].

These incidents are fully described with the monitoring capabilities of a network telescope. The characterization and categorization of these incidents will be analyzed basis to the efficient mathematical tools and algorithms operating at the network telescope. Furthermore, at Ch 3 will be analyzed the different cases of worm attacks and how pandemic/endemic incidents observed from Internet Monitor Sensor or other telescope implementations. CodeRed, for instance, was analyzed and observed throughout its propagation life cycle. Additionally, in the case of endemic incidents it was observed that most of the attacks were from competitors or for personal reasons.

1.4 Measurement Tools

Network telescopes use complex mathematical algorithms and tools. Usually, use statistical methodologies for measuring the start/end and duration times of the attacks. Moore *et al.* in their report for network telescopes fully analyzed the models of precision times and rates for event characterization. In the case of backscatter traffic , for instance, use the backscatter hypothesis to characterize and categorize the packets observed.

On the other hand, computer scientists borrow the epidemic models of the health community in order to characterize and categorize pandemic incidents. Through the SI and AAWP epidemic models network telescopes analyze and characterize the traffic flow observed from various incidents. Furthermore, through the algorithmic models used network telescopes can categorize the general traffic observed.

1.5 Detection, Deception and Prevention Topologies

The topologies of network telescopes, could be categorized into two. A passive network telescope records the observed packets and takes no further actions. By this interaction, network telescope has the opportunity to observe hosts and packet information. However, information about attack or misconfiguration might not be revealed. For instance, if a network telescope observe the TCP handshake will not monitor it. It will monitor it as attempt for connection with TCP[52].

On the other hand, an active telescope responds to incoming packets and tries to establish communication with the attacker. In the case of an Internet worm a network telescope will continuously communicate with more than 10 messages, sometimes, until the worm is identified. Active monitoring system can reliably distinguish attacks, emulate a service and analyze attacks. Furthermore, it can keep tracks of the attacker and offer scalability[34].

1.6 Evaluation, Analysis and Quality of Security

The evaluation, analysis and quality of security on network telescopes is an extensive matter that must be covered. Unfortunately, even if network telescopes have extensive use of mathematics, its implementations are difficult. There are many problems that must be solved, especially with sensors' topologies and how these sensors can monitor efficiently. Furthermore, it was observed that honeyfarms have extended features which must be implemented with extreme care.

Moreover, it was observed that network telescopes are effective in seeing large explosions of events and its effectiveness depends on proper statistical and mathematical tools[56]. On the other hand, Bailey *et al.*, observed that small darknets ,sometimes, can receive more packets/day than monitors which observe large IP ranges[52]. Moore *et al.*, added that a telescope's size is proportional to the size of the space monitored. Therefore, short and low intensity attacks generate less packets and thus the need for large monitoring space is required to resolve activity's information[61].

1.7 Outline of the following Chapters

Chapter 2 focus on the IP ranges and the need of telescopes to monitor large fractions of addresses. Furthermore, mathematical tools and how network telescopes use these algorithms for efficient traffic characterization will be examined. Moreover, on chapter 3 there will be an analysis of pandemic and endemic incidents, and how network telescopes observe and categorize these phenomena. Additionally, on chapter 3 mathematical tools, for instance backscatter analysis and worm propagation models, will be examined.

On chapter 4 there will be an analysis and description of various topologies. Furthermore, chapter 5 will evaluate the effectiveness of network telescopes and various problems related to its implementations. The last part of the research is the conclusion. Conclusion, will focus on summarizing important parts of the research and various observations for effective monitoring will be mentioned.

Chapter 2

Internet Protocol Monitoring

2.1 Introduction

On this chapter, analysis of network telescope's basic concepts will be conducted. There will be an examination of Internet Protocol ranges and the monitoring techniques used for observing endemic and pandemic phenomena. Also, analysis of single packet observations will be examined. How we can characterize packets and how to use these data. Furthermore, packets usually arrive from IP ranges monitored by a network telescope. Therefore, analysis of observation techniques from various researchers will be examined.

2.2 Internet Protocol ranges

IP is one of the most important protocols from the TCP/IP suite. IPv4 addresses are logical addresses consisted of 32 bits long and have 256 possible combinations, but 0 represents the local address and 255 the broadcasts. Therefore, the real ranges of IP are from 1 to 254 for network hosts. A part of the address is assigned to the network and part to the host. For instance, 172.24.206.18 IP address with 255.255.0.0 subnet has network 172.24.0.0 and host ID 206.18[85]. Unfortunately, the purpose of this section is not to introduce IP addressing but to briefly describe classes and prefixes of the networks, in order the reader to understand network telescopes. Whoever desires to comprehend IP in depth, there are a lot of books and Internet sites explaining the subject.

The IP classes are five in total. Class A is intended for large number of hosts and especially large corporations. Furthermore, class A has a range of 1 to 126 and allows 16,777,214 hosts per network. Class B, has range from 128-191 and assigns 65,534 hosts per network. Class C has a range of 192 to 223 and allows 254 hosts per network. Class D, is used for multicasting purposes and has a range of 224-239. Class E, has a range of 240-255 and is used for experimental reasons mostly*(**Note:**The class system is old. It is* *mentioned only for the purposes of efficient reading. Today, the networking community uses the Classless InterDomain Routing(CIDR) and notations like /x, described in the next paragraph)*[85].

Therefore, the resulting addresses are 2^{32}. Hence, a /8 (Class A) network has range of 2^{24} addresses which have in common the first 8 bits. The /16 (Class B) network, has range of 2^{16} addresses which have in common the first 16 bits. Moreover, a /24 (Class C) network has 2^{8} addresses which have in common the first 24 bits. Consequently, a /32 address is a unique IP address. The notation / will be used later with measurement and monitoring methods[64].

2.3 Why we need large fraction of IP addresses

Having a large amount of statistical data observers and scientists can conclude to safe results with higher probabilities. By asking an astrophysics professor what is the probability of seeing a rare phenomenon in the galaxy, the answer will be certain and precise because of statistical data gathered through eons. Therefore, a scientist needs data in order to conclude to a theory, and a theory that can be supported 100%. Also, we as human beings in our every day life take decisions through data gathered all these years. For instance, if a child would be asked : "If you place your hand to a heated metal, what will happen?", the answer is obvious because of previous knowledge and data gathered in early years. The same with a network telescope, gathers data for measurement, monitoring, categorization of the phenomena. Hence, if the network telescope have available a large amount of data there could be higher probability to detect and analyze certain phenomena.

A Network Telescope derives from the same idea as astronomical telescopes. Having a large address space of photons arriving at the telescope, there could be a higher probability for the telescope to observe more phenomena. Through this analogy, a network telescope observing a large IP address space has greater probabilities and more data for further analysis. While observing a large fraction of IP addresses, the ability of the telescope monitoring host behavior and categorize the features of this activity with start, end, intensity times and characterize the phenomena is becoming higher. Clarifying this phenomenon a further example will be given. Having an IPv4 network with size of /8 , there is a p probability monitoring a target. Consequently, the probability of monitoring a /8 network will be $p_8 = \frac{1}{2^8} = \frac{1}{256}$[64].

Depending on the range of IP addresses observed the amount of traffic and data could be quiet large[52], but generally a network telescope is able examining explosions of large events in a network, not small events, through statistical methods[59].

Furthermore, telescopes with a broader IP address monitoring can observe events generating fewer packets, because of short duration or low sending rates. But, the accuracy of observing start and end times of a phenomenon in a large network are higher[59]. Additionally, as it is described in figure 2.1, it can be comprehended that broader

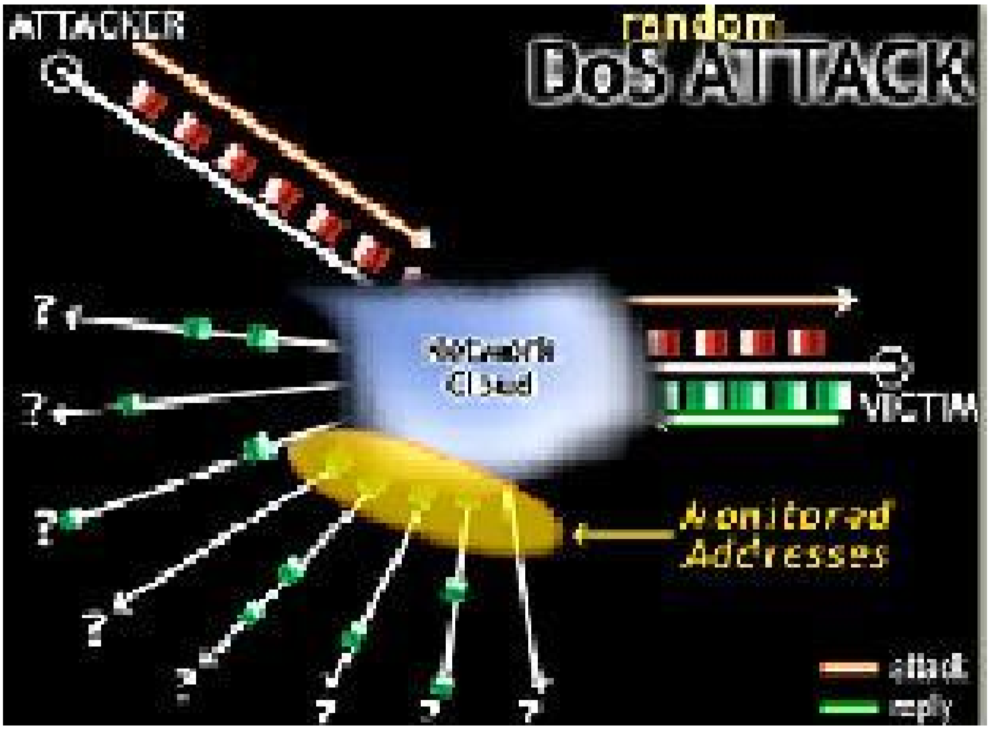


Figure 2.1: Monitored Addresses[59]

ranges of IP result on better conclusions through statistical models. Therefore, if a DoS attack floods a /8 network it is possible to detect the phenomenon in less hour given than in small telescopes with less ranges. For this purpose, further analysis will be given in section 2.5 of chapter 2.

2.4 Targets and Events in a network

On security community, the basis on characterizing an attack is to examine the destinations or the destination, the rate of the attack (e.g high worm propagation rate) and examine various approaches for categorizing a certain phenomenon. For instance, characterize an incident such as DoS attack with bots, a worm activity, the targeting rates, the propagation or how many hosts infected this certain attack.

A fundamental theory in network telescopes is the idea of target and event. Furthermore, a host in a network could be a device plugged and operating automatically or under the guidance of an operator. This device in order to communicate through the network sends packets received by other devices. Sometimes the traffic generated could be random and unbiased. On the other hand, because of an infectee the unsolicited traffic generated could be the result of a Denial-of-Service attack or worm infection. Moreover, a host selecting a destination or a *target* to send unsolicited traffic results in a *targeting rate*. Considering that a host selects targets at a given targeting rate results in an *event*[64].

Additionally, events detected by a network telescope could be divided into classes and effect on a network, according to Bailley *et al.* When an attack or an increasing probing occurs will have an impact on a very small amount of IP addresses locally or globally over the Internet, but this classification of the incident applies only to the target of this event[52].

2.5 Telescope Monitoring

Observations In section 2.3, it was analyzed why a large fraction of IP addresses needed. It was explained that a large fraction needed in order to have accurate statistical data and analyze the event in a network with higher probability and less time than in a smaller one. This section is dedicated to the methods which a network telescope make use of and these methods are analyzed in depth. Furthermore, analysis of backscatter traffic and worm propagation activity -as it is monitored by telescopes- will be given in chapter 3.

As it was mentioned in section 2.3, the probability of monitoring a target and an event is needed in order to explain how a network telescope observes. Notation / is needed and network telescopes use it in order to quickly find the probability of the telescope monitoring a target chosen by a certain host. Therefore, the probability p is given by a ratio of the address space monitored and the total available address space. Consequently, for a network size /x, the p monitoring a target is : $p_x = \frac{1}{2^x}$. Having mentioned before as an example, the probability monitoring a /8 network will be $p_8 = \frac{1}{2^8} = \frac{1}{256}$[64] presenting the accuracy and the efficiency of a telescope seen events at /8 network.

Analyzing the ability of network telescopes observing remote events it would be useful to apprehend the duration of the events for a /x size telescope, because many phenomena have targeting rates that are either fixed or relatively constrained. Phenomena such as DoS or worm infections, the victim's rate of response is either limited or increased depending on the capabilities of the host and the network. Therefore, the phenomena recorded must be treated as measurable quantities with targeting rates and time duration[64]. Before analyzing the original idea of Moore *et al.*, analysis on various researchers' opinions will be examined for broader understanding.

Bailey *et al.*, implementing the idea of telescopes observed that the router or the dynamic host server must forward packets to the monitor, since the monitor observes the unused addresses[52]. Furthermore, Joel Sandin added that when a sensor observes in real deployments 2^{20} unused addresses could have good results, but by avoiding the cost we loose in likelihood observing fault results[75]. On class A network, for instance, the telescope monitoring results in $1/256_{th}$ victim responses of spoofed addresses [59]. However, the never ending activity is an unproductive activity. Hence, two are the most useful types of background radiation, backscatter traffic from DoS attack and Worm activity arriving at a network telescope[34].

Yegneswaran *et al.*, proposed that in order the monitoring effectiveness to be highly appreciated both unused and used addresses must be monitored. From one point of view, packets arriving from an unused space could be the result of dropped packets by a network's gateway or border router. However, unused spaces offers advantages such as observing misconfiguration packets arriving and malicious activity, thus false positives which is a problem of Network Intrusion Detection Systems can be minimized. Additionally, active responses from monitoring tool such as network telescope can be used to detect precise attack information, but classes like A and B must be used to offer better results[93].

However ,the idea using Greynets, as proposed by Harrop *et al.*, which are a mix of unused and used addresses has different measurement techniques and topological observations. Therefore, definition of the potentials is needed which are the listeners and the distribution of listeners. P is the set of IP addresses monitored. P_m, is the set of potentials having m IP addresses and P_m is the subset of greynet topology. L is the set of unused addresses observed and L_n represents n listening hosts. Therefore, $L_n \Leftarrow P_m$, with $n \Leftarrow m, L_n$ being a subset of P_m contained within P_m. Hence, greynet will observe packets arriving from members of Ln, with $L_n << P_m$. Additionally, the L_n members will be distributed all over P. Having needed to introduce style X across space P, because of n addresses spaced around the circumference of P, L_nX is the result. Therefore, L_nX will have a rotational orientation relative to P, named θ. Consequently, the set of listening hosts from P is described by coordinates like (L_nX, θ)[38].

On the other hand, monitoring local addresses there is a possibility the traffic to be blocked by constraints, because companies will set filters for incoming traffic. Therefore, Cooke *et al.* proposed honeynet sensors as an alternative. Honeynet sensors can be deployed in many different parts of a network near systems and critically assets of the organization. Therefore, there is a need to examine "how big a network telescope must be?". The answer was given by Moore *et al.*, who will be analyzed later. Furthermore, evaluating the observations of Cooke, figure 2.2 describes the required time observing packets from a random scanning in different ranges at 95% confidence. [19]

In order event monitoring to be more effective, a telescope can be combined with multiple sensors distributed across the network, because malicious events can be short lived. However, certain events such as worms can have a lifetime over an hour and can be observed effectively by distributed telescopes[5]. Cuiy *et al.* observed, in a /14 plus an additional /23 network, that malwares such as Blaster running on a host assigned private block address will propagate very rapidly to public addresses, sending a bigger amount of traffic to the telescope. Therefore, a magnified visibility of the telescope in such kind of malicious traffic can be succeeded. Also, they observed that Honeyfarms, which later will be analyzed by this report, despite monitoring a /23 network telescope can observe more radiation background than a /14 network[23].

However, there are imperfect methods which limit the ability of monitors to represent unique attack sources. IP addresses does not represent individuals and the mapping of these addresses is not static; depending on the network. Furthermore, the variability

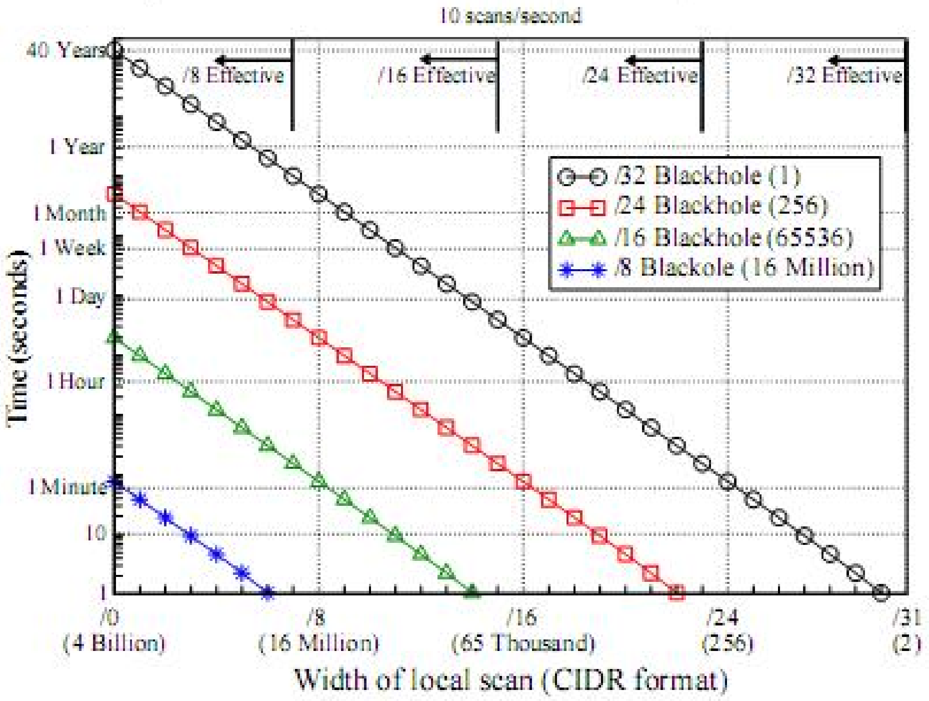


Figure 2.2: Time needed to observe with 95% confidence [19]

of the sources observed can have an effect on the monitor [5].

Figure 2.3, describes the distribution, that over 90% of packets monitored, at each sensor, less than 10% sent by IPs seen at the 14 telescopes [5].

Furthermore, the type of services in a network could limit the evaluation of an event. Therefore, the packets observed it would be efficient if the scanning types can be in a uniform matter. Additionally, as it was observed by researchers, a hybrid system or greynet can offer greater visibility and can help to a detailed analysis [5].

Chen *et al.*, on the other hand, show that using passive fingerprinting with monitor sensors can examine the characteristics of a target, such as the Operating System using. The results show that Microsoft Windows hosts are vulnerable and more frequently attacked by 91% and approximatelly 7% other Operating Systems Unix-like. Furthermore, researchers tried to recognize the top 10 targeted ports in their sensors. They observe, that ports like 137,135,139,445 are not only vulnerable to Windows systems. Also, while their networks were monitored, observe that well-known backdoors are still targeted, but the most targeted ports are for web services such as port 80(http)[72].

Pang *et al.*, on the other hand, use another approach to measure the activity of the traffic via network telescopes. They use two approaches, taming the traffic volume and building application-level responders. Through taming traffic volume can filter the traffic in order to balance between traffic reduction and the information lost[68].

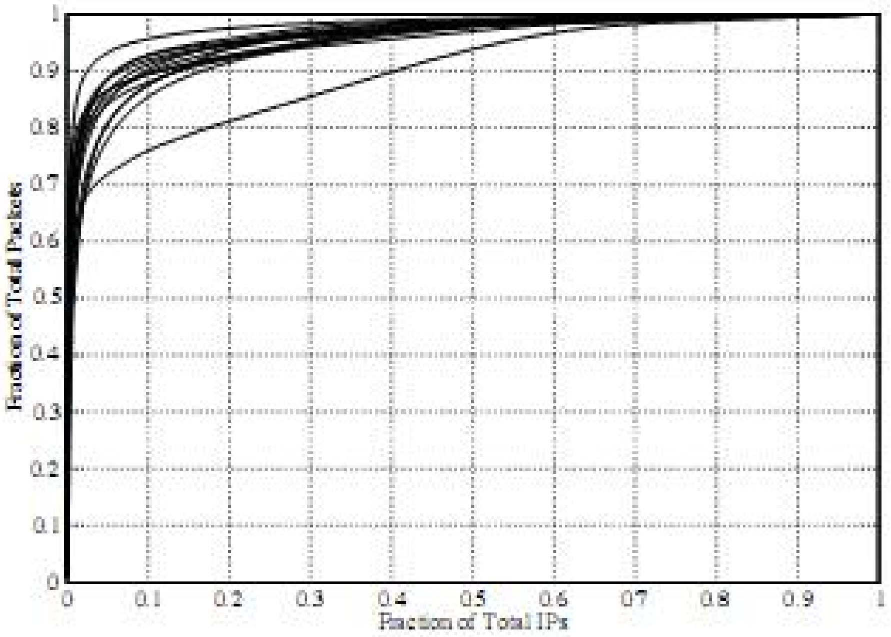


Figure 2.3: Contribution of total packets observed at 14 monitors. [5]

On the other hand, by keeping the connections initiated from each host and discard the remaining, can be an inconsistent viewing of the network because open connections create unreachable addresses. Similar to this measuring strategy, Pang *et al.* is that keeps port pair connection from each source, but also creates inconsistent view. Another strategy, is by keeping a type per source activity, but it is hard to be implemented. The final way is to choose, for experimental reasons, the IP addresses and assume that the affinity of background traffic monitored is the same like monitored addresses[68].

Another way of monitoring with a network telescope, proposed by Bailey *et al.* for the Internet Monitor Sensor, which can compute the MD5 checksum from the receiving packets payloads observed by the sensor and analyze them basis to the payloads monitored before. If the checksum or signature is already recorded, the passive monitor logs the signature without storing the payload. On the other hand, if a new signature is monitored, the payload and the signature observed is being stored in the database. This method, is extremely useful for the monitoring sensor, since a /8 telescope observing and storing payloads, can reach approximately over 100GB per day. Furthermore, the efficiency of checksumming, provides an advanced signature system that can be used for further measurement methods. By this way, the Internet Monitor Sensor, can differentiate the traffic and monitor with higher effectiveness worm events. Also can detect vulnerabilities in systems, like virus and worm backdoors. Additionally, observing DDoS events, is highly efficient even with attacks that generate less traffic in a wide range of IP addresses[4].

Measurement Methodologies Rajab *et al.*, to measure and sample the traffic takes advantage of the clustered nature of the IP space. Uses the depth-first strategy model and selects randomly from /8 prefixes, meaning TCP, ICMP packets or ACK(acknowledgment) packets from popular ports. Then, if a response arrives the /8 prefix marked as active and then send packets for the /16 within the /8 network. If there is no response, the prefix is considered inactive[74].

Then, Rajab *et al.* takes n samples needed to measure with high confidence. Therefore, $p_{l,g}$ is the probability exploring a host in g prefix. Given the n samples the α probability of accepting one response from a prefix g is:

$$\alpha = 1 - (1 - p_{l,g})^n \tag{2.1}$$

The necessity of contact with at least one host on a prefix with probability α and examine n samples is:

$$n = \frac{\log(1-\alpha)}{\log(1-p_{l,g})} \tag{2.2}$$

Ideally, to detect live /16 prefixes from a single active host n must be large enough, but for practicality reasons in order to detect /16 prefixes which entails active hosts the empty or sparsely populated prefixes must be excluded. By including live prefixes to empty, with host occupancy $(p_{l,g})$ below certain β threshold, calculation can be achieved by replacing $(p_{l,g})$ with β in equation 2.2. Then it can be noticed that the threshold increases and the samples number decreases[74].

But we have to evaluate β in order the sampling detection of live prefixes to contain the majority of live populations. By defining β as threshold of active host occupancy $(p_{l,g})$, the sampling process can detect active prefixes containing 99% of Internet active population. The distribution denoting the internet live population $P(g)$ residing in a /16 prefix is:

$$p_{l,g} = \frac{P(g) \cdot N}{2^{16}} \tag{2.3}$$

, with N to be the totality of Internet live hosts. The nominator is the expected number of hosts and the denominator is the size of prefix /16[74].

On the other hand, it can be shown base to [74] that $P(g)$ can also be estimated by Monte Carlo study, but considering the estimation of $P(g)$ using an active set of IP addresses, a small set, it can be obtained from various sources. Consequently, with the estimation of $p * (g)$ which is the marginal distribution of $P(g)$, which is the distribution of active hosts at /8. Then, deriving $p * (g)$ from the accumulated addresses, the learning set of /8 prefixes is the last step. Therefore, by collecting a small dataset of 20,000 active addresses the estimated distribution $p * (g)$ can have estimated error of $e = 4.3 \times 10{-}5$[74].

Another approach is given by Shakkottai *et al.* which is based on the approach using Peer-to-Peer network, to identify anomalies. In order to find an instance of worm behavior, researchers express that a worm propagates exponentially, considering monitoring M hosts. Hence, a particular infected host choosing one of the observing hosts, the probability is $\frac{M}{N}$. Furthermore, the hosts monitored by all infectees is $\frac{MI(t)}{N}$. Therefore, this is the rate of scanning monitored systems. After that it is needed to determine the number of scans applied to observing hosts, $\overline{M}(t)$ with time interval $[O, t]$: $\frac{d\overline{M}(t)}{dt} = \frac{MI(t)}{N}$[79].

Bo *et al.*, on the other hand, measures the data through network measurement and management tools like Simple Network Management Protocol. Defines that the connection grade in time interval contacting i host in n IP hosts is t. Therefore, categorizes host i having three states. Firstly the host sends packets. Secondly, by receiving the packets and lastly neither sending nor receiving. Furthermore, define binary time series as $\{W(t), t \geq 0\}.W(t) = 1$, meaning that a host sending at time t packets with $W(t) = 0$ when the host is not sending packets at time. Hence, $\{D(t, t_1, t_2), t_1 \leq t \leq t_2\}$ denotes hosts, which host i send packets from t_1 to t_2 and $Dst(t)$ be the destination of the IP address which host i sends packet at t. Then, using the connection degree:

$$C_t^{tT} = \int_{(t-1)T}^{tT} W(t)K(t)dt \tag{2.4}$$

, where C_t^{tT}being the i's host connection degree at t. $K(t)$ is the decision function with $K(t) = Dst(t) \oplus D(t, 0, t-1)$ and $\oplus$ between the variable and the vector. Supposed having a variable and vector *(*A), the definition using equation 2.4 is:

$$\alpha \oplus A = \begin{cases} 0 & \alpha \in A \\ 1 & \alpha \notin A \end{cases}$$

[11]

Another method of obtaining sampling is the *Sample and Hold*. Sample and Hold identifies flows larger than a specified threshold and it is based on random sampling and a table containing hashes to observe flow ID'S and byte counts. Arriving packets are sampled and maintained in the table. After that, packets belong to a flow are counted. This approach unfortunately results to both false positive and negatives, but its accuracy can be very high. Therefore, it is needed to accurately identifying the flows $T\%$ confiscate link's capacity. Oversampling factor O is selected for false negatives reduction. Hence, it is resulted in $HT_len = \frac{1}{T} * O$ location in the hash table. The sampling rate is set as HT_len/C, with C being the maximum capacity of transmission in a link over specified period[93].

Moreover, subnet selection is based on network and bandwidth constraints. If the mean is known and the variance of a traffic volume, then it is possible to divide the bandwidth by this and take the monitored subnets available. After selecting samples the detection ability problem is amongst the most serious, defining the accuracy of telescope. Considering an unbiased estimator $\widehat{\tau}$ of a total population of τ, the estimated $\widehat{\tau}$ variance is: $var(\widehat{\tau}) = N^2 \left[(\frac{N-n}{N})\frac{\sigma^2}{n} + (\frac{1-p}{p})\frac{\mu}{n} \right]$ where the total number of subnets is N, the sample is n, μ the median and σ^2 variance of the population, with p the probability of detecting a host[93].

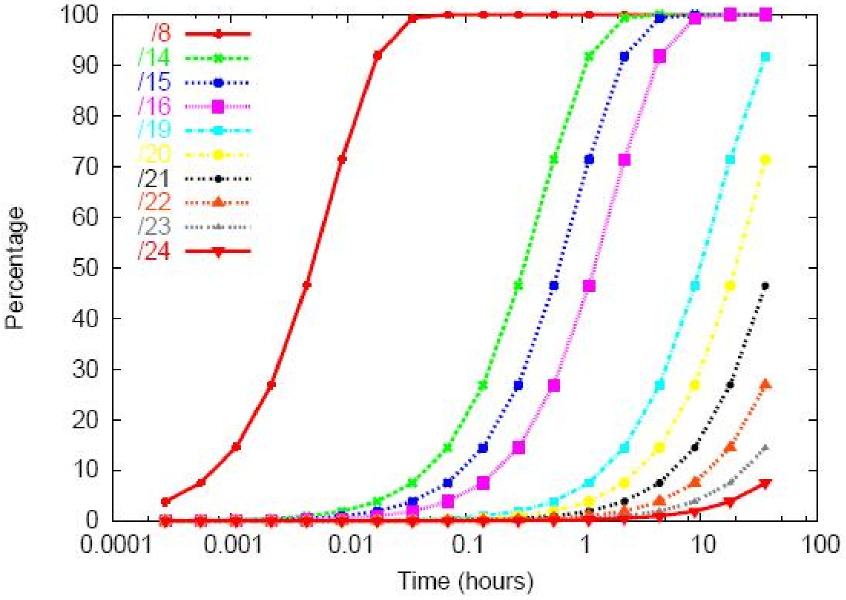


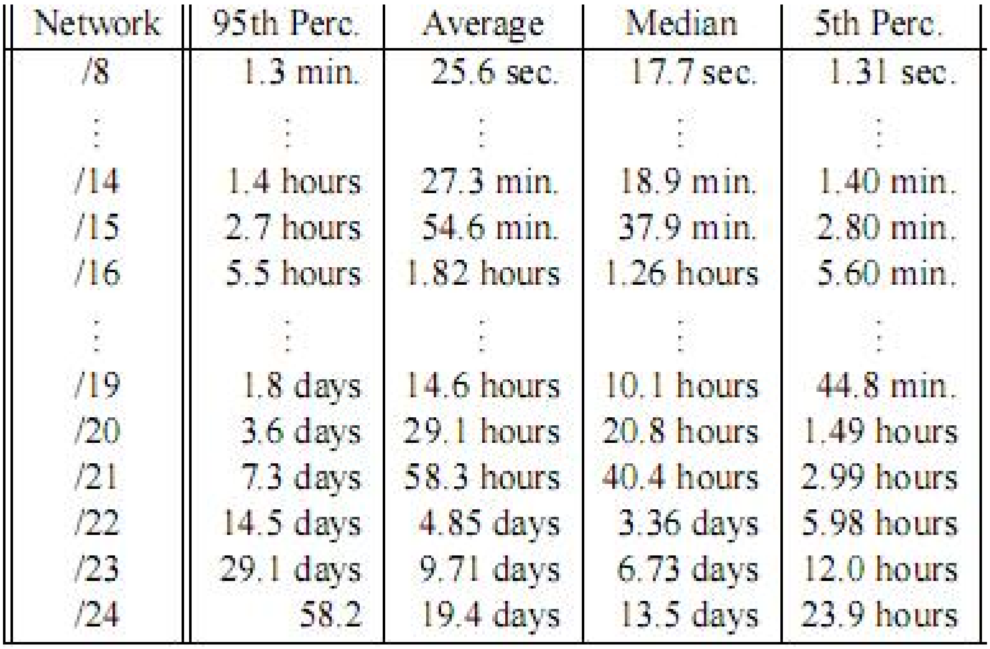
Figure 2.4: Probability of observing at least one packet[64]

Moore's law, lastly, is the most prevalent because of the modeling and the observations made. Moore, categorizes the ability of the telescope and analyzes from simple methods to complicated. Through Moore's work was explained how a telescope observes single packets, multiple and start/end precision times. Therefore, the probability of detecting a single packet when a host chooses IP addresses uniformly randomly is a geometric distribution. When a host sends numerous packets, the packets seen from the network telescope is a binomial distribution with p parameter. When a network telescope monitors a fraction of IP space, this fraction is mentioned as$p$. Assuming that target IP choices made by a host are unconstrained and having p odds of each packet send to the telescope is by definition a Bernouli trial[64].

Considering the packets generated by a host as the product of rate packets sent r, multiplied by the elapsed time T. The probability observing at least one packet in T sec_s is $P(t \leq T) = 1 - (1-p)^{rT}$[64].

As it can be observed from figure 2.4, the probability observing at least one packet from a host selecting random IP addresses at 10 probes per second ,on different telescopes, shows the likelihood of observing events[64].

Furthermore, T (elapsed time) before observing a packet of an event with Z probability is: $T = \frac{-1}{r \log_{\frac{1}{z}}(1-p)}$. Until the first packet observed the expected packets seen are $\mu_N = \frac{1}{p}$, with variance of $\sigma_N^2 = \frac{1-p}{p^2}$. Because of our interest on rates and time, the replacement of an absolute number of packets sent with rT and solve for elapsed time is: $\mu_T = \frac{1}{rp}$. Consequently, depending on the size of the telescope depends also the likelihood to



Network	95th Perc.	Average	Median	5th Perc.
/8	1.3 min.	25.6 sec.	17.7 sec.	1.31 sec.
⋮	⋮	⋮	⋮	⋮
/14	1.4 hours	27.3 min.	18.9 min.	1.40 min.
/15	2.7 hours	54.6 min.	37.9 min.	2.80 min.
/16	5.5 hours	1.82 hours	1.26 hours	5.60 min.
⋮	⋮	⋮	⋮	⋮
/19	1.8 days	14.6 hours	10.1 hours	44.8 min.
/20	3.6 days	29.1 hours	20.8 hours	1.49 hours
/21	7.3 days	58.3 hours	40.4 hours	2.99 hours
/22	14.5 days	4.85 days	3.36 days	5.98 hours
/23	29.1 days	9.71 days	6.73 days	12.0 hours
/24	58.2	19.4 days	13.5 days	23.9 hours

Table 2.1: Times on different /x sizes observe their first packet from a host choosing IP addresses at 10/sec IP addresses[64]

observe an event. For instance, observing a Code-Red-like infected host sending one or multiple packets on a /8 with 99.999% probability is 4.9 minutes[64].

Table 2.1 represents the standard deviation and median times, summarizing on average, in order to observe a packet from a host choosing random targets at 10/sec for different sizes of telescopes. If the table is noticed, it can be observed that monitoring an IP address at /32 the average of observing a host at 10 addresses/sec is over 13 years with 95% likelihood at 40 years. The 95% column represents the duration of an event on which a network telescope could monitor effectivelly 95% of the events. On the contrary, the 5% column shows that the telescope would miss 95% of the events[64].

From another point of view, monitoring two dissimilar IP addresses p_1, p_2 and comparing time T_1, T_2 given for detecting at least one packet P at rate r is:

$$1-(1-p_1)^{rT_1} = P = 1-(1-p_2)^{rT_2}$$

$$T_1 = T2\tfrac{\ln(1-p_2)}{\ln(1-p_1)}$$

Due to increasing mass in the distribution the detection would not scale linearly. Therefore, a /1 is better than a /2, while /2 takes 2.41 times to detect one or multiple packets at the same level of a /1[64].

Network telescope's ability to condense significant events from background traffic for multiple packets, is one of the utilities which the telescope has. Thus, our confidence of observing an event increases. Therefore, a telescope often must receive k or more

packets from an event in order to accurately classify them. Depending on the event monitored and the design, a threshold of k packets selected with a probability of monitoring k or multiple packets out of transmitted N packets is: $P(saw \geq k) = 1 - \sum_{y=0}^{k-1} \binom{N}{y} p^y (1-p)N - y$. Thus, if 100 packets from a DoS attack can be observed on a /8 telescope with 500pps last for 1 minute is:
$N = 1000pps \cdot 60sec = 30000packets$
$k = 100packets$
$p = 2^{-8}$
$P = 1 - \sum_{y=0}^{99} \binom{30000}{y} (2^{-8})^y (1-2^{-8})^{30000-y}$
$P = 95.2\%$
[64].

However, the importance to know start and end times of an event and duration is highly understood. The first packet observed is the upper limit of the start time and the lower limit is based on the last object observed. Therefore, if the start and end times are known, there is a higher likelihood for a telescope to examine an event and the time of the duration of the event. For instance, with a network telescope monitoring at a /8 there could be 99% confidence for an event with targeting rate at 10 addresses per second, that the event began two minutes before the first packet arrived. Hence, if the chosen IP ranges are independent and monitored by a telescope with p probability, the probability that N targets chosen outside telescope ranges even before the first observed destination is: $(1-p)^N$. Furthermore, the address choises are Bernouli trials, but if the network telescope observing a packet at a time, possible start times can be described by geometric distribution[64].

Additionally, estimating host's targeting rates from targets observed, obtaining certain boundaries could be useful. Therefore, rates can be determined exactly if the values of duration are known, the total number of targets and by having a binomial distribution with p parameter expressing the totality during an event with a host choosing N targets and the number of targets monitored. The expected monitored targets is: $\mu = Np$ with $\sigma^2 = Np(1-p)$ variance. Also, $p << 1, \sigma^2 \approx Np$ when N is large and the binomial distribution can be approximated by normal or Poisson distribution. Finding a distribution for the totality of targets N, observing n targets, an inverse problem is remaining. With $P[n|N]$ given by binomial distribution requires knowing $P[N]$ and $P[n]$, known as priori. Consequently, this allows the estimation of $\widehat{N} = \frac{n}{p}$ when N is large and the observed n is also large and the distribution is considered by normality for large network telescopes[64].

2.6 Summary

In this section, the usefulness of IP addresses and ranges for a network telescope were analyzed. There was an examination of why we need large fraction of IPs and how these ranges can be used for effective monitoring. Furthermore, there was an analysis on various techniques for packet characterization and observation. In conclusion, there was an examine of Moore *et al.* ideas who were the first implementing the idea of a network telescope.

Chapter 3

Endemic and Pandemic Incidents

3.1 Introduction

Endemic and Pandemic Incidents are the most important phenomena not only for observing them through a network telescope, but also as Internet phenomena that causes either overloading of network and resources or economical disasters. Endemic incidents are the DoS and DDoS phenomena caused by the attackers usually for vendetta reasons or to overload a network and hosts for economical reasons. On the other hand, Internet Worms (Pandemic incidents) are self propagating programs that cause overloads of networks or resources and usually their targets are governmental, military sites or database vulnerabilities. Worms such as CodeRed, Blaster and Witty, consider to be the most characteristic.

DoS attacks , usually, are generated from random sources. The packets observed are called "backscatter" packets and can be monitored across Internet, because of the uniformity which are characterized. In single targets, DoS can be observed as noise packets, but if collected in a properly installed network telescope, the backscatter and DoS attack can be clearly observed. On the other hand, worm attacks have similar patterns but to larger network spaces. Hence, both these attacks, DoS and Worms, can be observed either locally or globally through proper measurement and monitoring techniques such as network telescopes [65].

Thus, in this section, pandemic and endemic incidents will be analyzed through the view of telescope researchers. Analysis of DoS and DDoS attacks, and the backscatter packets observed by different sensors(blackholes) will be analyzed. Furthermore, Internet Worms and their behavior through the propagation face will be examined. In conclusion, there will be a further analysis on propagation models and how these models used for monitoring purposes.

3.2 Endemic Incidents

Denial of Service and Distributed Denial of Service

Denial-of-Service and Distributed Denial-of-Service, in the past years up to recently, are used as a tool to overload the network of a competitor or for vendetta reasons to overload network and resources of Internet sites such as yahoo.com. The attacker, usually, compromises a host and from this point starts sending packets to the target depending on the attack flooding technique used. In this section, the analysis of DoS and DDoS phenomenon from the point of view of the researchers will be examined. Not from the point of view of the researchers working on endemic incidents, but from the point of network telescope researchers. For instance, how the phenomenon is observed and how it was analyzed from a network telescope.

DoS attacks In DoS attacks the attacker spoofs IP addresses randomly and floods the targets with requests. The target responds believing that the sender is a legitimate user[59]. By spoofing techniques, attackers manage to conceal their identity in order to send their packets. Therefore, it seems that the packets arrive from more than one third parties. Furthermore, it was observed that the attacker can use the true IP address without spoofing a host[61].

Additionally, in DDoS or DoS attacks, the attacking hosts connects to the victim and appoints separate requests. Therefore, the target must maintain, through the processing power, all the connections created the overwhelm of the system. The target, also, tries to maintain the socket buffers of the TCP connections. Furthermore, this process creates CPU and network resources to be consumed[76]. On the other hand, Bailey *et al.* from the Internet Monitor Sensor research describes DoS as the denial of legitimate hosts to access resources and the attack has the ability either to crash the computing resource or to overwhelm every resource. DDoS attacks rely on this technique, but also consumes network resources and relies on large hosts assistant[4].

On figure 3.1, it can be observed that two large scale attacks against *www.sco.com* of the SCO Group on December 10,2003. The attacks use randomly spoofed hosts; Internet Motion Sensor was able to observe further backscatter data. There were classified 5 events, which 3 of them were against the web servers and two against the FTP and SMTP server. This figure shows clearly, if observed, the spectrum analysis as monitored by the Internet Monitor Sensor[4].

Furthermore, Moore *et al.* categorized DoS attacks into two categories. Those attacks which are flow-based and keyed on IP address and protocol, in which during the phenomenon, flow is defined with parameters like minutes, threshold and timeout. On the other hand, the event-basis is an event which can be observed for approximately a minute in a window, and there is no notion of the duration[56]. Additionally, one of the characteristics of DoS attack is that it consumes resources of the host or from the network, which serves legitimate purposes. Moore *et al.* characterized the DoS

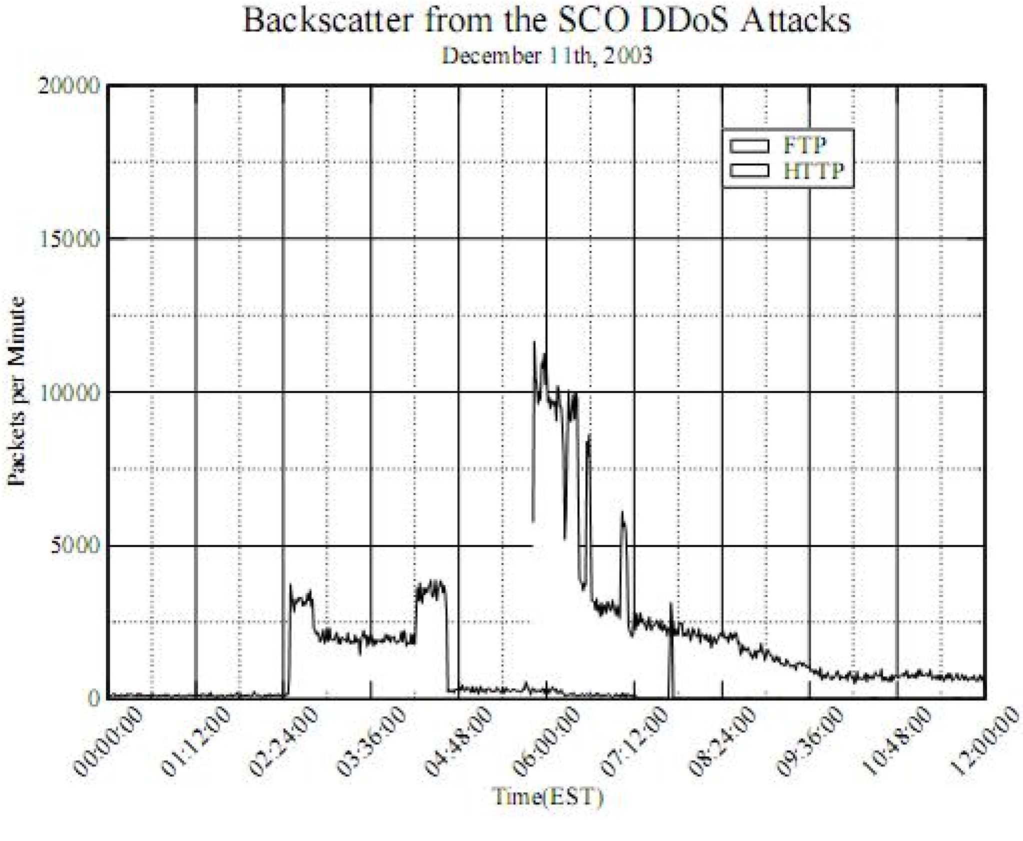


Figure 3.1: December 2003 www.sco.com attacks as monitored by IMS [4]

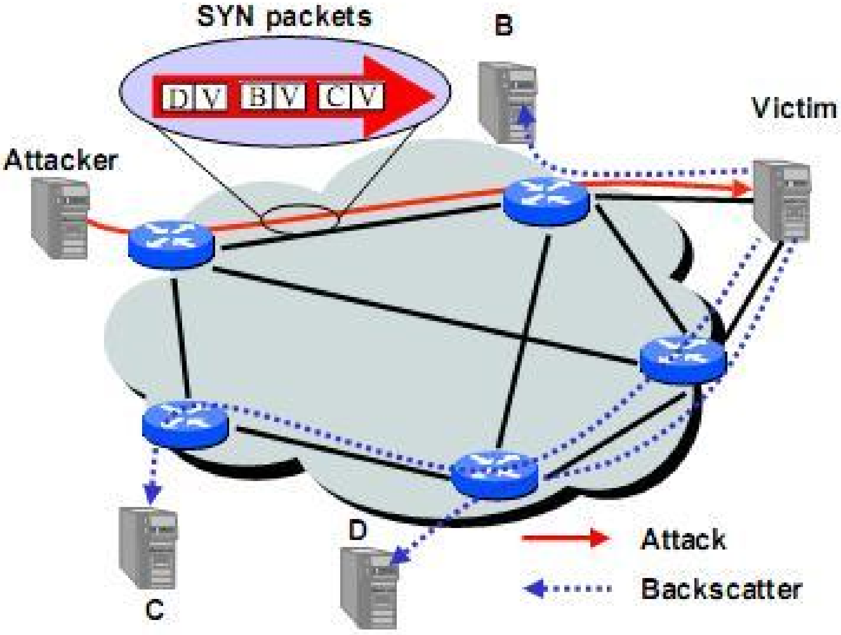


Figure 3.2: Backscatter Illustration [61]

attacks into logic and resource attacks. Logic attacks exploit existing vulnerabilities and cause servers either to crash or degrade their performance. Ping-of-death, is one kind of these attacks and can be avoided by upgrading software and sequences of packets. Resource attacks degrade the resources of the device or of the network by sending spurious requests. Defending against these kind of attacks, it can be difficult since distinguishing the requests can be complicated[61]. Furthermore, resource attacks has either as characteristic loading the network or impact on victim's CPU. Most of the attackers tries to overload the network by sending more packets to the devices more from which can process. At the same time, the attacker tries to overload the processing power of a device by requiring from the device to process more packets than can receive. One of the best known DoS attacks is the SYN flood. Generally DoS attacks try to find vulnerabilities and exploit them through a system. Today, sophisticated attacks focus mostly on infrastructure, backbone devices such as routers and database servers[61].

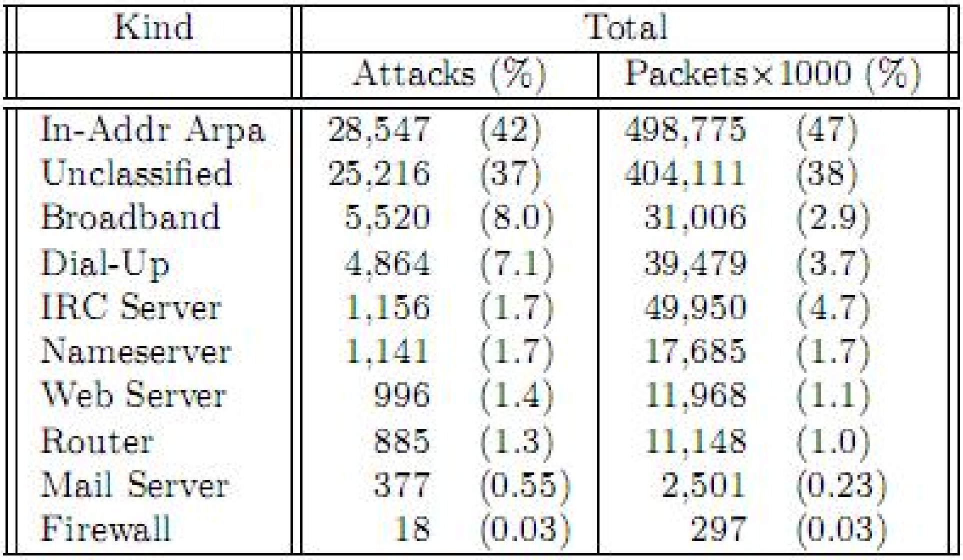
Usually, DoS attacks are randomly generated by programs which select addresses to spoof and send packets at random . The target having received the spoofed packet sends a response packet to the source claimed or to a network device which tries to reply, usually, with an ICMP message. At the end, these messages will be send to the randomly generated spoofed address or addresses. Furthermore, the source address be selected at random and the target's response will be distributed through a wide address space creating the "backscatter" effect[61].

In figure 3.2 a simple backscatter illustration is described when the attacker sends SYN packets toward the victim using spoofed addresses. While receiving the packets, the target tries to reply with SYN/ACK packets to the sources of the spoofed devices or hosts[61]. Furthermore, it was observed that 90-94% of the attacks are TCP and 43% approximately ICMP and the majority of the attacks are on multiple ports and few on services provided from HTTP and IRC. Additionally, flooding DoS attacks such as SYN and ICMP floods was examined that it can be succeeded through the spoofing of addresses randomly and through major attack tools. The targets respond, and then the usolicited responses or backscatter data accross the network or to internet space which can be monitored. The received backscatter is the event for a network telescope[56].

According to McPherson *et al.*, in their experiments with network telescopes, it was observed that attackers use allocated blocks and invalid addresses. Therefore, it might be extremely useful to recognize the ranges of spoofed addresses and install a filter to classify the source addresses. Furthermore, they select from a /32 network an address from a different range and advertise it to a network telescope. The workstation was monitoring and keeping logs through the use of a tool such as Arbor Network's Dark IP Application[53]. In addition, it was observed that event rates described the attack intensity, but not the attack duration. Therefore, measuring the duration attack through traces can be beneficial for a network telescope[61]. On the other hand, while the intensity of the attacks can be characterized, it is not easily sustained the ability to count the time of the attack. Moore *et al.*, found that most of the attacks has short duration time. Less than 10 minutes were 50% of the attacks and 80% less than 30 minutes. However, 90% even with less than an hour, the duration of the attack can be more than 5 hours and 10 hours or can be activated for multiple days[65].

Moreover, from a network telescope it was monitored that there is a significant percentage that shows attacks on dial up and broadband devices are larger than others. Attacks, especially on dial up hosts, can be rated with thousands of packets per second. Therefore, it could be concluded that these DoS attacks are for personal vendettas. Furthermore, more attacks can be observed to Internet Relay Chat users which can support multi-player games. 2-3% has been observed that attacks were directed on name servers and 1-3% to backbone routers. This can be really disturbing, since overwhelming a backbone router can cause permanent Denial of Service. Additionally, was observed that attacks are also directed to bellwether sites such as aol.com and amazon.com[65]. Moreover, the DNS lookup of the top-level domains show that over 10% targets were *com* and *net*, whereas 1.3-1.7% were *edu* and *org*. Consequently, com and net top-level domains are targeted mostly because of the commercial use. On the other hand, country domains also were targeted. For instance, Romania and Brazil, countries with limitation in networking infrastructure had frequently attacked in the past[61].

The classification of DNS name with target IP address can be observed in table 3.1. Unfortunately, the majority of the names could not be classified because the criteria do not match or the reverse mapping is not possible. In order to conclude and observe in better results of the source or targeting addresses, data from 2001 and 2002 are selected



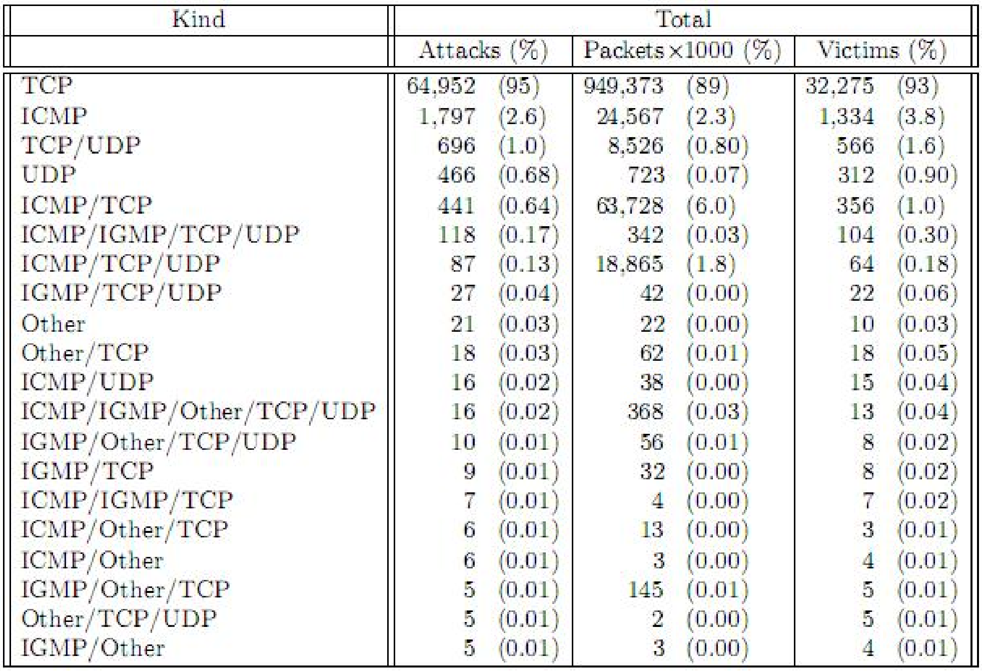
Kind	Total			
	Attacks (%)		Packets×1000 (%)	
In-Addr Arpa	28,547	(42)	498,775	(47)
Unclassified	25,216	(37)	404,111	(38)
Broadband	5,520	(8.0)	31,006	(2.9)
Dial-Up	4,864	(7.1)	39,479	(3.7)
IRC Server	1,156	(1.7)	49,950	(4.7)
Nameserver	1,141	(1.7)	17,685	(1.7)
Web Server	996	(1.4)	11,968	(1.1)
Router	885	(1.3)	11,148	(1.0)
Mail Server	377	(0.55)	2,501	(0.23)
Firewall	18	(0.03)	297	(0.03)

Table 3.1: Association of DNS name and target IP address[61]

and analysis of 100 target host names will be analyzed. Therefore, the categorization of target addresses can be realized. Furthermore, through this analysis it can be observed that approximately half of targets could be broadband users and 10% dial up. The 5-10% of targets were educational networks and a small percentage could be internet centers. The majority show that usual targets were either home users or small businesses. From this experiment, it can be observed that severe attacks are directed towards dial up or broadband users with thousands of packets per second. Actually, DoS attacks were directed for personal vendettas, for instance to users running IRC, multi-player games or even to sites needed parental advisory. Additionally, reverse DNS mappings were compromised as it was observed. Furthermore, network infrastructures can be attacked. Routers and name servers were attacked with rates 1.3% and 1.7% and compromised with a number of packets greater than could resolve. Hence, the router overwhelmed denied services and could also denied the connectivity. Moreover, attacks were observed to larger sites in particularly yahoo.com and amazon.com[61].

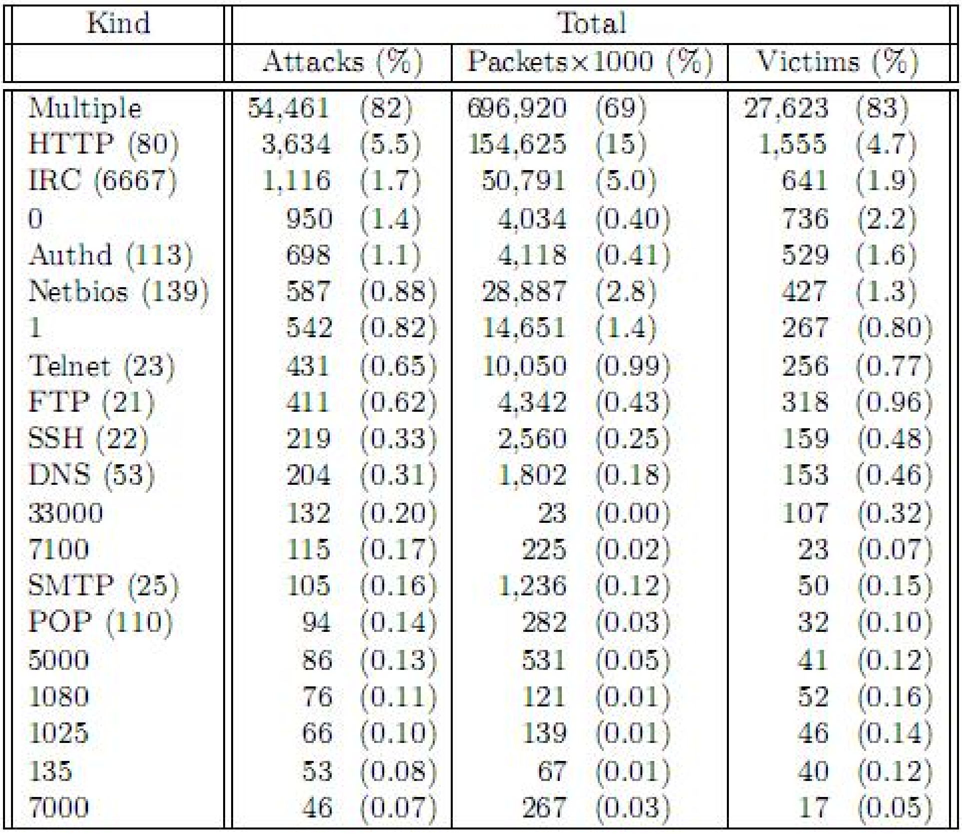
Table 3.2, describes attack protocols found in tracing procedures to allocate the source of the event. The backscatter event as monitored by a network telescope. The attack and the backscatter number packets can be observed. The majority of 93% and 88% were using TCP as protocol while a smaller percentage of 2.6% show which attacks were using ICMP packets, but on average was twice the number of TCP packets. The remaining of the attacks had a combination of different protocols per attack[61].

On the other hand, table 3.3 describes the monitored TCP attacks per service per target observed. The overall shows, popular TCP services, but having the majority of attacks targeted on multiple ports which were well defined through the port ranges, services such as HyperText Transfer Protocol (80), IRC(6667) and ports like 113 and



Kind	Total		
	Attacks (%)	Packets×1000 (%)	Victims (%)
TCP	64,952 (95)	949,373 (89)	32,275 (93)
ICMP	1,797 (2.6)	24,567 (2.3)	1,334 (3.8)
TCP/UDP	696 (1.0)	8,526 (0.80)	566 (1.6)
UDP	466 (0.68)	723 (0.07)	312 (0.90)
ICMP/TCP	441 (0.64)	63,728 (6.0)	356 (1.0)
ICMP/IGMP/TCP/UDP	118 (0.17)	342 (0.03)	104 (0.30)
ICMP/TCP/UDP	87 (0.13)	18,865 (1.8)	64 (0.18)
IGMP/TCP/UDP	27 (0.04)	42 (0.00)	22 (0.06)
Other	21 (0.03)	22 (0.00)	10 (0.03)
Other/TCP	18 (0.03)	62 (0.01)	18 (0.05)
ICMP/UDP	16 (0.02)	38 (0.00)	15 (0.04)
ICMP/IGMP/Other/TCP/UDP	16 (0.02)	368 (0.03)	13 (0.04)
IGMP/Other/TCP/UDP	10 (0.01)	56 (0.01)	8 (0.02)
IGMP/TCP	9 (0.01)	32 (0.00)	8 (0.02)
ICMP/IGMP/TCP	7 (0.01)	4 (0.00)	7 (0.02)
ICMP/Other/TCP	6 (0.01)	13 (0.00)	3 (0.01)
ICMP/Other	6 (0.01)	3 (0.00)	4 (0.01)
IGMP/Other/TCP	5 (0.01)	145 (0.01)	5 (0.01)
Other/TCP/UDP	5 (0.01)	2 (0.00)	5 (0.01)
IGMP/Other	5 (0.01)	3 (0.00)	4 (0.01)

Table 3.2: Attack Protocols found at trace analysis by Moore *et al.*[61]



Kind	Total		
	Attacks (%)	Packets×1000 (%)	Victims (%)
Multiple	54,461 (82)	696,920 (69)	27,623 (83)
HTTP (80)	3,634 (5.5)	154,625 (15)	1,555 (4.7)
IRC (6667)	1,116 (1.7)	50,791 (5.0)	641 (1.9)
0	950 (1.4)	4,034 (0.40)	736 (2.2)
Authd (113)	698 (1.1)	4,118 (0.41)	529 (1.6)
Netbios (139)	587 (0.88)	28,887 (2.8)	427 (1.3)
1	542 (0.82)	14,651 (1.4)	267 (0.80)
Telnet (23)	431 (0.65)	10,050 (0.99)	256 (0.77)
FTP (21)	411 (0.62)	4,342 (0.43)	318 (0.96)
SSH (22)	219 (0.33)	2,560 (0.25)	159 (0.48)
DNS (53)	204 (0.31)	1,802 (0.18)	153 (0.46)
33000	132 (0.20)	23 (0.00)	107 (0.32)
7100	115 (0.17)	225 (0.02)	23 (0.07)
SMTP (25)	105 (0.16)	1,236 (0.12)	50 (0.15)
POP (110)	94 (0.14)	282 (0.03)	32 (0.10)
5000	86 (0.13)	531 (0.05)	41 (0.12)
1080	76 (0.11)	121 (0.01)	52 (0.16)
1025	66 (0.10)	139 (0.01)	46 (0.14)
135	53 (0.08)	67 (0.01)	40 (0.12)
7000	46 (0.07)	267 (0.03)	17 (0.05)

Table 3.3: TCP attacks per target ports[61]

0[61].

DDoS attacks DDoS is a phenomenon of multiple compromised systems, attacking to a single target, causing denial of legitimate resources. The flood phenomenon can crash the system, if the capacity of requests is greater than the processing capability of the target. Therefore, the requests of the victims are lost in the unsolicited traffic[76]. In addition, in DDoS attacks, attackers in order to succeed maximum attack rate, combine the resources of multiple hosts. The attackers compromise a set of Internet hosts by installing a service or a daemon either manually or automated, to create "bots" and attack in full scale. This service allows the remote control by the attacker and variants of attacks. Through this daemon, the attacker can succeed coordinated attack with all the compromised hosts[61].

Andersson *et al.*, according to his Internet draft for the unwanted traffic in 2006, observed that DDoS traffic can be originated from everywhere on the Internet. In particular, regions with pipes and poorly managed hosts were used for the launch of these kind of attacks. Meanwhile, attackers preferred devices with large returns with the minimal processing effort. Furthermore, it was observed that backbone devices could easily absorb these kind of attacks without a serious impact on the network. However, DDoS attacks had a significant impact on end-hosts, with traffic arriving from many different directions [2].

On the other hand, Bouzida *et al.* by describing the architecture of a typical DDoS distinct the names of masters and slaves. The attacker begins a new session with a master host and the service is launched. This daemon can offer many facilities, so the attacker can have the opportunity to launch the desired attack. After the connection with the master host, the attacker launches the attack to one or many victims through the master. The master, after receiving the proper commands, sends notices to the slaves and launch a great scale of attack. Furthermore, the slaves send a notice to the master to inform for their condition(alive or not)[12]. Tools such as Shaft, TFN, TFKN2k, trinoo, were one of the most popular distributed attack tools. The target sends the reply to the source believing that is an appropriate response[65].

Moreover, Darmohray *et al.* refers that DoS attacks recently could be very challenging even to secure sites. Attackers use DDoS to gain an appropriate amount of processing power and cover their traces through other hosts. Through their experiment, it was observed that a traditional firewall can accept up to 500 SYNs per second, and if the attack is persistent and more than 500 SYNs, then the host can be temporarily unavailable[24]. It was observed that the speed of these attacks could reach up to 1.8-20 connection attempts/sec, by absorbing 2000-32000 bits/second. Even if these attacks seem not highly dangerous, the combination of hosts creates significant problems. In addition, it was observed that the generation of less data quantity helps to the difficulty of the detection of these attacks. On the other hand, by sending large quantities, it might be helpful, but it could be noticed more easily[76]. In conclusion, it must be mentioned that DDoS attacks can be long lived if combined with smaller events. Consequently, an address diversity is needed , mostly, in order a network telescope to accurately determine the range and the continuity of an attack[4].

Backscatter Analysis

Backscatter analysis was the first method to analyze unsolicited packets from DoS or DDoS attacks. Moore *et al.*, were the first used and developed this method basis to the research conducted. Unfortunately, the models measuring these endemic incidents through a network telescope are not plenty. Thus, backscatter analysis is used by most of the researchers and it is the main method for measuring this kind of activity through a telescope. We are referred to this, because our reader already read many times the name Moore *et al.*, so far. Therefore, the opinions of these researchers are more important, because they were the first implementing and analyzing methodologies for a network telescope.

Grace *et al.*, describes that backscatter is the excess of DoS attack. This excess derives from SYN flood which is a stream of TCP/SYN packets sent to the target. When attacker sends TCP/SYN packets from spoofed sources, the SYN packet received from the victim is for synchronization on a new connection. If the source was not found, the target will try allocation of new structures for the connection. Hence, if the target replies with SYN/ACK, the traffic becomes backscatter. Therefore, backscatter analysis is crucial, because of the approximation of accuracy provided by DoS activity. On the other hand, there is few information available since companies do not report this kind of private and sensitive information for further research[34]. Figure 3.3 is a basic scenario of backscatter principle. The attacker already uses three hosts with spoofed addresses. One is an active host and the other two are passive and observed by a network telescope[31].

Moore *et al.*, by the use of backscatter analysis, had observed 2000-3000 per week DoS attacks and for a period of three years monitored 22 traces revealing 68,700 attacks. Also, estimated the boundaries of these attacks in which the excess of these were over 100,000 packets/second[61]. Moreover, Yegneswaran *et al.* by examining Internet Sinks use backscatter analysis. The backscatter data were DoS attack responses and they were used before for the characterization of attack behaviors. Furthermore, through their experiments from Internet Service Providers observe that TCP packets with ACK/RST flag were the most usual response to a SYN flood. In figure 3.4 - a time series graph of backscatter volumes monitored from a service provider network telescope over 12 hours period- the vertical lines describe the less most usual duration spikes of SYN attacks. The ICMP TTL packets, can be manipulated either as loops created by routers or DoS attacks with lower starting Time To Live(TTL)[93].

Furthermore, with the active sink placed to the service provider, they had the opportunity to conduct SMTP analysis. They observed that there were specific address attracting large SMTP scans with range of 20-50 scans/second. Therefore, they concluded that by creating hot-spots in a network telescope range could be a good source of misconfiguration. With further observations from the telescope found that cable-

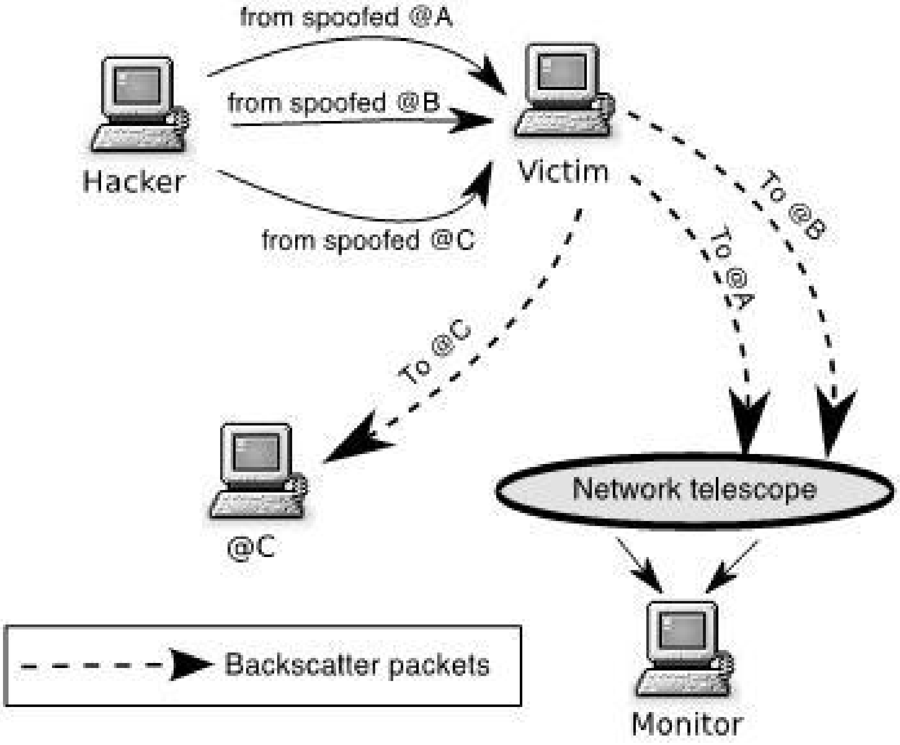


Figure 3.3: A simplified version of Backscatter[31]

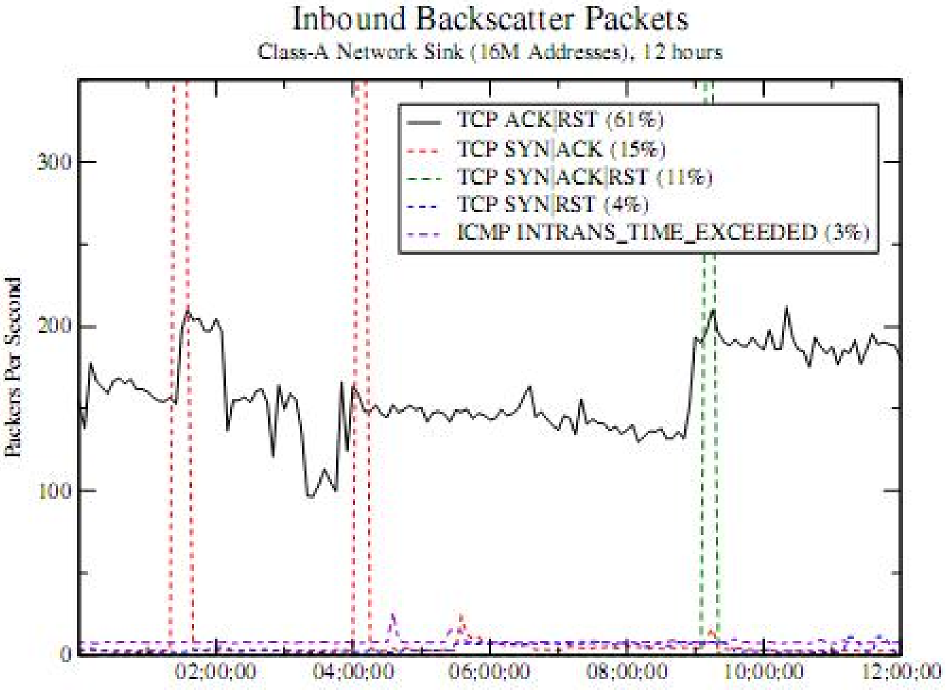


Figure 3.4: Service Provider's Sink observing backscatter data over 12 hrs period[93]

modem and DSL subscribers where the original source of these scans, but when they set up an SMTP responder, in order to find the source, discover that the source was a major vendor and that the emails received were firewall logs[93].

On the other hand, McPherson *et al.* in order to capture the activity of unsolicited data and analyze it propose that query blocks of addresses must be advertised . Then, by the use of a traffic collector such as TCPdump and static query collector, there is high possibility the backscatter data will be gathered and monitored from this block[53]. However, backscatter hypothesis can be biased through the port scanning packets arriving at a network telescope. This was observed, because not all TCP RST packets were backscatter data. Therefore, it might be needed verification through a vast majority of sites observing the same phenomenon[59].

Pang *et al.*, in their analysis, observed that the majority of the scans were TCP RSTs and SYN/ACK. Hence, these responses in vast majority were coming from flood attacks such as SYN-floods. Their analysis based on a /8 network, resulted in better conclusions. A significant portion of unreached messages showed that the flow arrived from spoofed source addresses from port specific in particular 53 and 1026. The first thought was from DNS poisoning, but these UDP packets can be observed in many networks[68]. Moreover, according to Francois *et al.* and their research, there was a large amount of monitored addresses which should never appear on the Internet. These misconfiguration were either of routers or firewalls that could be deficient devices. Furthermore, they conducted a backscatter analysis and compared the results found with active addresses. Also, they observed that multicast addresses where used

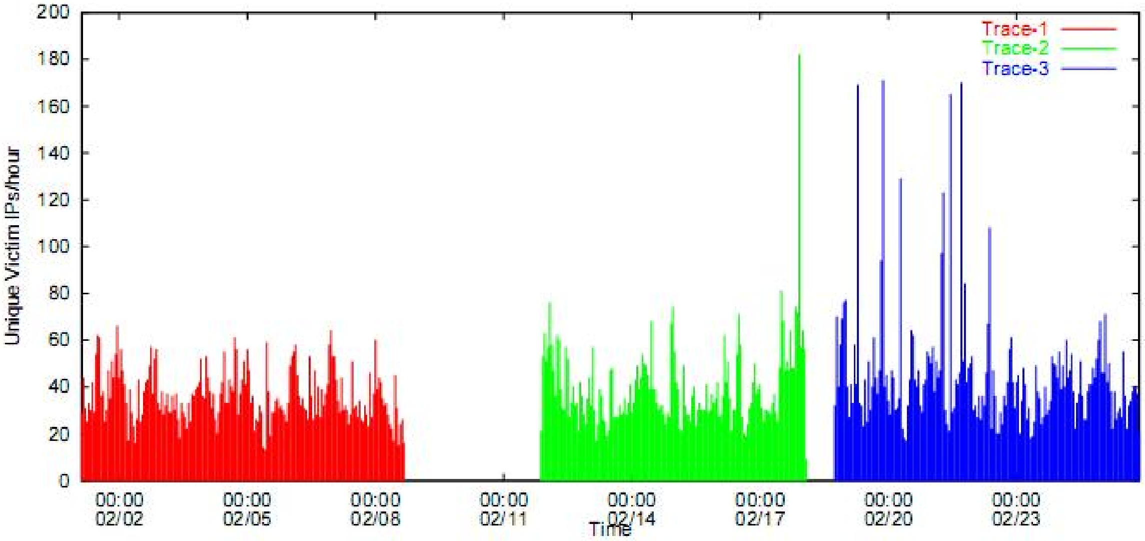


Figure 3.5: Estimation of attacks/hour[65]

as source addresses which could only be used as destination addresses. Further to the research, they observed that many of the unsolicited packets monitored by a network telescope were from misconfigured routers/firewalls/NAT from targets. Additionally, there was also an observation from Honeypot data. They observed that had different results through the private network and there was a possibility that these unsolicited traffic were arriving from misconfiguration of the local network[31].

Moreover, in order to describe the results of various measurements with backscatter analysis, figure 3.5 will be described. Figure 3.5, is time series graph of targets throughout three traces collected over one hour period. The gaps in the graph corresponds to the gaps of the traces in one hour periods. If it is observed the 02/20 (Feb 20), there is more than 150 targeted addresses per hour which were attacks against the hosts in the network monitored. Also it can be observed that the ratio increased above five in certain occasions[65].

Further to the analysis of DoS unsolicited data, Carl *et al.* described that monitoring packet headers in a network could be useful for activity profiling. They defined it as the average packet rate of a flow which consecutive packets of similar fields are consisted. The time between consecutive packets describes the activity level. Therefore, the measurement of network activity could be found by the result of the sum over average packet rates of inbound and outbound flows. Furthermore, analysis of UDP and TCP services must be included in order the number of flows to be determined. Also, they clustered the similar flows in order to determine the summation of constituent flows. Additionally, they described that spoofing results to a finite probability which the monitored range of addresses will accept packets from unsolicited traffic. They also referred to Laura Feinstein *et al.* that focus on activity and the uniformity of addresses. They clustered basis to addresses of the flows of the targeting hosts behind monitoring points[33].

Moore: The backscatter analysis Moore *et al.* was the first researcher who referred to this method. Moreover, referred to backscatter analysis as the probability of a host on a network (local or global) of receiving, an unsolicited response from a target is $1 - (1 - \frac{1}{2^{32}})^m$, assuming a reliable delivery and for every packet in an attack there is at least one response, during m packets of attack. Hence, if n addresses monitored the probability of the observance of a packet from attack is: $1 - (1 - \frac{n}{2^{32}})^m$. Therefore, the unsolicited responses observed through an attack with m packets is $\frac{m}{2^{32}}$, in a single host. By observing n addresses the expectation of responses monitored: $E(X) = \frac{nm}{2^{32}}$[61]. Furthermore, it was mentioned that the estimation of an attack rate could be calculated by the multiplication of the average rate of arrivals of the unsolicited(backscatter) data by 256. By this method, it was found that 50% of atacks have a packet rate more than 350 packets/second and intense attacks over 679,000 packets/sec[65].

Therefore, by observing an address range that is large enough sampling techniques becoming effective. In this sampling the identity of the target, the kind of attack and a timestamp are properly fitted for the attack duration estimates. By using $R \geq R'\frac{2^{32}}{n}$, it could be observed the average arrival rate of backscatter data monitored by the address range, in order to estimate the attack rate focused on the target. *R'*, is the average rate arriving as backscatter from the target and R is the conceived rate of attack in packets/second[61].

In addition, researchers in order to extract the packets from a backscatter phenomenon removed the packets from legitimate hosts monitored by the network telescope. After that, removal of response traffic that did not correspond was needed. Furthermore, removed TCP RST packets used for scanning, because port scan activity does not consist as response traffic. However, RST scanning can be used to infer policy rules of firewalls. Therefore, it is preferably to exclude this kind of packets since there is no reflection of DoS attacks. Hence, they used scanning techniques to remove TCP RSTs. Additionally, they performed aggressive duplicate packet suppression in order to assure that packets were multiplied either on network or from DoS attack targets. The removal of packets with same flow tuple was necessary. For ICMP errors extraction of IP address, protocol and ports was needed[61].

Moreover, after the backscatter data received, aggregation was needed, because aggregation of data is not a simple technique and technical challenges appear. For instance, attacks with TCP and ICMP could be classified together or separated. Furthermore, the problem of starting and ending time was arise, because in variability a threshold can be biased of analysis between attacks of longer duration and low packets will be biased through a large number of attacks with high rates. Therefore, knowledge of the attack or of the adversary intentions was needed, otherwise an active classification system was not possible to work properly[61].

On the other hand, the identification through a flow method is commonly used for Internet traffic successfully. Hence, flow can be determined as the series of consecutive packets with the same targeting IP address shared. Therefore, the first packet observed creates a flow for a target and association with the target of that flow if the network

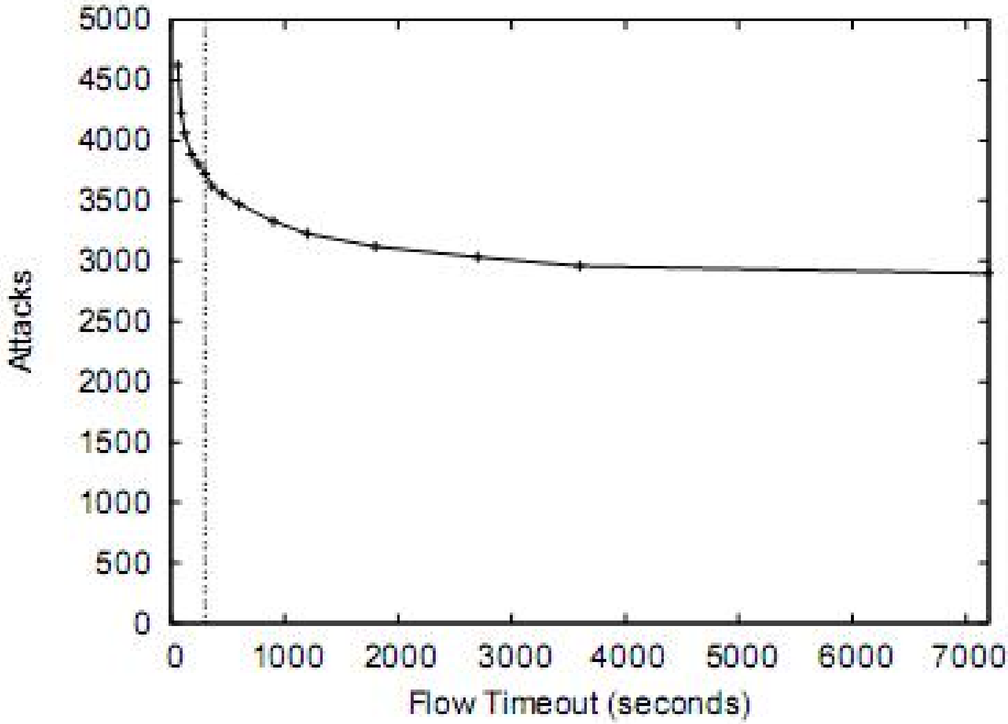


Figure 3.6: Sensitinity on attacks bases to the flow parameter[61]

telescope observes the packets within timeouts of this flow. Therefore, parameters are needed and can influence the outcomes, because longer attacks with shorter timeout leads to a bigger amount of short attacks. Hence, backscatter data arriving from the target, arrive from the same flow, have a timeout flow that defines the maximum interval of two packets. The flow timeout determines the start and ending of flows and notes that smaller flows partition continuous backscatter traffic into smaller ones. Figure 3.6, describes the attack to a range values for the timeout parameter. As it is observed, the curve describes the number of attack flows that changes when variety between 60 sec to two hours of timeout parameter exists[61].

After the partition of the packets into flows, there were three more parameters that could help to the classification of these attacks: packet threshold, attack duration and packet rate. Therefore, a sensitivity analysis can be used. When the values vary, the default values of parameters can be used and attained from the analysis. Consequently, in order to classify a flow the packets threshold is needed. Packet threshold defines the minimum packets observed in a flow for classification attack. There must be a filter for attacks in order smaller attacks that have greater impact on a host to be detected. Therefore, biased collection must be avoided[61].

Therefore, the attack duration which is an attack's flow satisfying the time duration threshold and the minimum time among start and end packets in a flow is needed for classification. Furthermore, filtering short attacks that have impact on victim are preferable instead of large durations which can not be qualified easily[61].

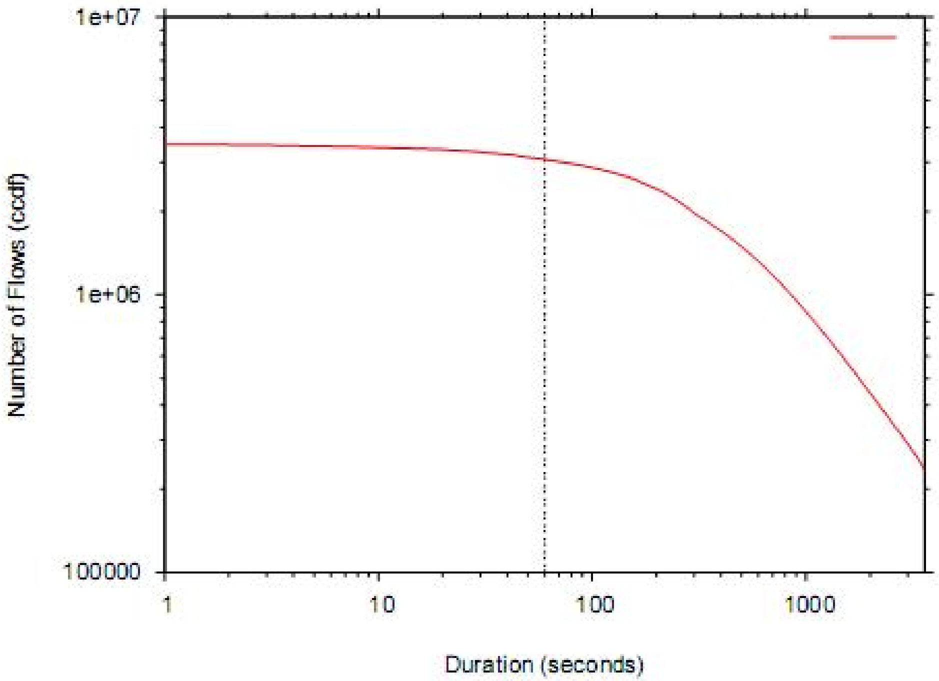


Figure 3.7: Sensitivity of flows to the duration parameter[61]

Figure 3.7, describes the sensitivity of counting flows base to the attack duration. If the curve is observed, there is a connection with the number of flows and the attack duration. Depending on the duration, varying from one second to an hour, the flows also depend. Also, it can be observed that durations less than 100 seconds are insensitive to the number of flows[61].

If the attack has a certain threshold of maximum rate of packets during the flow, then it can be classified successfully. Therefore, if the rate is more than a minute there are enough packets for classification. Hence, the flow of the packets must be greater than the threshold of packet rate. This is because larger packets thresholds conclude into less qualification of the attacks. Also, by filtering, the attacks can be classified into ones with the greatest impact[61].

Further to the specification of extracted packets, there must be an examination. While there is an examination, extraction can be arranged by observing the protocols. Therefore, in an IP protocol packet such as TCP, UDP can be extracted. Also, in TCP flags such as SYN/ACK and ICMP payloads which contain information about addresses, protocols and ports can be extremely helpful to organize a database in which the records can characterize each attack. Furthermore, this database could be used for further analysis of the attacks[61].

On the other hand, in order to validate the data needs a further examination to increase the confidence. Furthermore, there is an emphasis that not all packets provoke a certain response. Therefore, some of these packets cannot be used for monitoring and probing. Additionally, to their investigation they use the Anderson-Darling test for the determination of the distributed addresses that observed in each attack with uniform distribution. It was observed from datasets of a /16, that 98% of targeting IPs, backscatter analysis had almost a definite level of correspondence if the data were enough for evaluation and examination[61].

3.3 Pandemic Incidents

One of the worst threats today in the Internet are worms, which are easily created and have high probability to cause major damage. They can flash the BIOS, modify the system and sometimes conclude to DoS attacks and reveal crucial personal data[56].

Internet Worms

In recent years, one of the most dangerous threats in a network or all over the Internet are worms. Programs spreading without human help and overload networks and critical resources. Worms such as CodeRed I/II had specific targets usually Internet sites and resources for instance databases. The damages left were many of the times critical and economical disasters in banking systems happened. In this section, various worms and how they observed through a network telescope will be described. Description and analysis of worms basis to the epidemiological models and worm models in extent will not be examined.

Worm is a self-replicating/propagating program which exploits vulnerabilities without human intervention in order to infect hosts. The infected hosts continue the propagation and the spread of the worm. This also could be observed through figure 3.8[56]. Malicious programs such as worms send copies of itself to many locations over the Internet and a network. A network telescope has the ability to observe the frequency of these packets and compute the propagation in the network(global or local)[34].

Furthermore, there are few worm classifications as categorized from researchers which can help us understand the propagation models in the next section. The contagion worm propagates parasitically through normal communication channels. It can use exploits by infecting servers and vulnerable hosts' browsers. Additionally, it can propagate through peer-to-peer networks. On the other hand, a worm is not always needed to scan randomly. Meta-server worm, in particular, request the server for infected hosts and topological worms spread through the infected targets possessing local information[69].

Singh *et al.*, in their report for the EarlyBird System detecting worms proposed a different way of classifying the worms. They classified worms basis to multi-packet or single packet payload. The "substantial volume of identical traffic" worms has the characteristic that there is a stage before the initial traffic in order to infect further

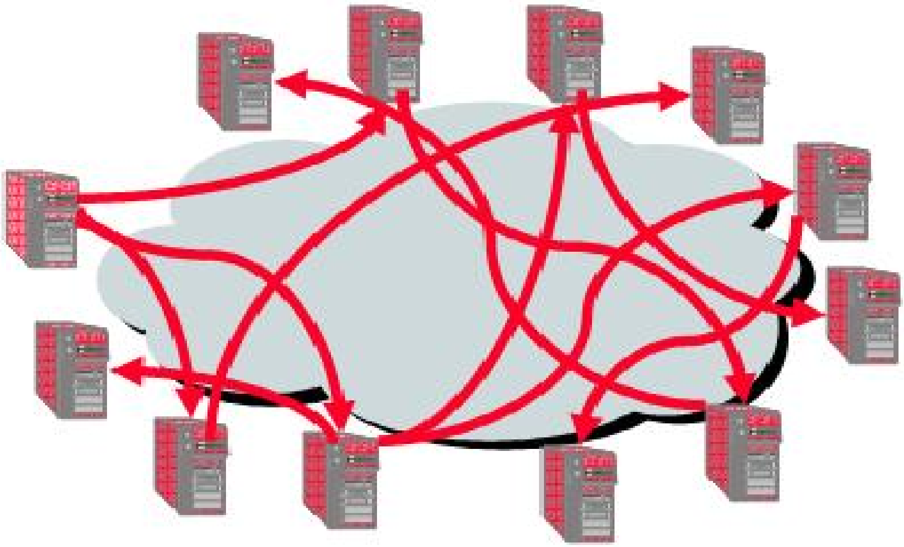


Figure 3.8: Worm activity[56]

hosts in a network before full infection. The “rising infection levels”, describes the steady increase of sources and destinations for infection. The “random probing”, make use of infected source to propagate by attempting to communicate randomly at certain ports and exploit services[81].

Moreover, the spread modeling for an active worm is the simultaneous scans and attempts for exploiting vulnerable hosts. When a vulnerability is found probes the hosts and copies itself to the vulnerable host. Then the infected host tries to infect other hosts, but when the worm phenomenon reaches unused address spaces its functionality stops. On the other hand, when users try to reboot the system and kill services or daemons because of the limited resources the system becomes more vulnerable and continues the infection. Additionally, when the worm will be found and the host will be patched properly the system recovers. Furthermore, it was observed that the attackers usually scan for vulnerable hosts before the release and create potential vulnerable hosts lists. Then the worm scans the lists and infects vulnerable systems in a network. After the infection of the hit-list(list of vulnerable hosts) is finished, worm will use hosts for propagation and as a base for further infections[17]. According to [59], worm attacks randomly generate addresses from vulnerable hosts already infected. With the network telescope, researchers managed to monitor $1/256^{th}$ of all addresses and observed the worm traffic[59].

Furthermore, it was observed that malwares and especially worms have a preference to nearby addresses, such as Blaster and Nimda[52]. Cuiy *et al.*, through a network telescope capture 66 different worms and unique fingerprints of 14 different categories. It was observed that most executables monitored had association directly with the worm[23]. Moore *et al.*, with a network telescope in 2001, observed the release of a self-propagating worm that was able to compromise 360,000 hosts in half a day and then could load a DoS attack against government site. Each worm followed from then showed

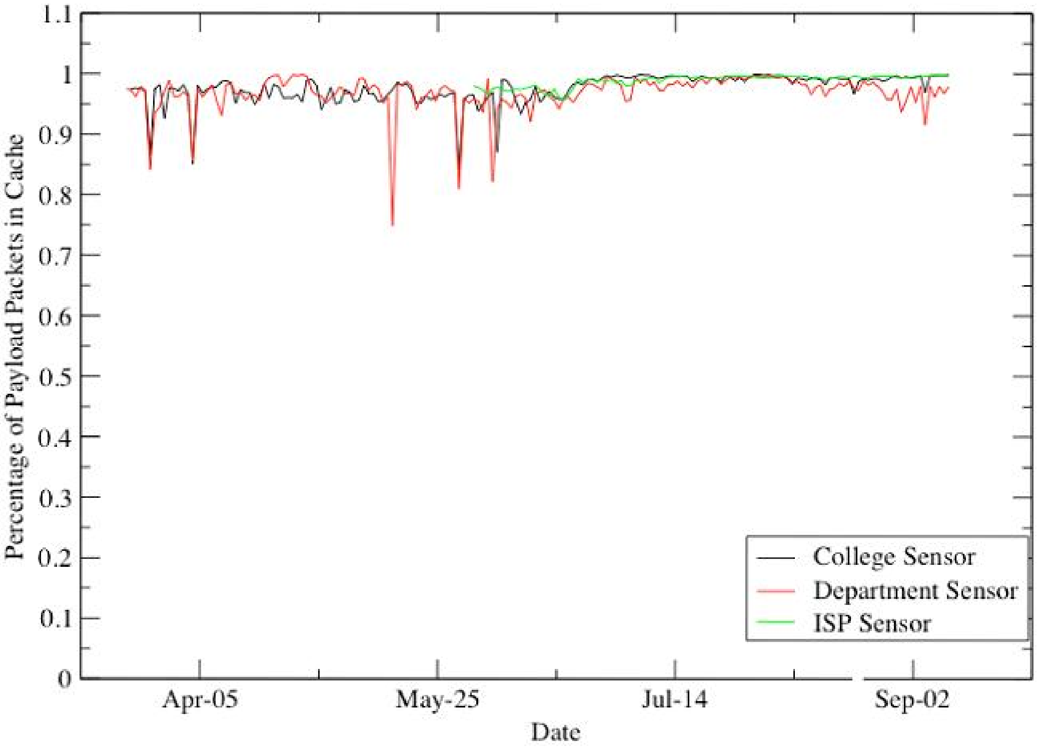


Figure 3.9: Payload Cache over 5 months [3]

an improve and some where using vulnerabilities that other worms left[65].

On the other hand, detection of worm attacks might be difficult, because architects of these programs add random filters by making the payload before and after the exploit useless. Also, worms could spread in many networks and the fragmentation created might be different in any case. Furthermore, it was observed that many worm programmers program in binary and the infected hosts can rewrite exploits at each point. Additionally, a worm can use contagion methods in order to slowly spread with the use of the minimum memory requirements[81].

According to Bailey *et al.*, in order to effectively monitor worms or threats arriving at the network telescope, proposed that because there are many payloads observed it would be efficient if new payloads are only stored. Also, it was proposed to store only the MD5 hash of payloads and only store payloads of unique hash functions. Figure 3.9 describes the percentage of payload packets in Cache over 5 months at three different sensors. It can be observed that 95% of hits at the cache were signature hits and most of the payloads had already observed at previous times.

Bailey *et al.*, with their research on Internet Monitor Sensor, described worms as standalone programs that are self-propagating and with the help of network as medium scan for vulnerable hosts and exploited them with the use of host interaction. Furthermore, they refer that network telescopes are useful in order to characterize, measure and track threats over a specific range. Internet Monitor Sensor provided an insight to

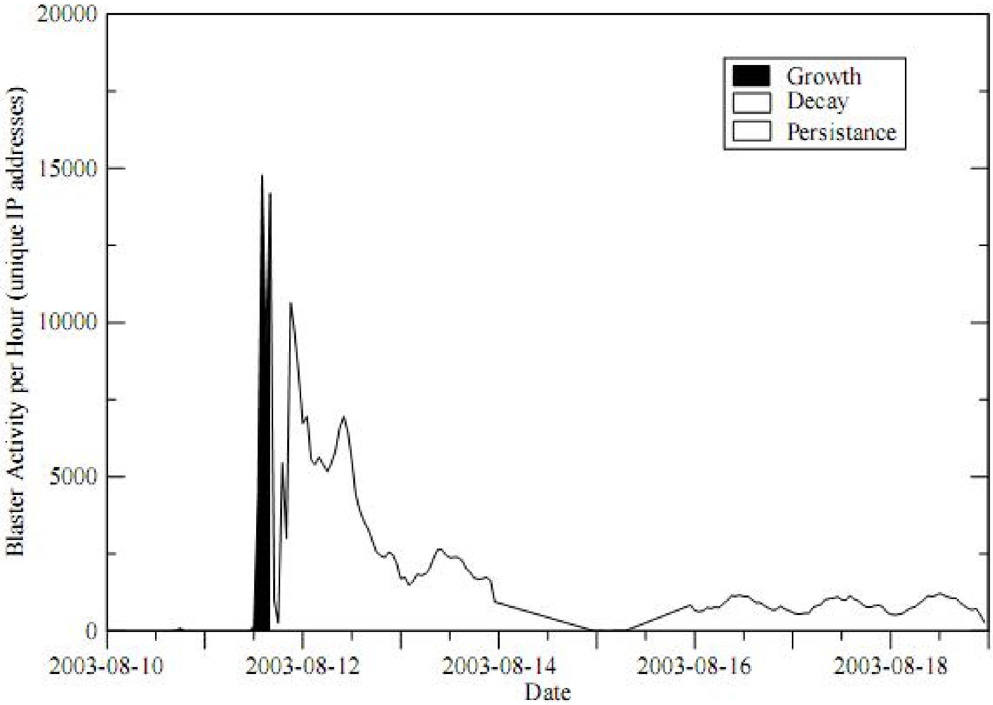


Figure 3.10: Blaster 3 phase lifecycle: as observed by the IMS[4]

worm behavior. Worm behavior -basis to the research- distinct through worm virulence (traffic flow resulted and paths congested), worm demographics(hosts infected and where; operating systems and bandwidth using), worm propagation and community response(policies employed for worm; affected organizations and responses; still infected hosts). As an example of the IMS analysis was the Blaster worm. Blaster affected Windows 2000 and XP running DCOM RPC services by using a buffer overflow vulnerability. It was observed that 60% of the addresses generated by Blaster were random and 40% located at /16 network and then the worm scan sequentially. The observed traffic gave information valuable for the release and propagation of Blaster[4].

Figure 3.10 is an illustration of 3 phase Blaster lifecycle, using SYN to TCP on a /8 network at port 135. The phase that first observed is the growth and describes the rate based up to hundreds of thousands infections per hour. The decay phase which was the second phase describes the filtering implementations used to stop the spread of Blaster. The last one was the persistence of Blaster which continued in all 2004[4].

The IMS was able to monitor 286,000 IP addresses and described the characteristics of Blaster. Domain inspections showed that .net, .jp, .com were infected seriously. The 10% monitored showed that were dynamical assigned IPs. Also, it was observed that domaim inspections had a maximum growth rate of 40,000 hosts/hour. At the second phase, Blaster activity showed that it was start fading after the countermeasures taken. Furthermore, it was observed that Blaster wasn't aggressive such as Slammer and Witty. Furthermore, from the Blaster analysis found that it utilized TCP and

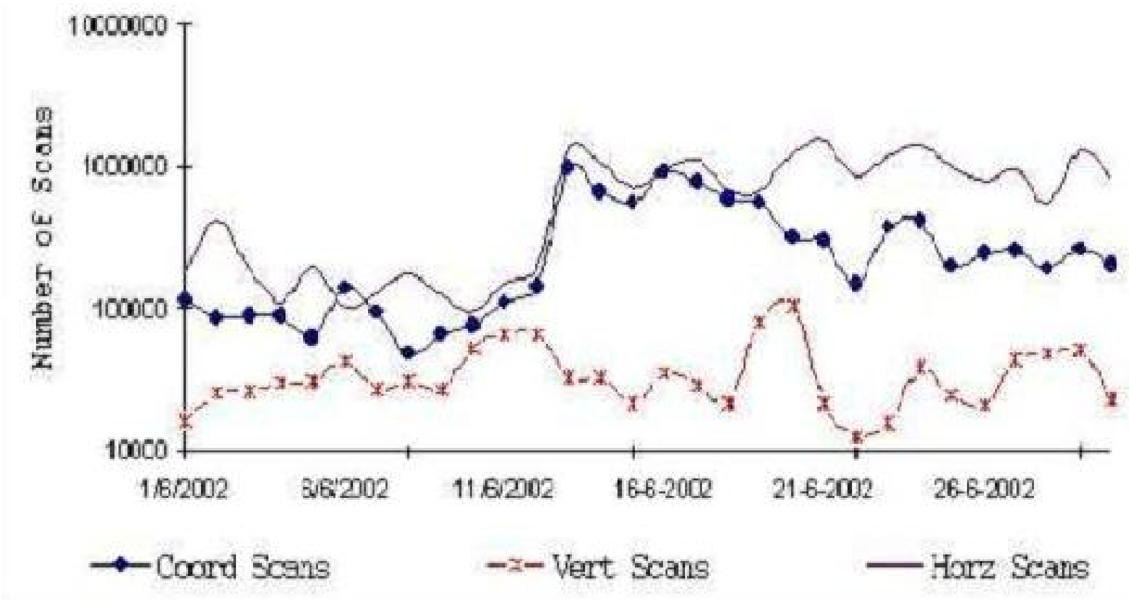


Figure 3.11: Horizontal and vertical scans, of coordinated distribution for June 2002[88]

thus tracking and characterization by passive network telescopes can be extremely difficult. Passive telescope were able to monitor SYN packets but not payloads. Further, IMS because accepted the first payload was able to differentiate the worm's variants. Additionally, IMS had the ability to monitor the new services installed with the worm attack and allowed the collection of payloads and scans[4].

On the other hand, worms such as Bagle and MyDoom, spreading on 2004, used ports such as TCP 2745 and 3127 respectively were monitored successfully over a week period in a /24 network monitored by the IMS. The top port captured was 2745/TCP and 3127/TCP for MyDoom. Also, it was observed from the behavior of these worms that MyDoom might change and new software uploaded[4]. Furthermore, Yegneswaran *et al.*, by monitoring the attempts of intrusions on the Internet, observed that network telescopes (monitoring destination ports) could be an effective way of monitoring. They mentioned that many intrusions were monitored days before the events. The most highly rated ports found to be 80, 1443, 137, of CodeRed, Nimda, Slapper, SQL-Snake and P2P scans respectively[88].

Figures 3.11 and 3.12 describe the distribution of daily scan types during June 2002. The indication of 60% of horizontal scans are from non-worm scans. It can be also observed that surprisinply daily scans might be coordinated or arrive from distriduted sources. Furthermore, it can be observed that port scans include ports in particular 11, 53 and web server ports such as 8000 and 8080. Although the common nature of horizontal scans were vertical[88].

On figure 3.13, it can be observe the packets monitored per 5 minutes over 3 days. The specific port was TCP 1023 which was Sasser.e operated. Two different telescopes observed the active on /18 and /17 networks. It can be observed that there are large short-lived spikes. Further from the research operated by Bailey and his colleagues

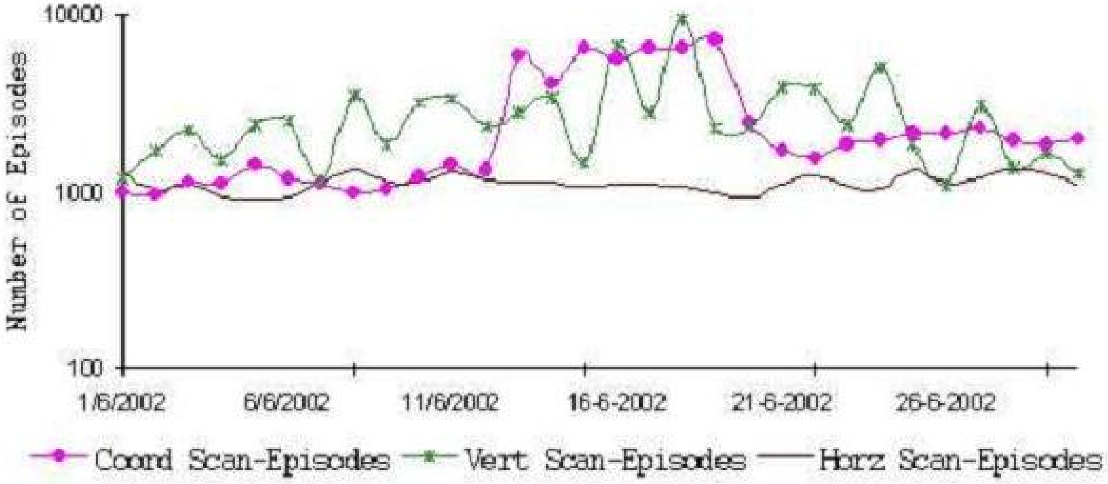


Figure 3.12: Horizontal and vertical scans, of coordinated distribution for June 2002[88]

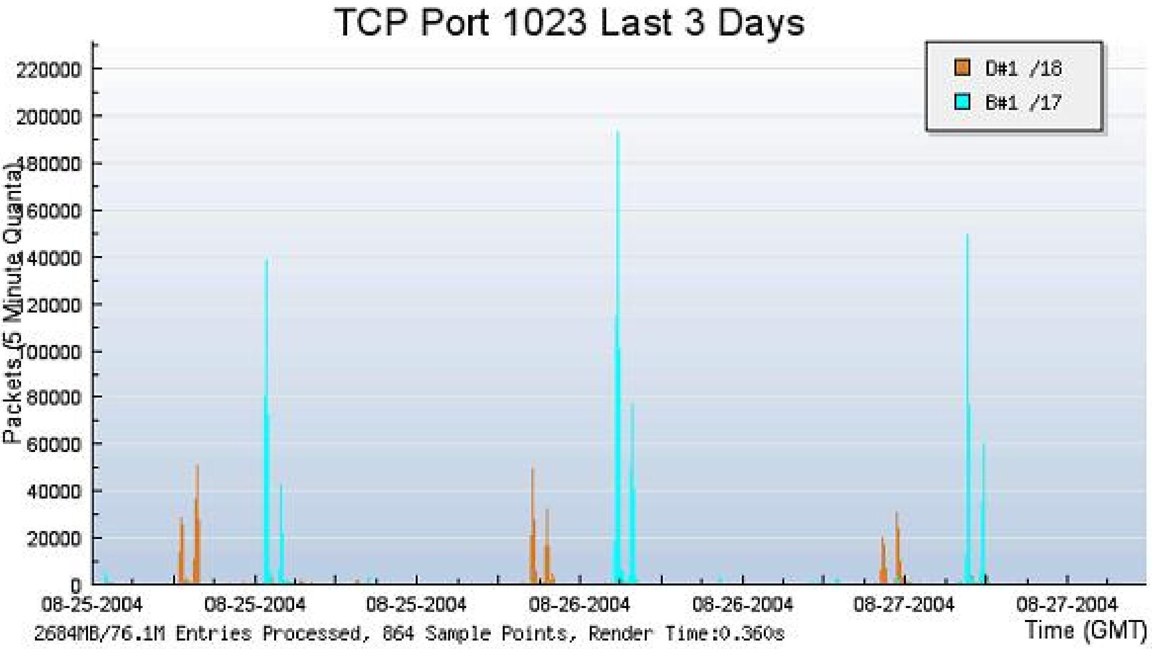


Figure 3.13: Sasser activity[3]

found that there were no captured data on 9898/TCP and only two different signatures on 5554/TCP and 1023/TCP. These signatures were observed through monitoring with a network telescope worms such as Sasser, Sasser.e and Dabber.a[3].

Moore *et al.*, also, observed that containment technologies can be used for effective blocking of infectious communications. This allowed an additional time for reducing the spread or even stop the infection. Furthermore, these mechanisms where the primary sources of protection through Code-Red attack or by isolation of infected hosts. Unfortunately, these mechanisms were not able to stop completely or guarantee the spread but provided a limited protection[65]. Furthermore, researchers rebuilt worm signatures in order a network telescope database could easily recognize further phenomena. Additionally, these fingerprints were saved with all the characteristics and the ports of every worm, because worms have the characteristic that attack to certain ports[81].

Chen *et al.*, emphasized on collected traces for example in a /8 network which could accurately emulate the hosts vulnerable distribution. The targeted addresses can be used for the estimation of distributed victims in group i. Consequently, the probability is: $p_g(i) = \frac{number\ \ of\ \ addresses\ \ in\ \ group\ \ i}{total\ \ number\ \ of\ \ collected\ \ addresses}$[18].

Code Red I/II CodeRed was launched at July 13, 2001. It was exploiting a vulnerability of the Microsoft IIS Web servers. After compromising the server infected site was defaced with the phrase “HELLO!Welcome to http://www.worm.com! Hacked By Chinese!” and this typed only if the language were English. At the 1st and 20th of each month it was spreading and at the 20th it was releasing an attack. The flood attack was against the www.whitehouse.gov. White House changed the IP address and this cause CodeRed to extinct for date up to the 20th. On the other hand, CodeRed II launched on August 4, 2001. It was completely different code base than CodeRed. It was creating a root backdoor and was programmed to crash NT working on Windows 2000. Furthermore, the code was preferring nearby addresses for propagating and it was programmed to destroy CodeRed and safely extinct in Oct 1,2201[69].

Code-Red worm I was scanning for a vulnerability on the Windows Internet Information Server with buffer overflow attack. It was named after all the pages infected and the sites marked as “hacked by Chinese”. Code-Red I in its first operation searched for vulnerable targets by the use of a IP generator. In its second operation at the 20-28 of every month stop propagation and loaded a DoS attack against the White House governmental site[88]. On the other hand, Code-Red II was using the same deficiencies as Code-Red, but it was completely different. Code-Red II had a propagation mechanism generating addresses and masks whose size similarity is determined from infected and probed hosts. It set up a root(administrative) backdoor and allowed remote execution[88].

From July 4 up to Augusts 25, 2001, researchers tried to analyze successfully the spread of the Code-Red worm. By using a network telescope they capture the traces of hosts probing random addresses. It was easily observed because of the differences between

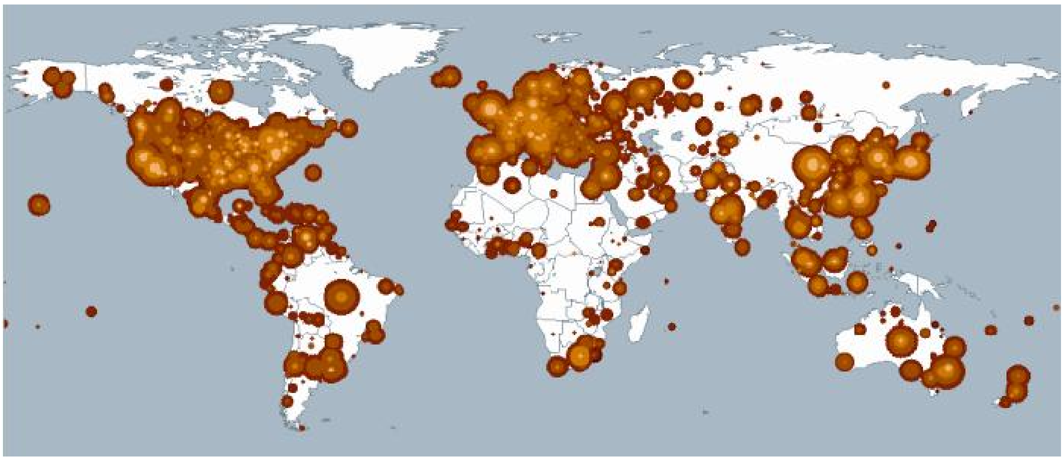


Figure 3.14: CodeRed propagation in ten days[59, pp.5]

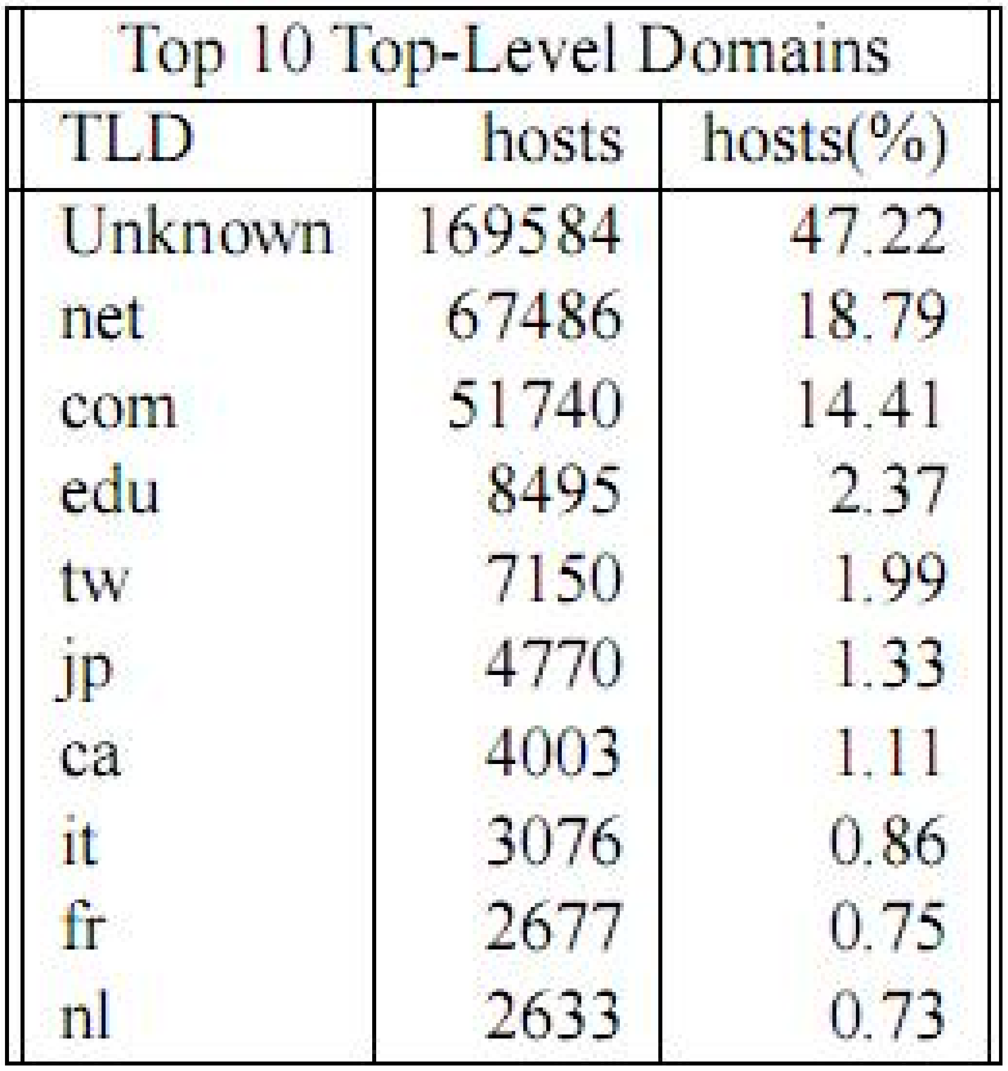
backscatter data and worm requests[65].

In few hours Code-Red worm infected more than 359,000 hosts of unique IP addresses. The outcome of the research showed that the infection rate was approximately 2,000 host/minute[65] in only 10 hours. There was no effective patching mechanism and there was a severe economic damage. Figure 3.14 shows the infection of CodeRed I worm in ten days. As it is observed, there were severe attacks globally. CodeRed was programmed on July 20 to deactivate and restart its operations on August 1. Everybody knew through the media for the return and the only mechanism protecting against was patching. Furthermore, the DHCP effect that CodeRed left was treated in order to produce skewed statistics over a period.[59].

In order to classify the victims researchers searched for the reverse DNS records. The outcome showed that only 22% of the hosts where identifiable. Broadband users and dial-up where the major victims of Code-Red worm, but it was found that Code-Red had a preferability to web servers. Additionally, 21% were home and small businnes devices and only 13% of the DoS attack targets had this characteristic. The top most infected domains were EDU, COM and NET[65].

In the case of CodeRed research it was observed that 47% of infected targets had no DNS records. Therefore, the top level domains were impossible to be determined. Domains like com, net, edu, mil and gov were infected. It was observed that 136 and 213 mil and gov hosts were infected respectively. Further, 390 private hosts were infected raising the possibility that more private networks were influenced[56]. In the case of CodeRed II the infection hosts remained the same and as it was monitored observed that there was no significant change on the unsolicited packets[62].

Figure 3.15 describes the probe rate from every 2 hours on August 2-22 in a /8 network. The spike on August 6 displays the backscatter flow from the DoS attack. Because of no susceptible hosts in the /8 network the probe rate seemed the same as CodeRed. Therefore, there was a difficulty examining infections, because could not easily distinguished from CodeRed[62].



Top 10 Top-Level Domains

TLD	hosts	hosts(%)
Unknown	169584	47.22
net	67486	18.79
com	51740	14.41
edu	8495	2.37
tw	7150	1.99
jp	4770	1.33
ca	4003	1.11
it	3076	0.86
fr	2677	0.75
nl	2633	0.73

Table 3.4: Top Level Domains infected by CodeRed[62]

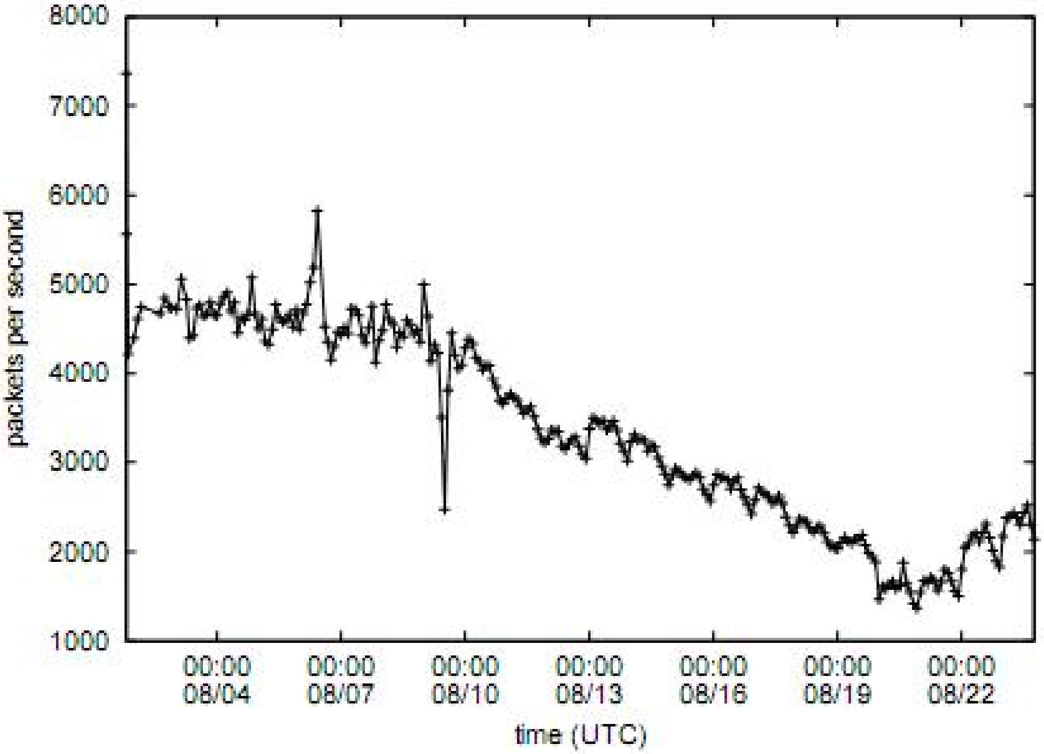


Figure 3.15: The probe rate observed by a /8 network[62]

Table 3.4, describes the top level domains infected by CodeRed. It can be observed that on July 19 .net, .com and .edu were mostly infected. Governmental infrastructures were also infected[62].

Witty Witty worm was another destructive worm on March 19, 2004 observed by a network telescope. It had a destructive payload and write 64k to disk randomly and launched more than 100 hosts. Witty had the ability to infect security softwares and spread quickly -even with small population infected- with approximately 12,000 hosts at 45 minutes[59].

It was targeting Internet Security Systems of network security products. Additionally, Witty was using a pseudo random generator and sent 20,000 copies of itself to different destinations. Two telescopes monitored and captured the activity of Witty one in CAIDA and one on University of Wisconsin. Network telescope help the researchers to trace the propagation and also found the first infected hosts (or patient zero). They could observe this phenomenon even when one packet arrived at the monitored space. They could determine the seed of the pseudo random generator and could calculate the random numbering system the target generated and how many packets sent by each of these hosts. With knowledge of the packets sent researchers managed to observe the propagation model of Witty worm. It was characteristic that CAIDA during the 75 minutes of propagation, the telescope congested from the packets since witty was

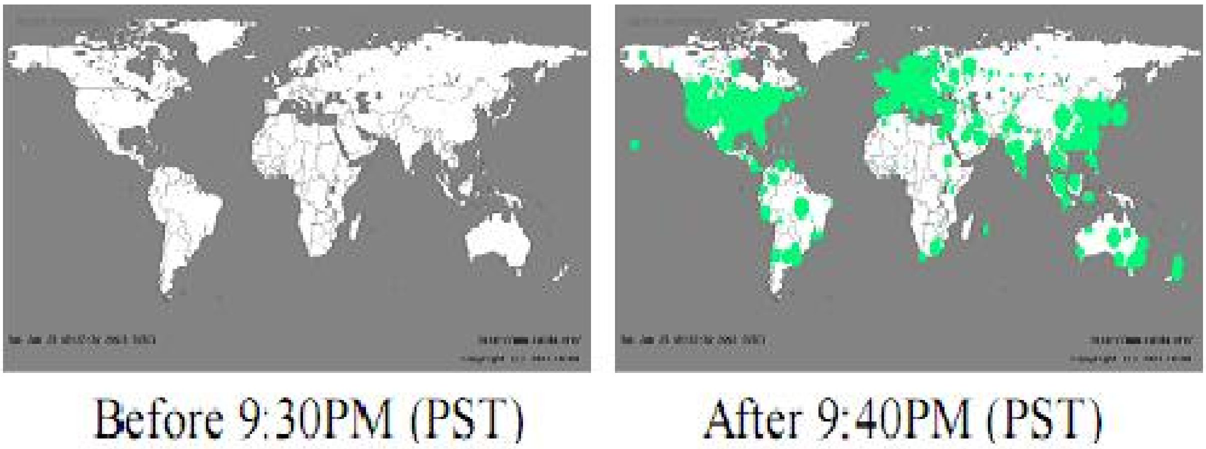


Figure 3.16: Sapphire infection map[59]

using 95 Mbps from 100 Mbps in total. Furthermore, CAIDA telescope monitor more packets than Wisconsin, because of the different connectivities. The patient zero (or the first infected target) was not generating with the same patterns as the other infected targets. Consequently, there is high possibility that this was the first machine used by the adversary to launch an attack. Finally, researchers found the source of the Witty attack and it was observed that the IP address corresponds to a European Internet Service Provider's host[34].

Furthermore, Witty was creating a bandwidth limitation and its payload was easily to corrupt disk blocks. The flaw exploited by Witty was announced for patching the previous day. The network telescope's initial observations revealed that the spread seeded by a list targeting a military base. The analysis also revealed that the first infected host was European retail Internet Service Provider and witty worm was written from a professional[69].

Casado *et al.*, in their analysis for the Witty worm, found that 1 in every 256 packets observed sent by Witty infected host. The research was based on a /8 network. Furthermore, it was observed that Witty was using a pseudo random generator using linear congruential number generator. Additionally, it was observed that by examining the packets sent by Witty infected hosts, the randomness becomes deterministic and predictable. This determinism offers the ability to predict the packets sent to the network telescope and offers the possibility to compute the bandwidth of the local link with accuracy[14].

Sapphire or SQL-Slammer and SQL-Snake In the case of Sapphire(or SQL Slammer) was monitored that 100,000 hosts infected in only 10 minutes and observed that more than 55 million probes send per second globally. There were severe damages to the Bank of America ATMs and in Airlines. From figure 3.16 it can be observed the severity monitored by a network telescope in Jan 24, 2003[59].

Slammer worm exploited UDP services and entire worm could fit in a single packet.

Additionally, the worm infected more than 75,000 in 10 minutes and at its peak double its propagation[69]. When it begun spreading in ten minutes more than 90% of vulnerable targets were infected causing severe damages to the financial systems, transportation and institutions. Slammer began its spreading on January 25, 2003 and was exploiting a buffer-overflow to Microsoft SQL-Server. It was considered as a high-speed propagating worm comparing to other malwares such as CodeRed. It was overloading the network and it had effectiveness by disabling the database servers. Furthermore, there were many backbone internet distruptions[58].

On the other hand, SQL-Snake was detected also on May 2002. The scanning model was allocated for MS-SQL running on port 1433 and to hosts with default accounts such as admin and started random scanning to all address ranges[88]. From the analysis of Bakos *et al.*, it was observed that SQL-Snake was based on an old vulnerability on SQL Server and combined with few features. The worm had the ability to execute ActiveX object commands via SQL Server and passed to non-password protected administration account[7].

Blaster and Sorbig On August 11, 2003 Blaster worm appeared. Blaster had the ability to scan addresses of a /24 from 0-254 and scan a network for port 135 TCP, listening on port 69 UDP and there was an attempt to connect when the vulnerability found. Then connected to port 4444 of TCP and download a certain file for execution. Unfortunately, Blaster was not easily detectable. No response to 135/TCP SYN and there was no active sampling and no 4444/TCP flow[66].

It was exploiting security flaws on Windows Remote Procedure Call(RPC) by copying an exploit and add that within the initial code. After that, a DoS attack was launched to windowsupdate.com. Within one week 100,000 Windows hosts infected and after a year it was steal active. Researchers in order to measure Blaster activity used a /8 network. This sensor was under the use of the Internet Monitor Sensor. Furthermore, prefiltering of the packets eliminated false positives as it was observed. Additionally, Blaster was an example of the lifecycle observed which all phases included such as latency, growth,delay and persistence. In the latency phase it was observed high activity of scanning on TCP port 135 correlating exploits from individuals or groups. The growth phase was followed by decay were effective patching and removal was efficient. Also, it was observed that after efficient treatment, the phenomenon of Blaster start diminishing. Finally, it was monitored that the persistent phase were small from the remaining infections[6].

After one year of the release Blaster was observed that the persistent phase was not diminished. In August 2004, it was discovered that more than 200,000 addresses monitored were scanning the darknet. Further to the research found that 90,000 IP addresses were under infection. The phenomenon of persistence was not new, but the infection of Blaster was interesting. Additionally, Microsoft released a tool to remove the worm executable and then patch the system. Unfortunately, six months later the same year was observed that the infected population still was large. Figure 3.17, describes the Blaster lifecyle. Anyone, can observe the four phases: latency, growth, decay, persistent

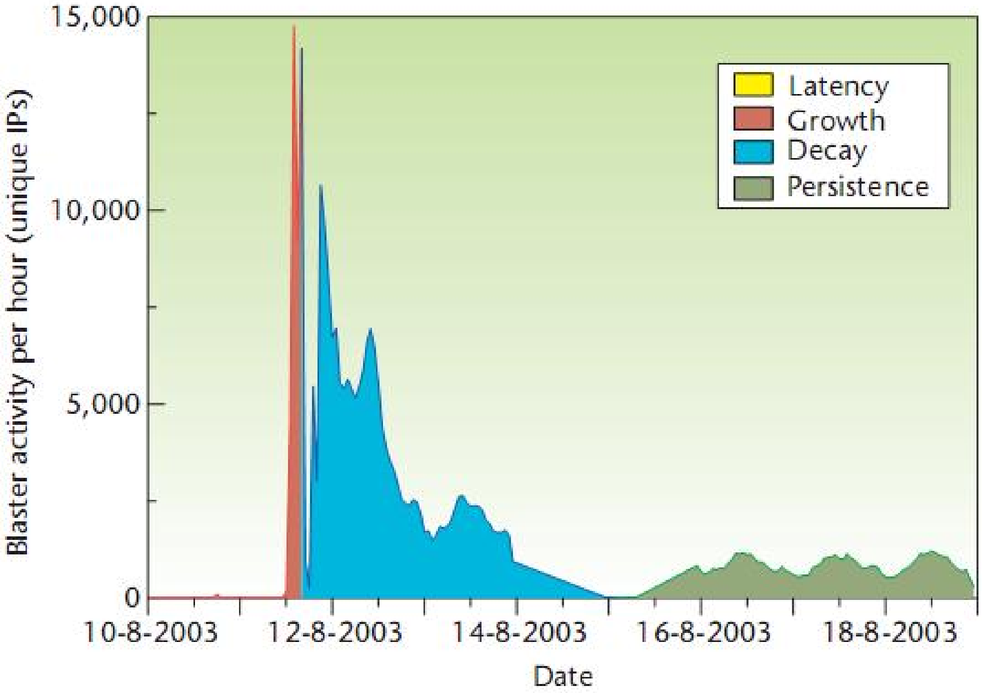


Figure 3.17: Blaster Life Cycle[6]

phase[6].

According to observations and analysis of Dubendorfe, Blaster had an outbreak on August 11, 2003. There were many infected hosts, approximately 200,00 Internet storm centers and 8 million hosts. Blaster was exploiting the remote procedure call which was a buffer overflow in Windows systems on port 135/TCP and it launched a DDoS attack against windowsupdate.com. It was observed that through the infection stage there were no response from target at 135/TCP and even if there was response port 135 was closed. The target were downloading the executable exploit but not the worm code[27].

On the other hand, Sobig.F had an outbreak on August 13, 2003. The worm was attached to the e-mail. The interesting point for Sorbig was that used its own SMTP engine to spread. Figure 3.18 describes the flow of the Sorbig worm. It can be observed an increase over 5 in e-mail traffic, during the outbreak, as it was monitored. Furthermore, 140,000 e-mails/hour were transmitted during the outbreak of Sorbig[27].

Nimda Nimda released on Sept 18, 2001. Had a multi-mode spreading and was attacking via e-mail and address books to IIS servers. It was sending copies across networks and was modifying sites and scanning for CodeRed II vulnerabilities or backdoors left[69]. Nimda worm, had a targeted detection that is not well researched, but follows approximately the model of 50% addresses with same 2 octets, 25% match with the first octet and 25% is randomly selected. Figure 3.19, describes the source scan-

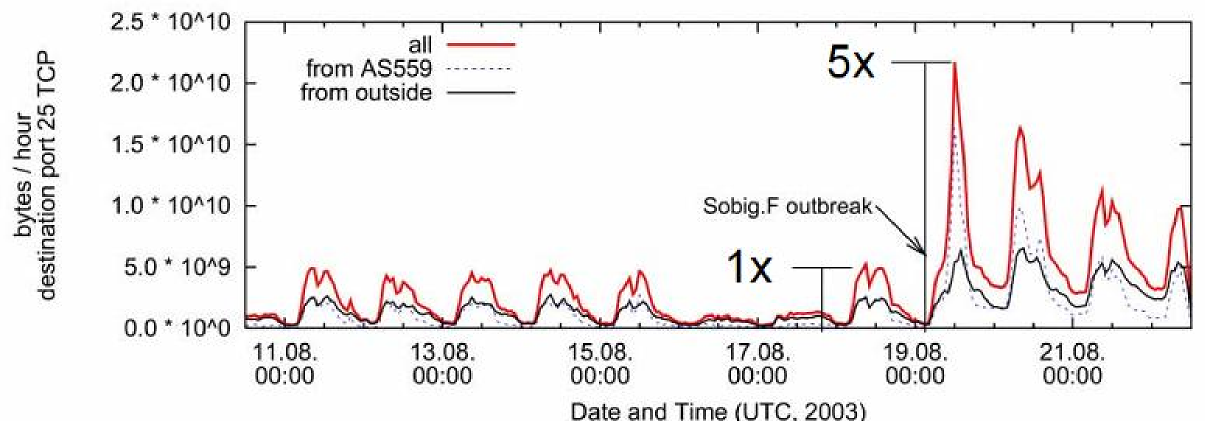


Figure 3.18: Sorbig flow analyis[27]

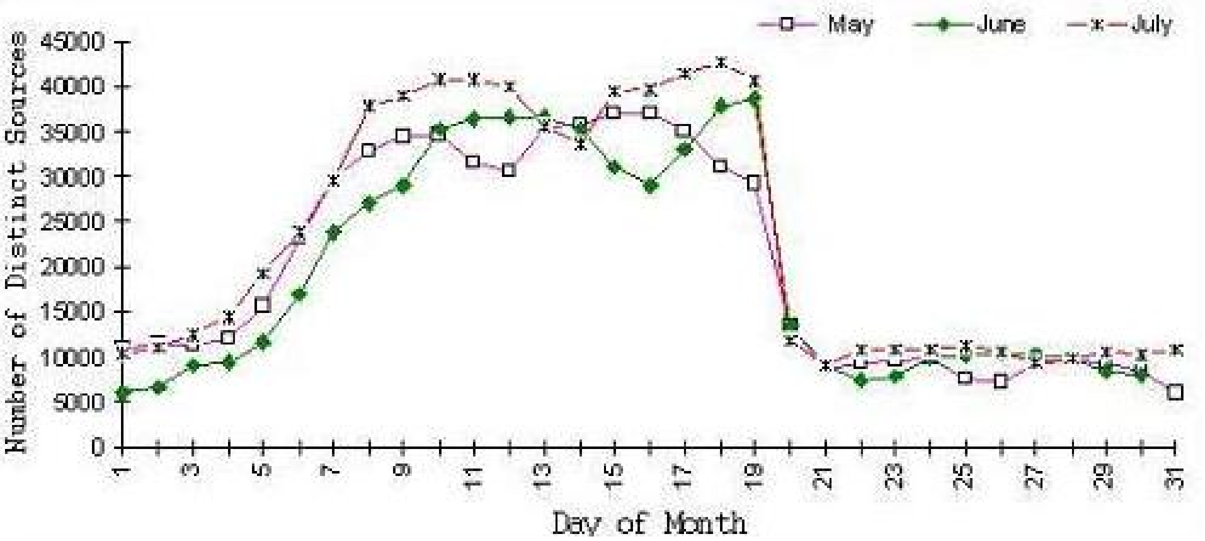
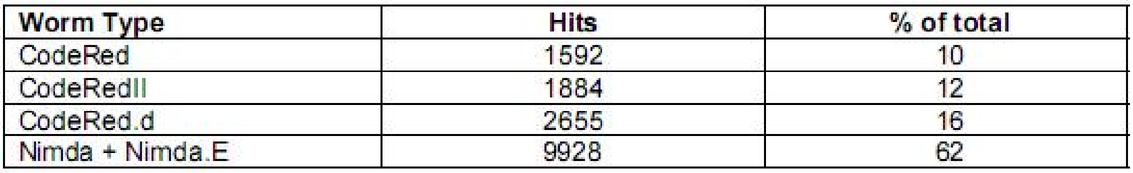


Figure 3.19: May-July 2002, Daily scans on port 80[88]

nings of port 80 in 2002. It can be observed the nature of port 80 scans. The drop in 19th of every month confirms the liveliness of Code-Red I. Also, it can be observed that vulnerable targets reinfected and start producing slowly at 1-7 days[88].

Another characteristic of worms is the persistent attacks. Through a research conducted by Yegneswaran *et al.* in a 3 month period on port 80 in /24 and /32, found that CodeRed II and Nimda had the affinity toward local victims[88]. On the other hand, Song *et al.* observed in a seven week period in 2001 with their network telescope 2,500,365,946 TCP SYN packets to nonexistent servers. The peak rate was 2000 hits/second. From the analysis of 16,433 hits was observed 5 fingerprints of worms. From table 3.5 can be examined that worms monitored are CodeRed I/II/.D and Nimda/.e. From the requests of signatures for each worm monitored Nimda had higher hits, because of the CodeRed's periodicity or timely death[82].



Worm Type	Hits	% of total
CodeRed	1592	10
CodeRedII	1884	12
CodeRed.d	2655	16
Nimda + Nimda.E	9928	62

Table 3.5: Infection Attempts at 1 over 100,000 sampling through network telescope[82]

Worm Propagation

In section 3.3 it is described how worms monitored through a network telescope. Discussion on various worms will be conducted through the network telescope point of view. In this section, it will be provided the most important epidemic model that can be used to examine the worm propagation and examination of worm propagation models will be used for network telescopes.

Traditional epidemiology commends that important for the propagation of a worm are the vulnerabilities of the certain population. Meaning the period and the rate of the infection. Additionally, there are three interfering phases of a worm: prevention, treatment and containment[65]. On the other hand, Yegneswaran *et al.* in order to categorize the propagation of worms divided scans into four categories. The vertical scan is the sequential or randomly scanning on multiple ports on an address during an hour. The horizontal scan is the scanning efficiency of a source combining the power of several machines in a network targeting the same port and the same vulnerability. The coordinated scanning(or distributed scanning), is scanning from many sources aiming a specific port on specific destinations in a certain /x subnet within an hour period. The stealth scanning is horizontal or vertical scanning in a frequency low enough to avoid detection[88].

In the case of a well designed propagation mechanism was observed that the initial acceleration with hit-lists of % vulnerable network hosts at 100 scans/second could be a vast infected population in just minutes. On the other hand, *Staniford* observed that with a hit-list of vulneranle hosts and propagation model well designed could propagate at 10^6 hosts in less than 2 seconds[69]. There is an observation though that worms prefer propagation through nearby addresses. Therefore, it can be effectively spread and increase speed due to local hosts and distances. Furthermore, hit-lists helps the propagation strategy through firewalls and security as it is observed through network telescope sensors. Thus, it is preferable to place network telescopes to nearby locations for effective monitoring and measurement of network activity[20].

From the monitoring of CodeRed was observed that in approximately a day propagated through 359,000 unique addresses. A further analysis showed that CodeRed was exponentially spreading and at its peak rate of infection, 2,000 hosts/minute were infected. Through the research, it was observed that 55% of the reverse DNS records didn't existed to the infected hosts and 22% only were manageable of identification[65]. Paxson on the other hand, through a network telescope, estimated that the number of

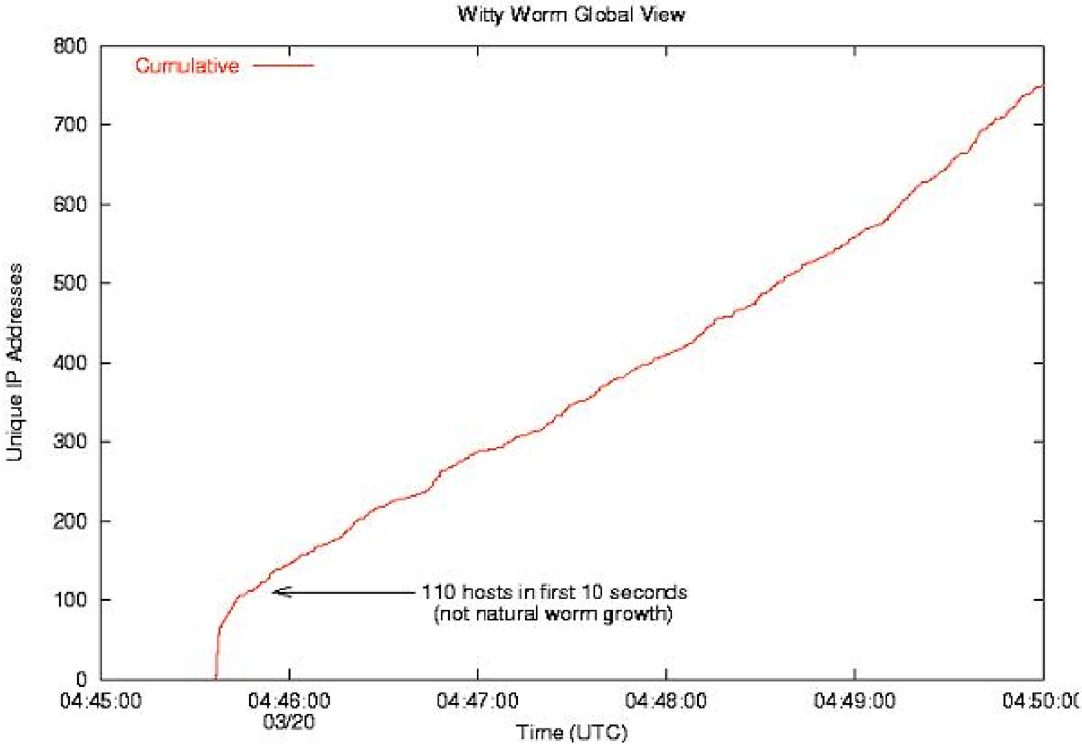


Figure 3.20: Growth of Witty worm[59]

infected hosts from the propagation of CodeRed was 360k[69].

In the case of Sapphire approximately 100,000 hosts in ten minutes successfully infected. On the other hand, the most destructive Internet worm so far was the Witty. It was the first one with destructive payload and was successfully wrote 64k blocks to the disk. The zero-patients or the first infected hosts were more than 100. The propagation even if the population used was small succeeded to be approximately 12,000 hosts in 45 minutes when reached the highest limit of infection. From figure 3.20, it can be observed that the propagation model was extremely fast. In only the first 10 seconds 110 hosts were infected[59].

Furthermore, Sapphire/Slammer worm behaved in the first 60 seconds as classic worm activity, but doubled its propagation approximately in every 8.5 seconds instead of CodeRed which doubled every 40 minutes. Additionally, it was observed that in more than a minute started to overload the bandwidth and started scanning more than 20,000 hosts/second. Also, it reached its peak at approximately 3 minutes and 55 million scans/second. Further, to the research and monitoring through a network telescope was observed that in less than 10 minutes 90% of the internet already scanned. Moreover, through figure 3.16 can be observed the infection rate. Actually, 100,000 hosts infected in ten minutes and reported a probing rate of 55 million per second[56].

Moore *et al.* monitored that more than 90% of infected hosts were infected in 10 minutes. By exploiting the vulnerabilty of SQL-Server, the worm infected approximately

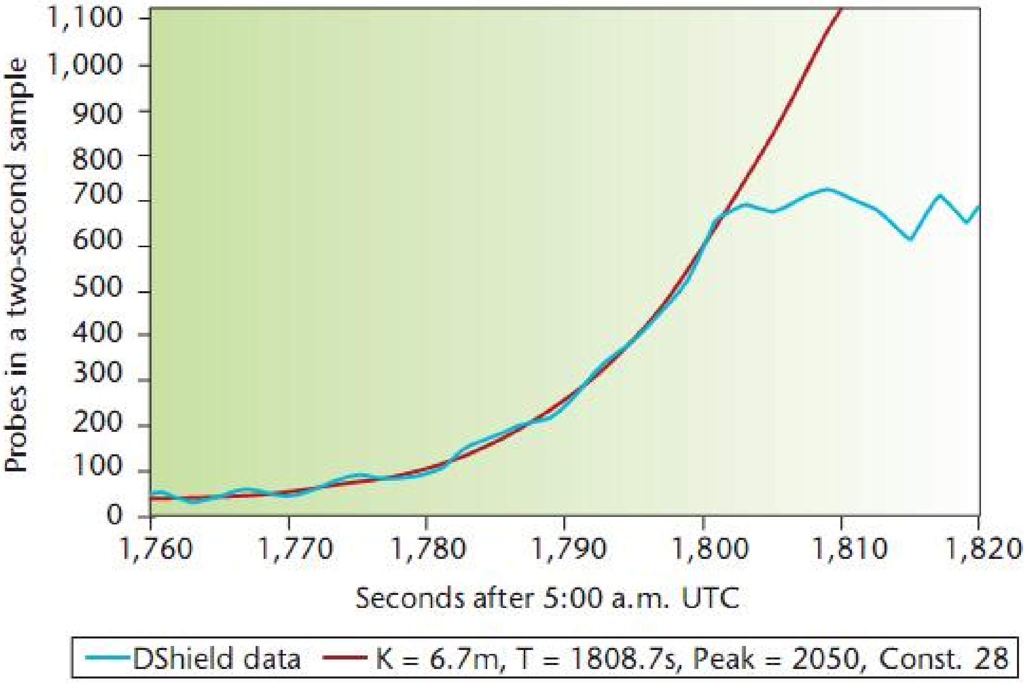


Figure 3.21: DShield project monitoring a random scanning worm[58]

75,000 hosts and caused severe problems especially to ATM machines. Its propagation model can be characterized as unique, because in 3 minutes approximately achieved its peak rate and the considerable amount of 55 million scans/second monitored. The worm propagation model of Slammer resided on randomly scanning. Through this characteristic the random scanning worms have the ability to propagate exponentially, but new infections slow the continuity of the worm, because Slammer retried infections[58].

Figure 3.21, describes the Distributed Intrusion Detection System data sets in comparison with a random constant spread model. As it can be observed the bandwidth overload and failures in networks produced changes to the growth rate of the probe[58].

Nazario, on his research for Blaster, in order to detect its propagation start detecting first port 135 of TCP scans. The network telescope used for this propagation modeling was a /24 network[66]. Furthermore, Cooke *et al.* on their research in order to test the propagation hypothesis of the Blaster worm isolated the signature. Blaster as a worm had simple propagation model based on sequential scans. The first step observed for Blaster was that choose local addresses on a /16 as source 40% and randomly selections 60% of the time. Then the propagation strategy was to scan sequentially with 20 attempts at a time. In figure 3.22, it can be observed the propagation attempts as monitored by a /24 network. Furthermore, differences in sensors can also be examined. There is a significant change in sources. However, spikes in I/17 conclude to that might be hotspots which cannot be correlated in the overall traffic[20].

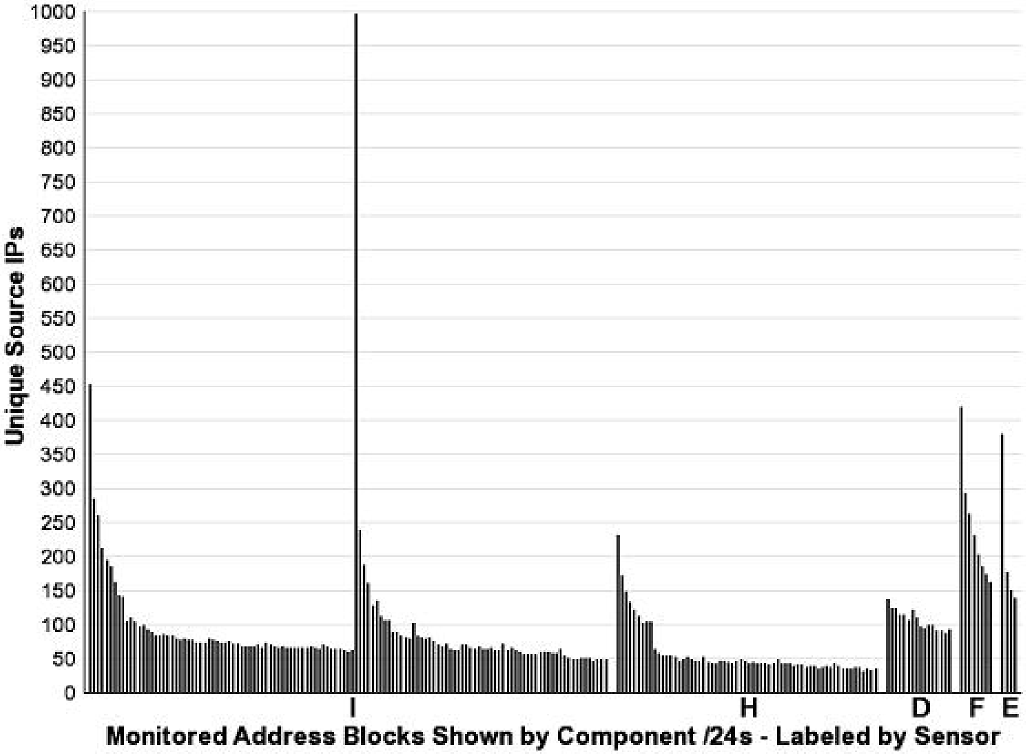


Figure 3.22: Blaster infection attempts recorde by a /24 and /16 network[20]

In addition, it was monitored that within a week Blaster infected more than 100,000 Microsoft Windows systems. The propagation strategy had a preference on local /16 network addresses. Furthermore, it was observed that the propagation model when was installed to a host was scanning for infection 20 sequential addresses on port 135/TCP. The propagation model of Blaster observed by a /8 unused space(network telescope) which represents roughly the 1/256 of the Internet addresses. Also, was observed that scanning from Blaster were 4-10 and increased at the day of patching mechanisms released at 100-300 per day. Additionally, was monitored that in the first few hours of propagation Blaster had been propagated exponentially and doubled in approximately 9 minutes. Also, Blaster examined through the network telescope and observed that 15,000 addresses in an hour and 106,000 addresses in 24 hours has been scanned. Further to the research of Blaster, the observations after one year on Blaster's release, showed that still was active. The network telescope monitored that the peak was 4,100 addresses per day[6].

Worm Propagation Modelling The SI epidemic model illustrates the growth of infectious pathogen propagating through similar random contacts between Susceptible and Infected hosts. Therefore, worms can be well described through this model. The description of infected hosts proportional at time t is: $i(t) = \frac{e^{\beta(t-T)}}{1+e^{\beta(t-T)}}$. This equation is well known and is applied by the health service to digital pathogens. In order this formula to be arranged base to worms, variables changed their meanings. Population

N is the Internet vulnerable hosts for exploitation. The *S(t)*, susceptibles are the vulnerable hosts, but not yet exploited. *I(t)*, is the infective; hosts that help with the propagation of the worm. β is the function of probing rate r and the algorithm in use for selecting new targets[65].

This formula, has the characteristic that small values of t helps the phenomenon to grow exponentially up to the point of vast infection and then the phenomenon slows exponentially up to zero as infections are completed. Moore *et al.*, on the other hand, assumes that hosts infected were chosen randomly such as CodeRed worms. Therefore, $\beta = r\frac{N}{2^{32}}$ because probing will reach with probability $\frac{N}{2^{32}}$ a vulnerable device. Hence, if β is fixed, *N,r* are inversely proportional. Furthermore, propagation of a worm in N population of hosts vulnerable to worm attacks at rate r is equal to the propagation of N devices probing at $fracra$ rate[65].

Staniford, in order to explain the propagating model of CodeRed, used the epidemic model. In the epidemiological model there are two states. The Susceptible host, is infected from other infectees while can recover and become Susceptible. Therefore, the epidemiological model is using the *susceptible*→ *infected*→ *susceptible* or SIS model. In order to describe the SIS model, the use of nonlinear differential equation for the measurement of infected population is: $\frac{dn}{dt} = \beta n(1 - n) - dn$, where *n(t)* is a fraction of infectees from the vulnerable hosts. β, is the rate in which infectees infect susceptibles and d is when infectee becomes susceptible. Hence, the equation is: $n(t) = \frac{n_0(1-p)}{n_0+(1-p-n_0)e^{-(\beta-d)t}}$, where p and n_0 are $p = \frac{d}{\beta}$, $n_0 = n(t = 0)$. Concerning random propagation models the rate β becomes $\frac{sN}{2^{32}}$, where N is the vulnerable hosts total number and s the scanning rate[16].

On the other hand the AAWP or Analytical Worm Propagation model, proposed by Chen and his colleagues, has the ability to model propagation of active worms of random scanning. This is a nonlinear difference equation model: $n_{i+1} = (1 - d)n_i + (N - n_i)[1 - (1 - \frac{1}{2^{32}})^{sn_i}]$. n_i, is the expectation of the infectees at time *i*. The AAWP model considers time that a worm occupies to infect a certain host[16].

Cooke *et al.*, in their research for propagation, tried to observe and characterize the botnet propagation. They tracked the propagation activity of bots. Also, they observed the scanning mechanism for backdoors left and vulnerabilities. Through this, it was examined the Internet Monitor Sensor and observed the activity of the backdoors[21].

On the other hand, Rajab *et al.* used another method for effective modeling and monitoring a worm. Their methodology has two invariant behavioral properties, *(i)* the spread by active scans on a certain address range searching for probable infectees and that the worm follows the pathogenic model of propagation in a fixed population. With these invariants they estimated the worm evolution and identify the first infectee, by monitoring the scan ordering and arrival times from consecutive scans arriving at the blackhole monitoring system[73].

In order to evaluate the accuracy of a telescope to monitor the sequence, they distinguished between the average time of infection and the time of detection by a network telescope of a new infectee and denoted as T_{in} and T_d respectively. Hence, they also included the possibility that a newly infected host will send a packet to the telescope before the additional infectees[73].

Furthermore, they considered the uniformity of worm scanning propagation with s host/scanning rate over vulnerable V population. Additionally, by the use of discrete time model and the model of Chen *et al.* concluded that n_i the infectees number at the i-th step is:

$$n_i = n_{i-1} + (V - n_{i-1}[1 - (1 - \frac{1}{2^{32}})^R]) \tag{3.1}$$

R, is the number of scans in totally by n_{i-1} infectees. The right part of equation 3.1 is the increase of infectees. Furthermore, in order to calculate time T_{in}, there is a necessity for one more infectee to be set one. Therefore, the second equation is:

$$(V - n_{i-1})[1 - (1 - \frac{1}{2^{32}})^{T_{in} s n_{i-1}}] = 1 \tag{3.2}$$

[73]

Therefore, by solving for T_{in} is:

$$T_{in} = \frac{\log(1 - \frac{1}{V - n_{i-1}})}{s n_{i-1} \log(1 - \frac{1}{2^{32}})} \tag{3.3}$$

Furthermore, in order to compute the α probability that one scan will reach the telescope at least is:

$$\alpha = 1 - (1 - \frac{M}{2^{32}})^{R_T} \tag{3.4}$$

[73]

Additionally, by solving for R_T, from equation 3.4 is:

$$R_T = \frac{\log(1 - \alpha)}{\log(1 - \frac{M}{2^{32}})} \tag{3.5}$$

Thus, $T_d = \frac{R_T}{s}$ where s is the average rate of scanning[73].

Moore *et al.*, in their initial research for the completion of a network telescope referred that a network telescope even if it depends on the proper analysis of single hosts events there must be also proper analysis for the pandemic incidents, because affect a large population[64].

Figures 3.23 and 3.24, describe the effect observed by a network telescope of host infection from a random propagating worm. Actually, it is compared the observed infections in a /8 and /16 telescope by simulation method. It was monitored 360,000 vulnerable hosts with 10 scans/sec from a CodeRed similar worm. The /8 curve on top of the actual curve and the /16 is distorted because of the logarithmic spread of the random worm propagation[64].

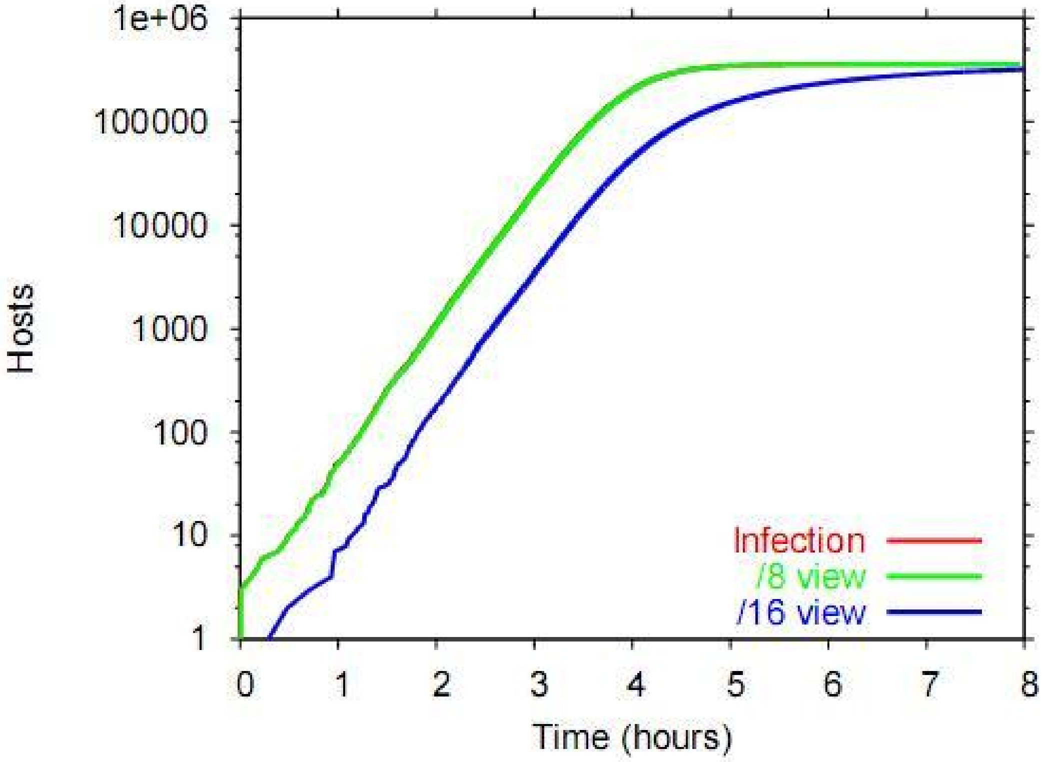


Figure 3.23: Logarithmic Infection[64]

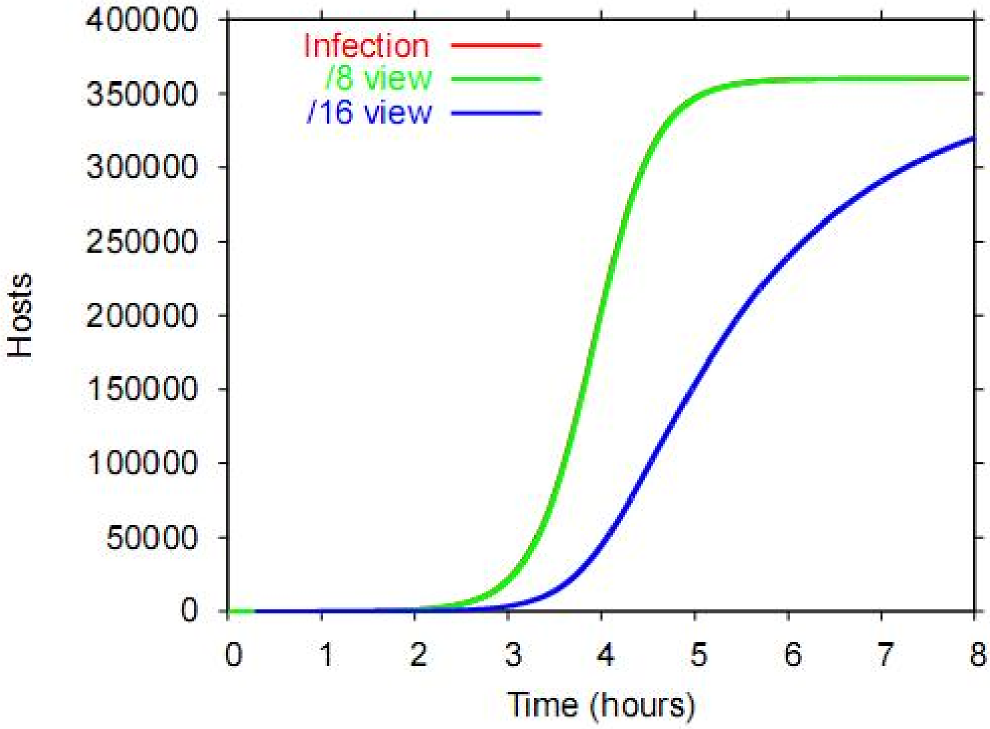


Figure 3.24: Linear Infection[64]

Firstly, it is examined the start and end precision times. The same principle as single host events lies here, even if if is not possible to measure exactly the precision times. However, time distributions and confidences can be used in order to determine time ranges for high probability of start and end times. For uniform incidents with respect to rates and times the calculation is fairly easy. The first probability distribution of the packet received from a certain event is the probability of the event. The same lies for the end for an incident. Furthermore, because incidents vary with rates diversity is considered to be caused by the characteristics or host specific incident characteristics[64].

In the case of random propagation worms, propagation can be computed through the epidemic SI model. Therefore, the equation for infectees at t time is:

$$I(t) = N\frac{e^{\beta(t-T)}}{1+e\beta(t-T)} \tag{3.6}$$

where T, is the constant integration of fixing time where half of the infectee population exists. Additionally, β which is the contact rate is the function of r scan rate of the worm over the targeting propagation algorithm. Hence, in 32-bit address the rate will be $\beta = r\frac{N}{2^{32}}$, because the scan will contact the vulnerable host with $\frac{N}{2^{32}}$ probability. Further, there are two approaches for tracking packets: count the packets arriving at specific times and the infectees addresses[64].

Additionally, the epidemic model insists that a host will continually be infected if only once infected. Also, the total infectees in time ($t_\alpha \leq t \leq t_b$), is the $I(t_b)$ infected at end. However, the scan number in total in a time is the sum of scans sent by each infectee at each time:

$$scans(t_a, t_b) = \int_{t_a}^{t_b} rI(t)dt = \frac{rN}{\beta}\ln\left(\frac{1+e^{\beta(t_b-T)}}{1+e^{\beta(t_a-T)}}\right) \tag{3.7}$$

Therefore,with a network telescope, equation 3.7 must be modified for proper scaling of the scan number arriving in /x network:

$$\begin{aligned} telescopescans(t_a, t_b, p) = p \cdot scans(t_a, t_b) = \\ = \frac{prN}{\beta}\ln\left(\frac{1+e^{\beta(t_b-T)}}{1+e^{\beta(t_a-T)}}\right) \\ = 2^{32}p\ln\left(\frac{1+e^{\beta(t_b-T)}}{1+e^{\beta(t_a-T)}}\right) \end{aligned} \tag{3.8}$$

[64]

Therefore, equations 3.6 and 3.8 are used for the effective data counts. So, $I(t)$ and t_{i-1}, t_i are used for scan counts. Recalling the β rate, scans t_{i-1}, t_i occur when *r,N* are together as a pair. Therefore, scan data by themselves do not provide useful information for the rates *r,N*[64].

Furthermore, Moore *et al.* correct the distortions from propagations which is used from the continuous SI model and describe the actual size of infectees at *I(t)* time.

Consequently, the expectation for observations with a telescope counting hosts is:

$$O'(t) = \int_{-\infty}^{t} I'(t-x) Prob\left[delay = x\right] dx \tag{3.9}$$

and because geometric distribution is needed, they compute *I(t)* from equation 3.9:

$$I'(t) = \int_{t}^{\infty} O'(x) Prob[delay = x - t] dx \tag{3.10}$$

[64]

3.4 Summary

In this chapter, in depth analysis of pandemic and endemic incidents through the functionality of network telescopes were examined. In DoS section was examined the way that these phenomena send unsolicited packets through a network. Further, description of unsolicited traffic and characterization of backscatter packets through different ways conducted. Also, this chapter examined backscatter models and how endemic incidents monitored from network telescopes.

Moreover, pandemic incidents were analyzed such as CodeRed I/II, Blaster, Witty, Slammer and SQL-Snake. Characteristics of each pandemic incident described and observe its propagation models. Additionally, the propagation models examined and how can be used for monitoring techniques.

Chapter 4

Topologies

4.1 Introduction

On previous chapters there was a discussion on various ways to monitor pandemic and endemic incidents. There was a reference to various mathematical tools and how researchers categorized pandemic and endemic incidents. Furthermore, there was an examination on various worms and detection methodologies and how these methodologies could be used for detection techniques on a network telescope. On this chapter there will be a discussion on various topologies implemented or existed for research purposes. Moreover, an analysis on various types of architectures such as passive and active telescopes will be conducted. Moreover, there will be analysis on the topologies on itself and not for the features that a certain topology used for effective reasons. Meaning that there will not be an analysis on virtual machines or gateway routers that these topologies use.

Furthermore, architectures will be analyzed. In case that whoever needs further information about containment technologies, virtual machines or gateway protocols, please refer to the research done and the documentation gathered. In addition, full analysis from the inventors and the implementers of these telescopes will be discussed. Additionally, there will be an analysis on how the researchers built the topology and what features they used.

4.2 Passive

One of the uses of a network telescope is passive monitoring. Through passive monitoring, a telescope can observe the packets and keep logs and discard them, but without interacting with the attacker[34, 77]. On the other hand, passive telescopes are useful for measuring the attack behavior especially for pandemic incidents such as worms[88].

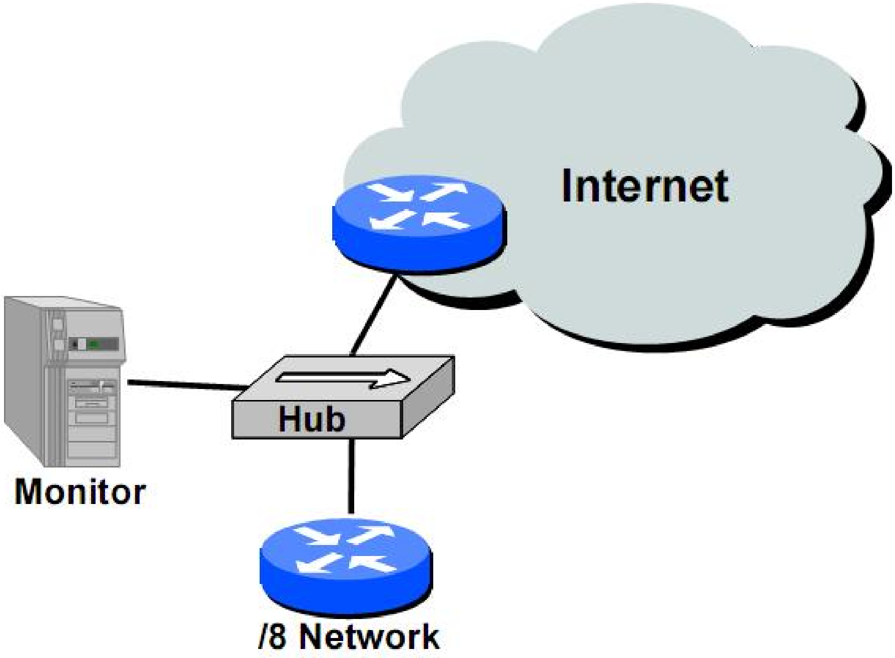


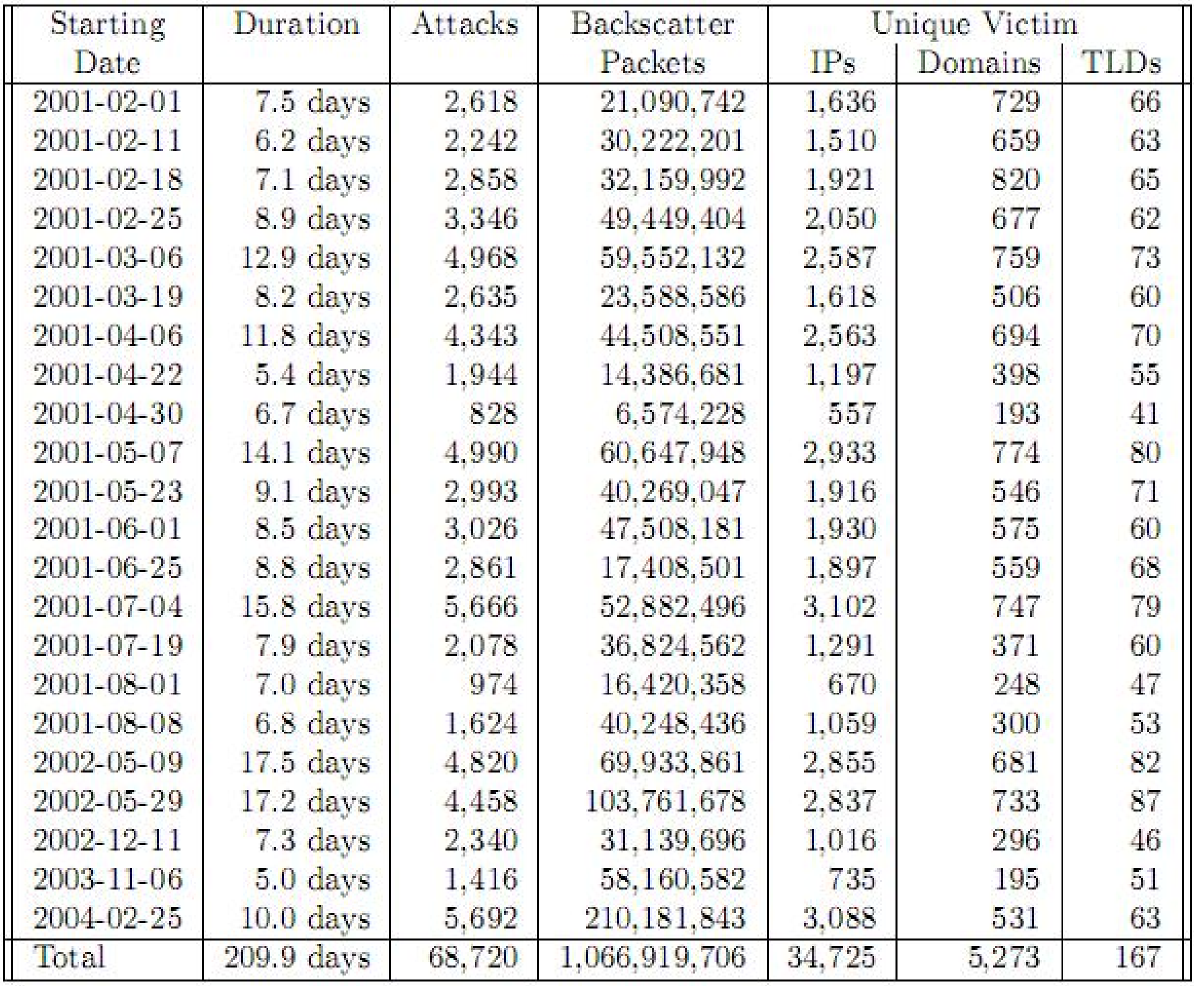
Figure 4.1: Passive telescope topology[61]

A passive network telescope records the observed packets and takes no further actions. By this interaction, network telescope will have the opportunity to observe hosts and packet information. However, information about attack or misconfiguration might not be revealed. For instance, if a network telescope will observe the three TCP handshake will not monitor it. It will monitor the handshake as an attempt for connection with the use of TCP[52].

In passive network telescopes, the headers and payloads received are analyzed offline for characterization of malicious traffic. Passive telescope characterizes traffic basis to protocol, number and sources type. Type, number of destinations and ports attacked through a period. Also, it can detect pandemic and endemic incidents such as worm payloads and backscatter data. Unfortunately, a passive telescope cannot identify attacks for instance exchange packets before malicious activity. Furthermore, it cannot detect activity if attacker or attackers disconnect or discard the attack quickly[34].

Furthermore, a passive telescope must be able to monitor packet information with high accuracy. Tools used or methods for packet capturing must be flexible and efficient in order the data to be monitored and logged properly[93]. For instance, one of the network telescope's implementations used by Sandvine to monitor real traffic and observe a DDoS attack[76].

Figure 4.1 is an experimental platform used for DoS analysis. The network was utilized /8 and consisted of a host monitoring ethernet traffic and was placed after a router where the network terminated. Furthermore, it must be mentioned that the router was



Starting Date	Duration	Attacks	Backscatter Packets	Unique Victim IPs	 Domains	 TLDs
2001-02-01	7.5 days	2,618	21,090,742	1,636	729	66
2001-02-11	6.2 days	2,242	30,222,201	1,510	659	63
2001-02-18	7.1 days	2,858	32,159,992	1,921	820	65
2001-02-25	8.9 days	3,346	49,449,404	2,050	677	62
2001-03-06	12.9 days	4,968	59,552,132	2,587	759	73
2001-03-19	8.2 days	2,635	23,588,586	1,618	506	60
2001-04-06	11.8 days	4,343	44,508,551	2,563	694	70
2001-04-22	5.4 days	1,944	14,386,681	1,197	398	55
2001-04-30	6.7 days	828	6,574,228	557	193	41
2001-05-07	14.1 days	4,990	60,647,948	2,933	774	80
2001-05-23	9.1 days	2,993	40,269,047	1,916	546	71
2001-06-01	8.5 days	3,026	47,508,181	1,930	575	60
2001-06-25	8.8 days	2,861	17,408,501	1,897	559	68
2001-07-04	15.8 days	5,666	52,882,496	3,102	747	79
2001-07-19	7.9 days	2,078	36,824,562	1,291	371	60
2001-08-01	7.0 days	974	16,420,358	670	248	47
2001-08-08	6.8 days	1,624	40,248,436	1,059	300	53
2002-05-09	17.5 days	4,820	69,933,861	2,855	681	82
2002-05-29	17.2 days	4,458	103,761,678	2,837	733	87
2002-12-11	7.3 days	2,340	31,139,696	1,016	296	46
2003-11-06	5.0 days	1,416	58,160,582	735	195	51
2004-02-25	10.0 days	5,692	210,181,843	3,088	531	63
Total	209.9 days	68,720	1,066,919,706	34,725	5,273	167

Table 4.1: Backscatter Database observed by passive topology[61]

filtering the traffic, but it did not had any impact on the data received. Additionally, with this simple architecture researchers collected data for backscatter analysis over a three year period. Furthermore, table 4.1 is a characteristic example of passive monitoring database which is a summary of traces and attacks observed. There were observed 68,700 attacks in 34,700 addresses over 5,300 DNS domains[61].

4.3 Active

An active telescope responds to incoming packets and tries to establish communication with the attacker. In the case of an Internet worm, a network telescope will continuously communicate with more than 10 messages, sometimes, until the worm is identified. Active monitoring is reliably distinguish attacks and can emulate a service and analyze attacks. Furthermore, it can keep tracks of the attacker and offers scalability. On the other hand, it can be resource-intensive and must be decided by the architect the responder's type. A stateful responder will keep each connection's state if it is active. Stateless responder will design an application response based on previous packets. Furthermore, filtering might be needed for effective monitoring of the traffic[34].

Active telescopes response to arriving packets and by this methodology information about application, exploit, attempt and attacker intensions can be collected. A response, for instance, can be a TCP SYN/ACK to a SYN packet. By this way complex attempts such as pandemic incidents can be categorized effectively. Furthermore, because of limited information more responses might be needed to resolve the attempt. Therefore, in order to collect more data creation of an emulated host can be realized. By the use of an emulated host observance of more data can be succeeded, since the active telescope has the ability to emulate an application or maybe a service. On the other hand, the emulated host can be identified and the attacker to avoid the unused space. Hence, the solution to this problem can be given by a host running real services and applications if needed such as a Honeypot. The honeypot will provide more information about a certain activity and will profile behaviors, intensions and purpose of the attempt. The simplest method for this service, is to install virtual machines on a real host[52].

In the case of iSink design, active response had the feature to gather information that is detailed from abusive traffic. This feature was applied by generating transport and application packets to the intrusion activity. Furthermore, few active telescopes have the ability for tarpitting in order to beneficially interfere with malicious activity[93].

On the other hand, Song *et al.* in order to measure the activity of a DDoS phenomenon used an active telescope and intrusion detection techniques offering a finer-grained traffic monitoring. Their information from a network telescope consisted of packet from application, traffic levels and worm fingerprints. The results of their monitoring can be observed from table 3.5[82].

4.4 Distributed Network Telescope

A distributed network telescope is the combination of telescopes for the purpose of monitoring different ranges of addresses into a large one. Can take the form of contiguous ranges such as a heterogeneous distributed system or an area of P2P networks[64]. Distributed telescopes belong to passive monitoring systems which are useful for measuring, especially, Internet pandemic incidents. For instance, from figures 4.2 and 4.3 it can be examined the daily observations from a Class B telescope. These observations made basis to scan rates for non-worm and 80/TCP port scans. On figure 4.2 can be observed that traffic has spatial components and can be monitored from a distributed set of telescopes. However, for figure 4.3 even a /16 network telescope can observe effectively the traffic and scan activity[88].

One of the distributed systems used today is the IMS. The Internet Motion Sensor has the ability to detect from distributed networks of /24 nets around the global

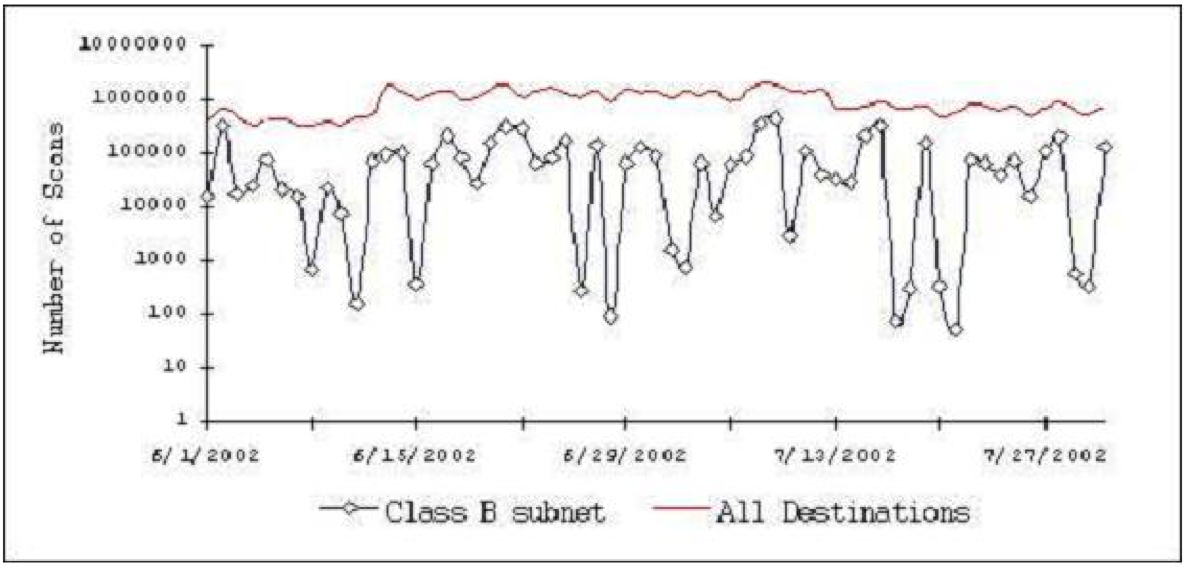


Figure 4.2: Daily rates of scans observed by a /16 telescope[88]

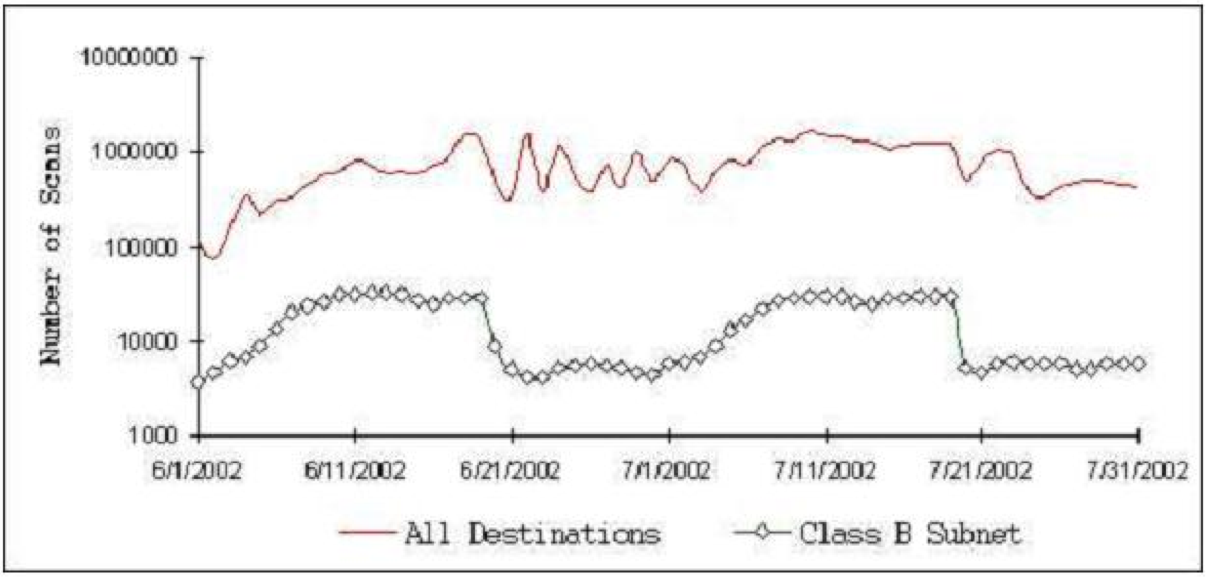


Figure 4.3: Daily rates from 80/TCP observed by /16 telescope[88]

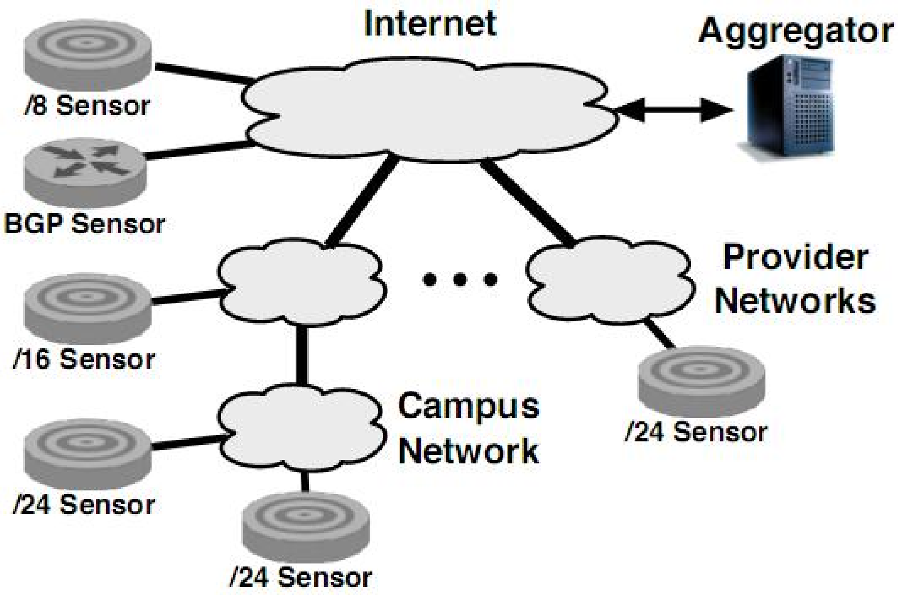
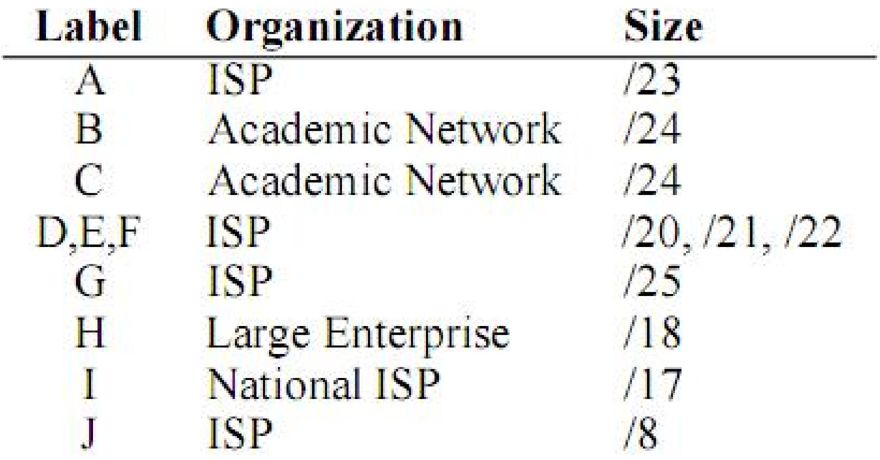


Figure 4.4: IMS[20]

network[38]. Consists of 60 network telescopes from 18 organizations, enterprises and networks from academia in 3 continents world wide. Furthermore, observes approximately over 17 million addresses[52]. Moreover, it was observed that an IMS sensor had a rate of 9 packets/second over 2.5 years. Additionally, it was observed that the lowest rate was 0.6 packets/second and the highest daily was 290 packets/second. Furthermore, the bandwidth that a sensor such as IMS is using is 7 Kbps for a /24 sensor, 60Kbps for a /16 sensor and for an /8 sensor 40 Mbps[52].

IMS was designed for interaction across all the network telescopes. Furthermore, it was chosen for interactivity and to differentiate and characterize the traffic. Additionally, it can provide a wide visibility for Internet incidents. As it can be observed from figure 4.4, IMS is consisted from various sensors and aggregators. Therefore, IMS sensors can be categorized to blackhole and topology monitors. The blackhole collects data from threats and topology monitors provide the appropriate information[20].

The blackhole sensors work on both active and passive state. Furthermore, the data storage of the IMS helps with the analysis and collection of various packet information. It supports real time trending and data analysis for processing, and because of the large amount of data gathered it can be an inefficient analysis. Therefore, for this purpose each sensor gathers the information needed and performs further processes and trigger alerts when is needed. Moreover, for efficiency reasons a sensor stores the MD5 checksum and compares it with the data arriving. When a new hash checksum is observed it is stored for future comparisons. From table 4.2, different deployments of the Internet Monitor Sensor can be observed. These deployments are representative sample of all /8 addresses. Furthermore, these sensors can help both comparison of data analysis through the Internet[20].



Label	Organization	Size
A	ISP	/23
B	Academic Network	/24
C	Academic Network	/24
D,E,F	ISP	/20, /21, /22
G	ISP	/25
H	Large Enterprise	/18
I	National ISP	/17
J	ISP	/8

Table 4.2: IMS deployments[20]

However, Sandin proposed that in a distributed system making use of Intrusion Detection Systems and use of Honeypots could be beneficial, referred in his research for effective measurement methods on peer-to-peer systems. Also, referred to Chord *et al.* who proposed that efficient routing in the case of faults and hosts might be possible. Furthermore, Sandin proposed that fault tolerant aggregation algorithms are efficient for the reliability of the sensors[75].

On the other hand, Bailey *et al.* proposed a hybrid model for distributed monitoring. As it can be examined from figure 4.5, the hybrid architecture consists from an IMS and a Host Motion Sensor(HMS). The IMS will be monitoring the range of addresses and the activity will be proxied to the HMS for in depth analysis of the incidents. Furthermore, in oder to avoid false positives and scaling issues combination with a pre-filtering installation could be examined. Additionally, the HMS can provide a forensic analysis depending on the data. It consists of a host resource, virtual machine and a detection module[5].

Zou *et al.*, on the other hand proposed another system for monitoring , the Malware Warning Center (MWC) which is based on the distributed topology. As it is observed from figure 4.6, the distributed system is using two monitors for effectiveness. The ingress monitors can be placed on routers locally or on passive topologies for logging traffic. On the other hand, egress monitor is used for monitoring outgoing traffic in order to infer potential behavior from worms. For ingress monitor it is really difficult to monitor properly global incidents, but for egress monitors is effective because of the scans sent outside the network by infectees. Furthermore, for effective analysis of a pandemic incident the distributed sensors must send continuously observations to the Malware Warning Center. Therefore, in order to avoid congestion of Internet activity data mixers are used. Data mixers, are installed between the Center and the monitors. Mixers after fusing(e.g. removal of unnecessary addresses from infected hosts) the data,

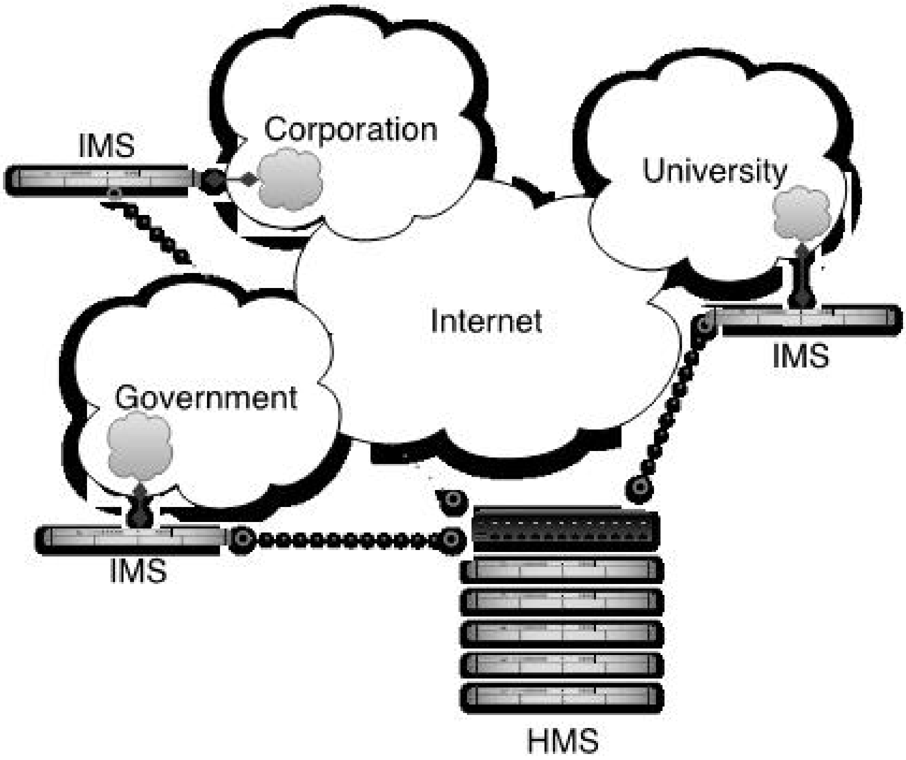


Figure 4.5: Hybrid Architecture[5]

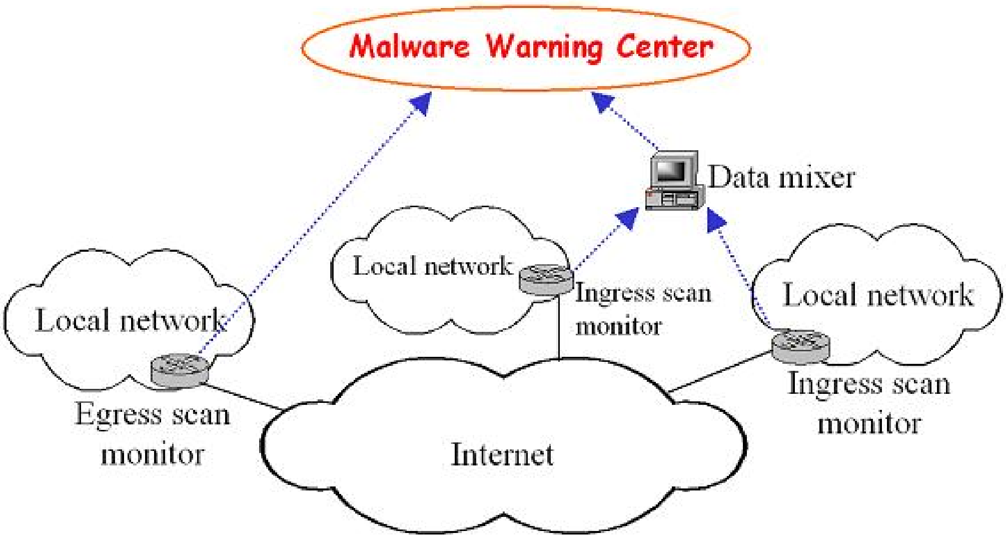


Figure 4.6: Malware Warning Center[95]

pass the information to the MWC[95].

4.5 Anycast Network Telescope

Anycast network telescopes are using multiple locations for the proper advertisement of routes at the same /x network. A telescope of this category does not monitor large ranges such as a distributed monitor, but share some of the advantages and disadvantages. By advertising many locations of /x prefixes, the telescope provides availability for monitoring effectively the traffic flow of events. Furthermore, this event traffic flow will be smaller because of the hosts' locations and telescope monitors in a /x network. Therefore, a telescope of this category will distribute the flow and load over many sites and the traffic event flow will be observed faster than the other models of telescopes[64].

Figure 4.7 describes the anycast topology of a network telescope. An anycast telescope, basis to McPherson *et al.*, allows the garbage of packets, effective distribution and management. Furthermore, anycast is a technique which is used effectively over DNS services, distributed telescopes, telescope routers for IP management and routing reasons. Additionally, an anycast telescope needs two different IPs. One for effective management and the other IP for anycasting. Moreover, because darknets are used to attract attacks, a proper placement in a network is required with proper integration and must have small impact on performance and availability of a network[53].

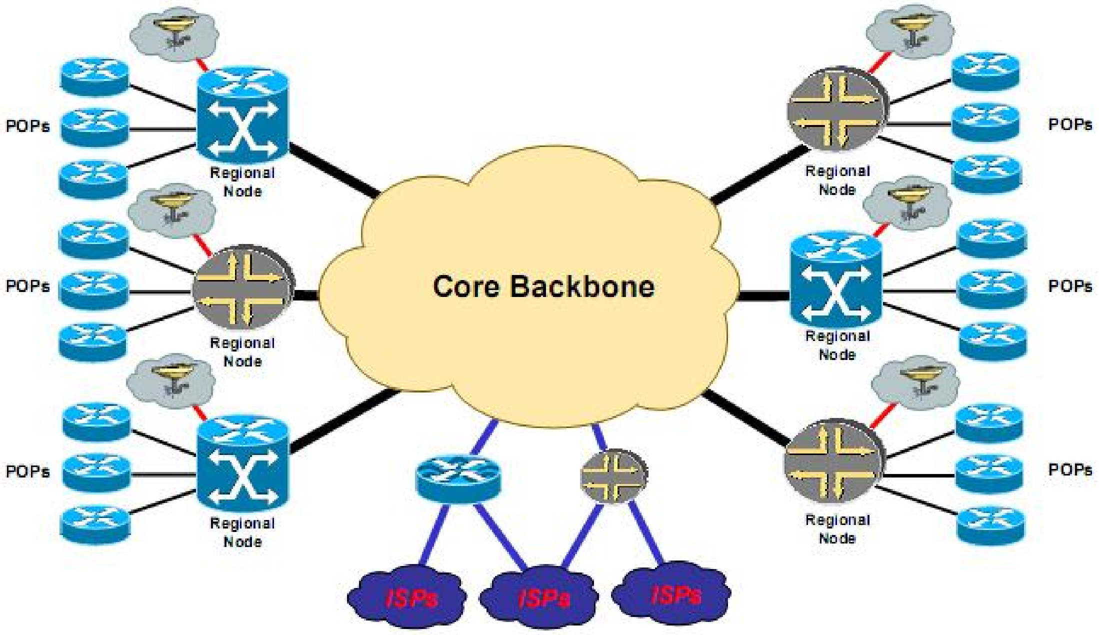


Figure 4.7: Anycast Topology[53]

4.6 Transit Network Telescope

Unfortunately, this category results found were only from Moore *et al.* from their technical report for network telescopes. They described that a transit network telescope monitors IP ranges, but from within the transit network and not from the edges of /x network. It observes large ranges of addresses and manages to monitor centrally and do not have synchronization and distribution problems. Furthermore, a telescope of this category can only monitor effectively IPs from the same network. Moreover, characterizing accurately events is not efficient with a transit telescope. Therefore, a transit telescope is efficient for the detection of occurrence of events, but cannot characterize properly in details. For instance, cannot describe the headers of the packets in detail such as a distributed system[64].

4.7 Honeyfarm

A honeyfarm network telescope belongs to the active topologies of telescopes. It actively responds to traffic basis to its features. Furthermore, it must be decided by researchers/administrators of the network the range of monitoring addresses. Additionally, for utility reasons a honeyfarm must observe high rates of traffic in order to avoid any correlation with the background flow. Moreover, the active responses can overload a network and a special consideration is needed when a honeyfarm operates[64].

On the other hand, a honeypot can run daemons or services that a sensor such as a network telescope can monitor. It can observe the activity and indicate infections. With the use of a honeypot, fault tolerance can also be checked. Furthermore, honeypot can be used with telescopes and monitor traffic and track configuration mistakes. The high tolerance offered by this model creates an ideal infrastructure[75].

A honeynet is a high interaction honeypot build to monitor information about security threats. Provides systems, applications, services that are real for interaction and provides emulation of certain operating systems and daemons(services). Furthermore, it can be considered a network of real hosts. The victims within this system can be resources of any type. In particular, can be an internet site, Solaris servers and even VAX systems. Honeynets, on the other hand, are not production systems. Hence, interaction with these machines implies malicious activity. The architecture of honeynet implies , also, a highly controllable network, because monitoring and control operations are efficient[70].

Figure 4.8, describes the architecture of the Honeynet. The honeywall gateway which is the key for the effective architecture separates the network and the honeypots, but it is invisible to an attacker. The key requirements for a honeywall are: Data Control, Data Capture, Data Analysis and Data Collection. The Data Control, defines the activity ratio which will be processed to the honeynet without the knowledge of the attacker. The Data Capture will capture the activity. Data Analysis is the ability of the system to analyze data for further results. Data Collection has the ability to collect information about activities from multiple sources or honeynets to a certain source. Data Control has the ability to mitigate properly the risk[70].

A Honeyfarm is a collection of honeypots monitored by a network telescope. Furthermore, it was observed that any outgoing traffic from a honeyfarm will be an activity from a pandemic incident. Additionally, it was examined that will be efficient if signatures observed from inbound and outbound traffic can be stored. Moreover, researchers by examining that the telescope's range is N addresses then there is an expectancy for proper detection after $\frac{1}{N}$ infected population[69].

Furthermore, one of the implementations of a honeyfarm is the HoneyTank project. HoneyTank, is a workstation accepting TCP traffic arriving from blackholes and replies through emulated services. Moreover, its features provides advantages such as simultaneous connections and characterization of vast amounts of data[87].

Figure 4.9 describes the architectural model of Collapsar which follows the Honeyfarm category. There is a number of high-interaction and virtual honeypots and are located in a local network. The honeypots in Collapsar are easily configured, monitored and are manageable. The end-systems or the routers used by Collapsar redirect the network flow to the Collapsar Center. Furthermore, it was observed that end-systems provide an additional delay to the packets arriving and routers require high manageability. Additionally, Collapsar has the ability to detect and stop propagation or backscatter phenomena[43].

Furthermore, Potemkin is a honeyfarm based on network gateway and a virtual machine monitoring the flow of network. Figure 4.10, describes a honeyfarm architecture. The

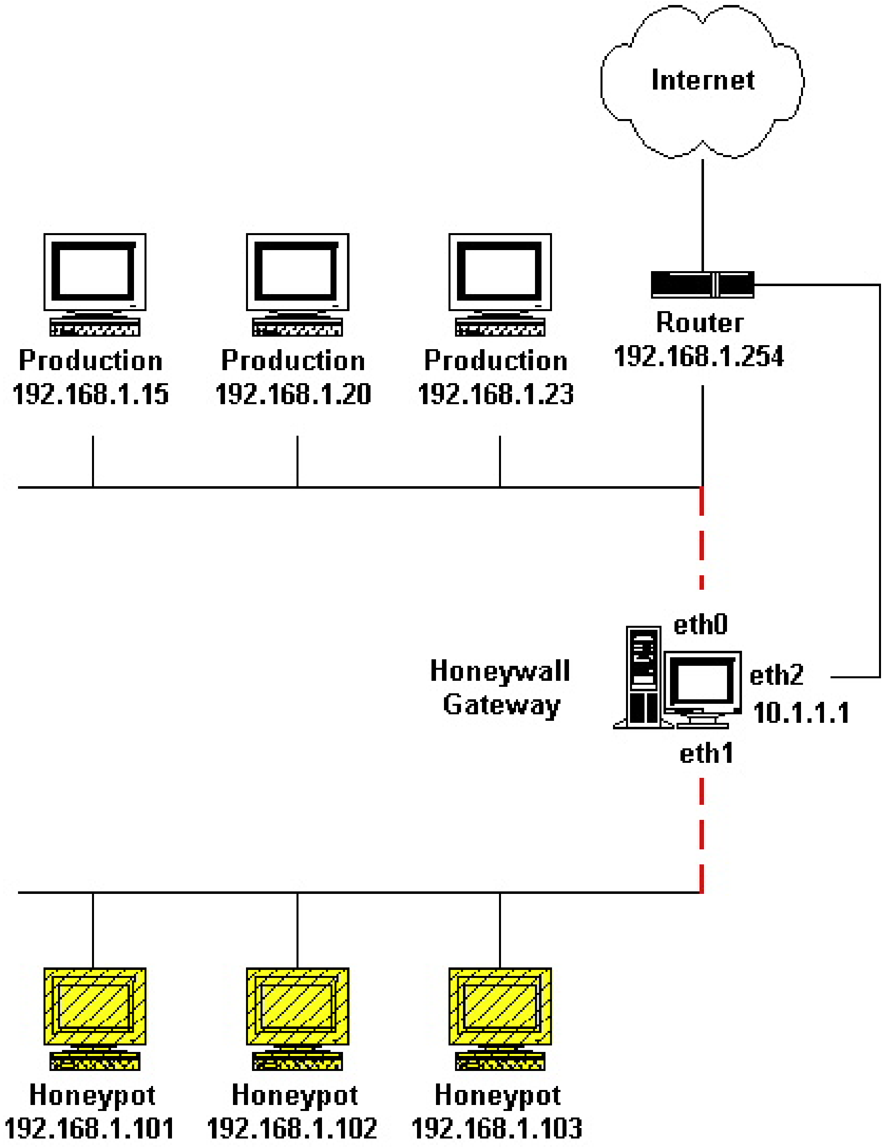


Figure 4.8: Honeynet architecture[70]

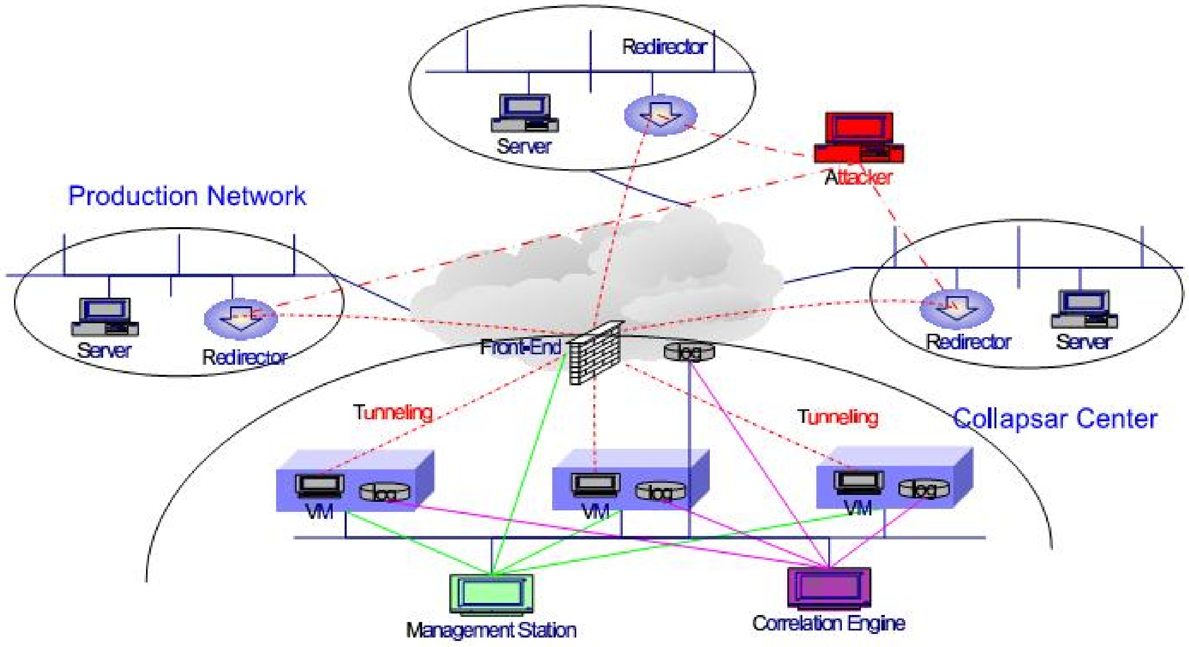


Figure 4.9: Collapsar architecture[43]

gateway redirects the packets to the servers and the honeyfarm servers create virtual machines for further interactions. The outbound traffic is under the control of the gateway containment policy[89].

Potemkin Honeyfarm, it is based on scalability and containment technologies. For scalability reasons, in order to avoid deficiencies, Potenkin stresses the latency of resources to requests. While packets arriving, a router binds the addresses to the honeyfarm servers for interaction. For the interaction process, the servers create virtual machines by creating the illusion of attached hosts. Moreover, for containment purposes, Vrable *et al.* implement containment policy to the router(gateway router). Therefore, the gateway must be able to track the communication between the IP addresses and the virtual machine created to the honeyfarm servers. Moreover, the gateway must proxy the outbound requests. Furthermore, the virtual machine monitor creates virtual machine for each distinct address which scans the darknet. By this measurement isolation is successful, but proper installment is needed or else it can be very costly[89].

4.8 Greynets

Harrop *et al.*, proposed another topology for a network telescope. A greynet is consisting of unused IPs and assigned ones. The network is sparsely populated with unused IPs and interspersing active addresses for effective traffic monitoring. The active IPs are assigned to hosts on the network. By interspersing unused IPs among active ones there is higher probability for the network telescope to observe a phenomenon such as

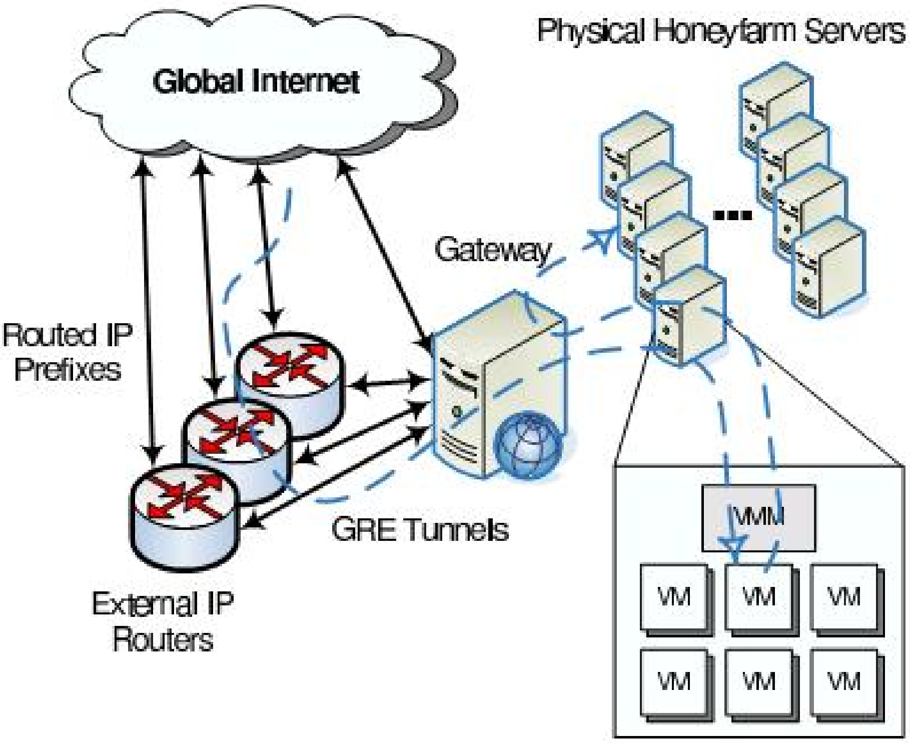


Figure 4.10: HoneyFarm Architecture[89]

malware attack[38].

4.9 Summary

On this chapter examination on various topologies of network telescopes conducted. There was an analysis on passive and active architectures and what features it can be used for effective monitoring. It was observed that passive topologies do not interfere with the traffic flow and just monitor for incidents and categorize them basis to the packet information. On the other hand, active topologies have the characteristic to reply on various incidents and track the user. In conclusion, there was also an analysis for greynets which is a combination of darknet and active IPs.

Chapter 5

Evaluation and Analysis

5.1 Introduction

The last part is the evaluation of network telescopes. Through this chapter there will be a discussion and analysis on the effectiveness of network telescopes. It will be analyzed the advantages and the disadvantages of network telescopes and the deficiencies observed through various experiments. Furthermore, there will be an opportunity for the reader to understand the theoretical part and thoughts of the researchers and how they concluded to various topologies and various algorithms for effective monitoring.

5.2 Effectiveness of Network Telescopes

A network telescope consists of unused addresses and no legitimate traffic exists in the monitored space. Therefore, since no legitimate traffic exists in a network telescope monitoring space, results from activity must be misconfiguration, backscatter activity, worm propagation or other type of network probing[52].

Network telescopes are effective in seeing large explosions of events and its effectiveness depends on proper statistical and mathematical tools[56]. Furthermore, Bailey *et al.* observed that small darknets sometimes can receive more packets/day than monitors which observe large IP ranges[52]. Moore *et al.*, added that a telescope's size is proportional to the size of the space monitored. Therefore, short and low intensity attacks generate less packets and thus the need for large monitoring space is required to resolve the information of the activity[61]. Furthermore, it was observed that the addresses generating data for the network telescope had differences in magnitude. Hence, there must be a mechanism in order to check if the data are manageable and can be generalized[20]. Additionally to what Moore and colleagues examined, it was observed that a /24 has a rate of 9 packets/second, a /16 75 packets/second and a /8 telescope approximately monitors 5,000 packets/second[52].

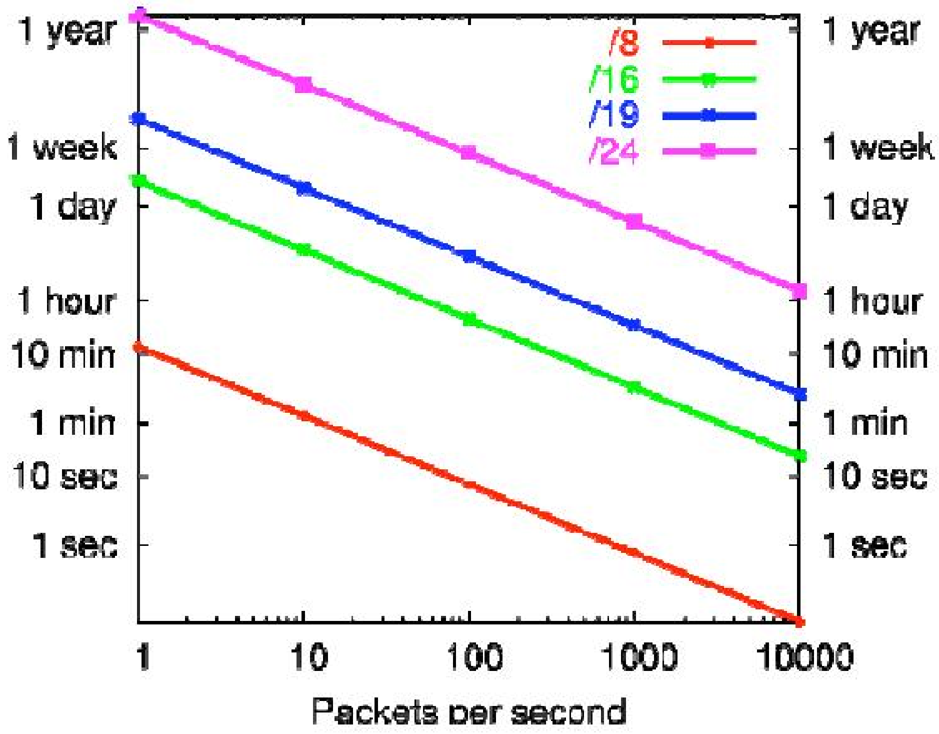
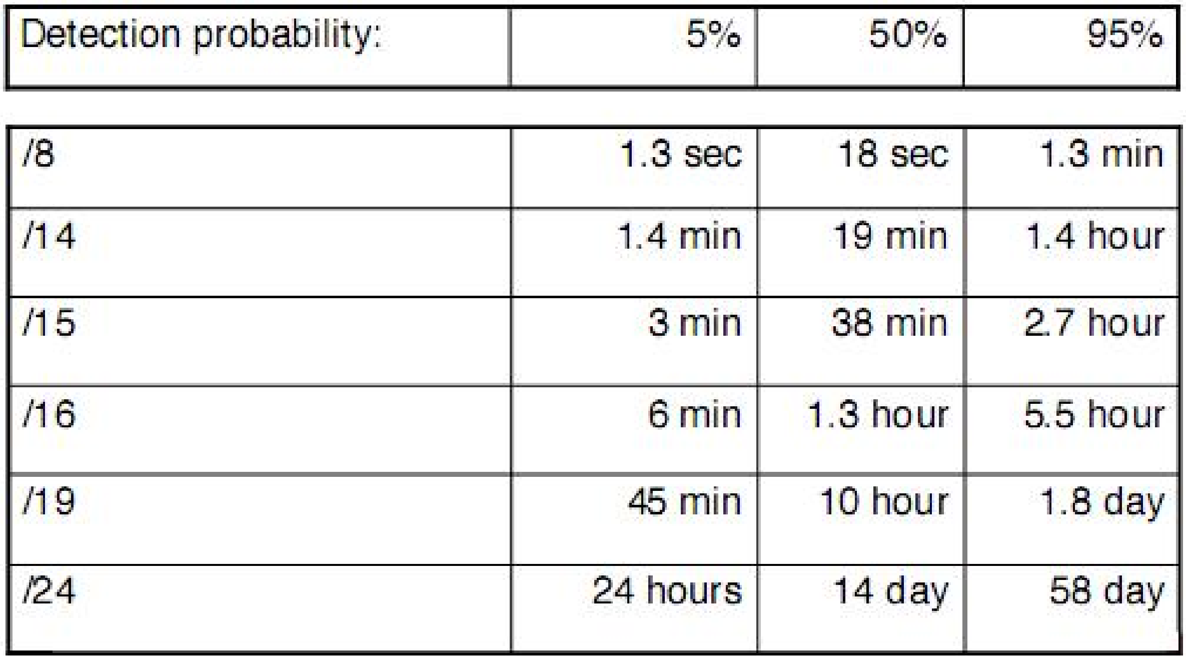


Figure 5.1: Detecting Events on different Network Telescope sizes[59]



Detection probability:	5%	50%	95%
/8	1.3 sec	18 sec	1.3 min
/14	1.4 min	19 min	1.4 hour
/15	3 min	38 min	2.7 hour
/16	6 min	1.3 hour	5.5 hour
/19	45 min	10 hour	1.8 day
/24	24 hours	14 day	58 day

Table 5.1: Detection probabilities on different network telescope sizes.[59]

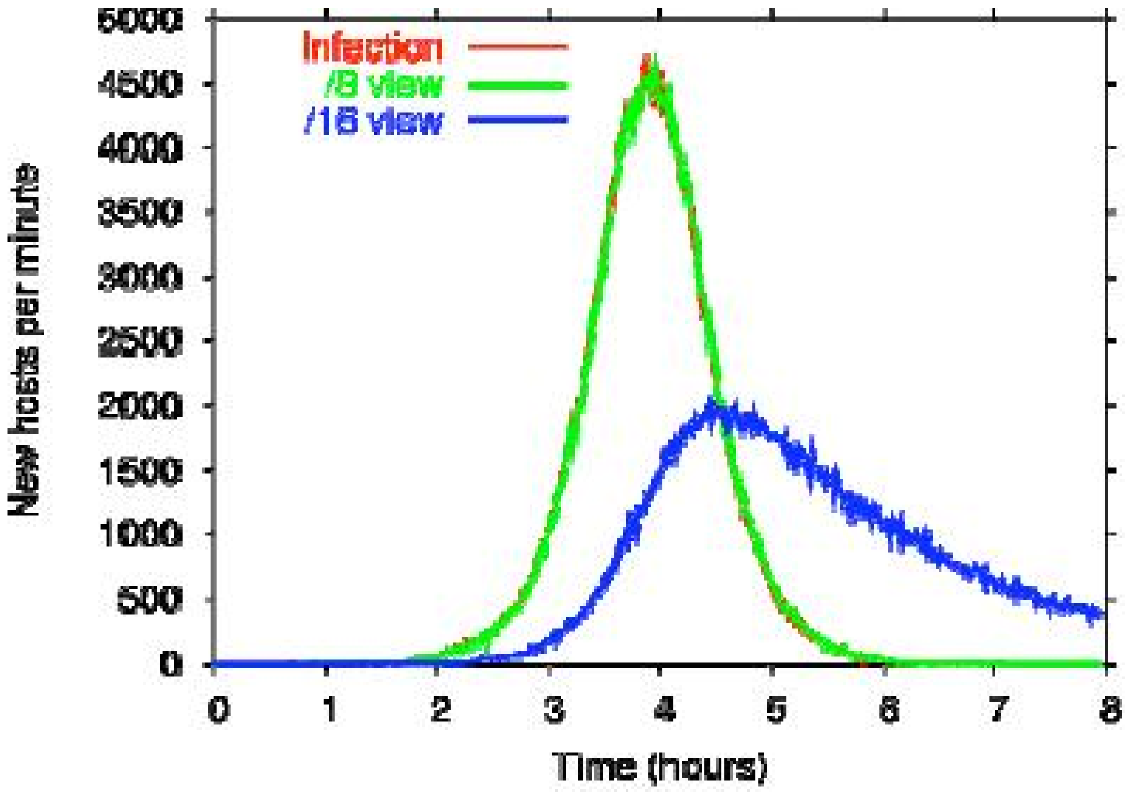


Figure 5.2: Worm Infection observed by two different telescopes[59]

Figure 5.1 and table 5.1 show the detectability of different network telescopes. As it is observed, the lines to the right offer approximately 95% probable detection. Furthermore, there are significant changes in detectability proportionally to the size of a telescope. A /8 telescope has 95% probabilty to observe an incident in 1.3 minutes, but for a telescope of /24 it is 58 days. Additionally, from figure 5.2 can be observed that a smaller telescope like /16 has less accuracy for determining incident times. On the other hand, a /8 telescope has higher accurancy to determine start and end times for the duration of an incident[59].

Furthermore, it is important to be understood that unused addresses must be globally reachable and stable, because router problems can effect the monitoring power of a telescope. In addition, resource constraints can determine the performance and availability of the sensors. For instance, if the network that the sensors rely upon is under DoS attack, the event could cause congestion and overloading of the network. Therefore, the visibility is affected and the traffic observed is not properly characterized. Furthermore, the statistical differences in various sensors arises from the sampling traffic observed and the results are different in every monitor. Hence, hypothesis testing for homogeneity can be used to resolve these matters[20].

Moore *et al.*, on the other hand, explained certain assumptions in order to emphasize that while selecting an address range for network telescopes, there is a variety of reasons that IPs must be selected randomly and uniformly. Often, in order to select addresses there is a bias to some regions, because some are heavily used instead of others. Therefore, there might be serious problems if the selection of IPs is not at random. Worms, in particular, spread with the help of nearby addresses. If the propagation is generated into ranges not observed by the telescope, the worm activity will not be monitored effectively. Hence, address range must be selected at random and uniformly. Additionally, the monitoring algorithm must be chosen carefully and under certain considerations or else if the selected algorithm is not considered to have the appropriate features for a certain telescope, the results observed will be for a telescope of complete different size and topology[64].

Another assumption that must be considered is the targeting rates. A network telescope has the tendency to underestimate the targeting reasons, because of the aggressive events that propagate and overload the network used by the darknet. Furthermore, there must be a consideration of the recorded events; how they sored, where and when. Furthermore, the data collection and analysis systems might have limitations to their capacity and processing capabilities. Therefore, in the case of an aggressive event at its peak, the storage and processing systems might collapse. Additionally, routing instabilities effect the results observed through a network telescope and traffic data might be lost, resulting to deficient analysis and classification. Lastly, because of the combination of pandemic and endemic incidents observed, the network telescope must have the ability to accurately categorize them. One kind of solution considered to the above problems is the distributed telescope which was analyzed to the topologies chapter[64].

Furthermore, it was observed that scanning and probing locally is of great concern than global activity, because internal hosts are already inside the security perimeters of the network[38]. Therefore, the vicinity of active hosts seems to be a crucial matter for a topological placement[52]. Hence, the topology of a network telescope always characterizes its monitoring ability. If a network telescope is placed behind a security device(e.g firewall or filtering machinery), then the visibility of a network telescope is decreasing since it cannot observe efficiently external traffic. On the other hand, telescopes can provide important monitoring activity inside a local network. Darknets that are placed inside and outside a network perimeter can monitor with further efficiency[52].

Additionally, Cuiy *et al.* emphasizing on the effectiveness of network telescopes, observe that passive analysis of network telescopes have limitations because of the passive monitoring and that do not interact with the attacker. By interacting with an attacker, valuable information for the source of the attack will be gained[23]. Dubendorfe, added that with passive topologies monitoring e-mail and multi-stage worms might not be efficient[27].

McPherson *et al.*, on the other hand, consulted that for efficient deployments in a network telescope topology, must be no leak of bogons. There must be no exporting policies to Border Gateway Protocol communities, also. Furthermore, there must be an explicit use of egress network policies. Additionally, no leak of traffic from the network telescopes is permitted, because if a backscatter traffic leaks from a Darknet it is possible to defeat the functionality of the sensor[53]. In addition, a system like a network telescope must avoid collisions with pandemic and related incidents, because relying on signatures of worms is dangerous. A new worm can create false positives and there is danger that the worm architect knows the systems. So, the sensors can create false positives and the architect will create pandemic phenomena that can avoid the sensors of network telescopes. Therefore, the deployment of telescope sensors must be under control in order the attacker not to monitor the location of sensors and in distributed environments the sensors not to be controlled by the attackers[75].

Network telescopes while observing the incoming activity collect the information needed for further analysis. This incoming activity can be large depending on the space that a network monitor is observing[52]. Therefore, the collection of compressed data that can be stored by a network telescope can sometimes surpass 30G/day. So, it needs efficient storage system and for evaluation real-time reporting services. It also must store packet information like headers effectively[59]. Moreover, a critical matter to the operation of network telescopes is the services running on the hosts either emulated or real daemons. This matter is very important, because of the decision taken it depends also the reactions of network telescopes to various incidents. Therefore, appropriate services running is complex and crucial for the visibility and interactions of a telescope[52].

For instance, in hybrid systems like honeyfarms, information received from unique threats are not clearly defined. Therefore, hybrid systems need intelligent systems for decision making. Meaning that, imperfect information must be categorized and the systems must collect the packets properly from the sensors[5]. Furthermore, the responder used by the IMS is a lightweight and there might be high probability, that information on application threats cannot be properly monitored[4]. On the other hand, devices like ingress filters and reflectors, can bias the results(e.g. rate estimations) of the attacks. Additionally, lost packets, machinery or network overload and limitations can bias the attacks rates and durations. Also, backscatter hypothesis can bias if unsolicited traffic send on purpose such as port scanning. Moreover, not all packets like RST/TCP are monitored by an endemic incident[59].

On the other hand, Honeynet allows the collection of specific information on various incidents. It obtains information, by allowing attackers to have privilege access[70]. This successful interaction is based on the use of Virtual Machine Monitors which allow the correct manageability of the Virtual Machines in order to be loaded or used on demand. Furthermore, offer a platform for the efficient instrumentation and monitoring of compromised systems. Therefore, the researchers/administrators can efficiently interact with the memory, disk allocation and system calls of the infectee[89].

HoneyTank, which belongs to the category of honeyfarms, showed an increased moni-

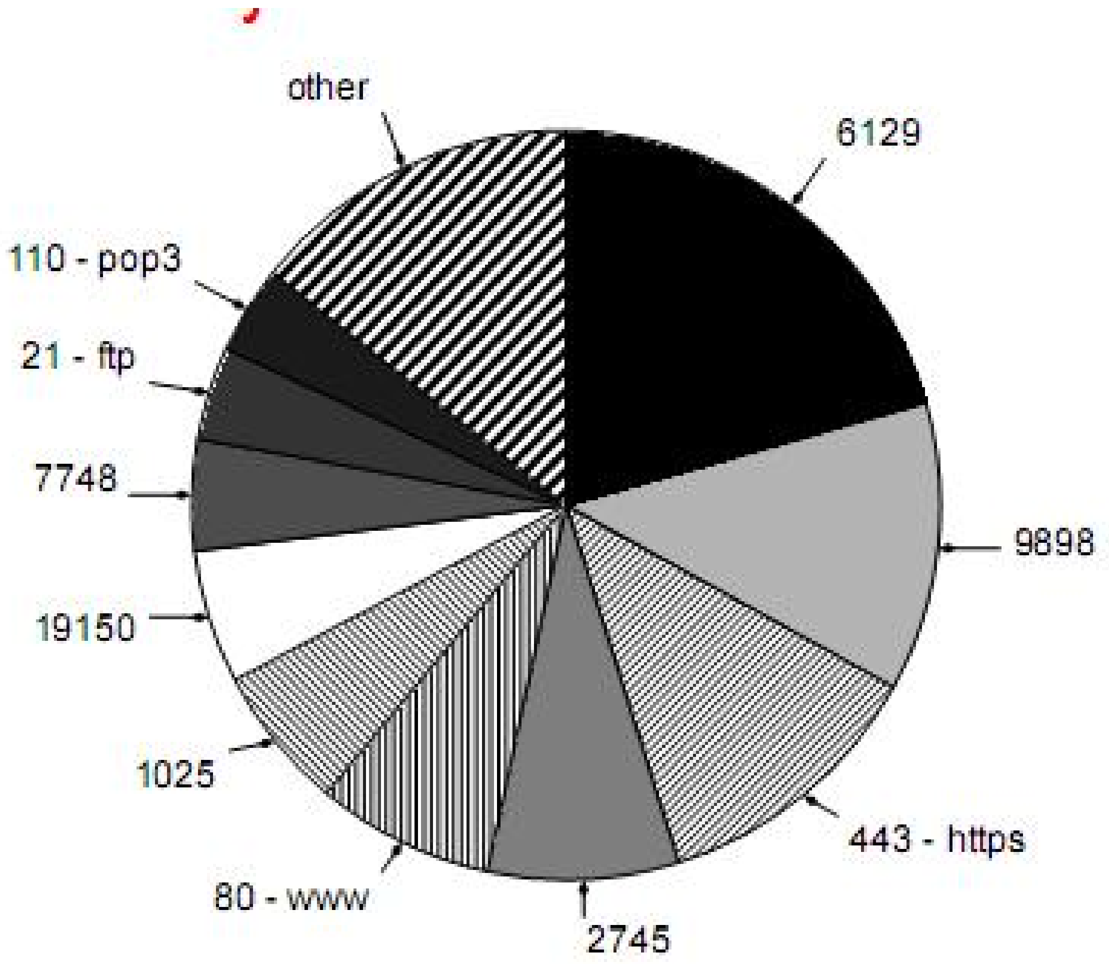


Figure 5.3: HoneyTank deployment traces[87]

toring activity even in experiments. Despite the firewall, HoneyTank had the ability to collect large amount of malicious activity. Figure 5.3, describes the targeted ports in traces from attacks. Even if through the experiments port distribution was changing, 80/TCP (HTTP) remained the top of the port preferences. Port 6129 , was mostly attacked, is associated with the Dameware Mini Remote Control Protocol. Port 9898, probably is the attack of Dabber worm and at port 2745 is considered to be the backdoor left by the Beagle worm[87].

On the other hand, Honeytank is easily detectable because of the emulation services and an experienced attacker can understand the system's features. So, there is a problem with the emulation. Therefore, emulation services must be configured accurately and maintain their states. However, while emulated services interact with worms can not be understood and so can interact effectively. Furthermore, as all honeypots, HoneyTank when is not maintaining a state is not vulnerable to endemic incidents. Therefore, a proper installation is needed, because if it is compromised by an attacker can easily be the amplificator for propagation or for unsolicited traffic[87].

Furthermore, Dark Oracle has a need for host configuration resources. Meaning that, because networks are complicated and sometimes there are hosts that do not allocated to databases, gaining access to hosts in environments such as Data Centers and configure devices for statistical data might not be efficient. Furthermore, it can be a misclassification of addresses due to instability reasons. Moreover, there is probability that the attacker can fingerprint the dark addresses and avoid them efficiently. On the other hand, Dark Oracle is resistant to fingerprinting techniques, because of the independent data sources used by the system. In addition, the researchers observed that a pool of addresses could be used in unused ports of TCP and UDP. Furthermore, it is observed that by sharing IPs to organizations and customers can be beneficial for the monitoring activity[19].

Additionally, Vrable *et al.* referred that with the use of honeypots in the Potemkin honeyfarm there is a possibility the attracted attacks from P2P, Internet messengers and e-mails to need further mechanisms for efficient monitoring. For instance, in the case of e-mails the system must be configured properly and examine the message for viruses or malwares. Furthermore, the attacker may attempt to detect the topology and the environment and modify the attack behavior. Therefore, a virtualized environment can offer camouflaging techniques and protect the system. On the other hand, a virtual machine cannot offer a full camouflage and further extensions. Therefore, the system must be available depending on the needs of the organization or the institution. However, an attacker with extended knowledge of the honeyfarm topology can violate the system. Consequently, a detection and Virtual Machine policy is needed in case the system is compromised. If compromised, the policies must define the lifetime of Virtual Machines and how the system must interact on those occasions[89].

5.3 Summary

On this chapter, there was an analysis on the effectiveness of network telescopes. It was analyzed the deficiencies observed through research and how these deficiencies can be solved. Furthermore, from the evaluation of network telescopes, it can be understood that proper scientific implementation is needed for effective network telescopes.

Chapter 6

Conclusion

Network telescopes observe through the help of unused addresses and characterize the traffic flow of a network by monitoring techniques. Special algorithmic models are used in order a darknet to categorize the activity of various incidents and misconfiguration packets. Through this report, the effectiveness of network telescopes was analyzed to observe pandemic and endemic incidents, meaning DoS and worm attacks, which are the most crucial phenomena for the liveliness of a network and its resources.

As it was examined, researchers categorized telescopes basis to their ability to monitor incidents on various networks. Therefore, a /8 network telescope has the ability to collect vast amount of information and thus has extended visibility. Therefore, as it is understood, a network telescope depends on the address ranges that monitors. Hence, a smaller telescope will process less data for classification than a /8 network. Moreover, basis to the range of addresses observed that a telescope has a certain possibility to detect phenomena that might overload resources or the network.

Furthermore, network telescopes make use of complex mathematical tools in order to categorize an incident. For unsolicited traffic of an endemic incident, backscatter analysis is used in order to monitor and extract packets, having the ability to categorize the packets also. On the other hand, researchers use the SI or the AAWP models for propagation modeling of pandemic incidents such as CodeRed worm and Blaster.

Moreover, researchers extended the models of network telescopes and implement various architectures. Through the various architectures, the ability of a network telescope was examined to observe packets passively or actively. For passive monitors such as distributed, anycast and transit network telescopes , it was observed that telescopes have certain features on how to categorize the activity without interactions. On the other hand, the honeyfarms were also introduced which have the ability to interact with the attackers and conclude to proper results, about the categorization and characterization of the attacker and of the incident.

Also, for the hybrid systems, it was mentioned on previous chapters that several research problems must be solved for successful operation. Filtering interactions, for instance, observe a vast amount of traffic which must be redirected to the honeyfarm.

In order to redirect data to the honeyfarm, data must be reduced. For the emulated hosts, the problem is that hosts can be fingerprinted. So, there must be a highly sophisticated and complex system in order to avoid fingerprinting results. Furthermore, the need of forensic techniques might be crucial, because generated actions must be categorized properly. There must be a categorization of signatures and behaviors that characterize the traffic in a network, also. Additionally, managing virtual machines with efficiency is difficult. VMs need efficient management, because it is expected to interact with several requests[5].

From the research conducted so far it was observed that network telescopes have several advantages and disadvantages. The advantages are that network telescope can be easily deployed, with proper algorithm can be used to all infrastructures and deliver a certain quality of security. A quality of security can be delivered especially for pandemic and endemic incidents, because these types of attacks generate vast amounts of information and can only be categorized through the information received and by proper use of specific algorithms. But network telescopes, on the other hand, have many deficiencies to certain areas. In particular, the algorithms can be too complex for an organization, business, enterprise or institution to work on these models, except if they have already researchers on this field. The implementations can be costly, if for instance an enterprise needs a honeyfarm.

Furthermore, network telescopes depending on their range of monitoring can conclude safe results. For instance, with a /8 telescope, you can have extended visibility, instead of a /24 telescope. Additionally, special sensors are needed or lightweight responders to interact with the attackers. Also, specialized softwares such as Virtual Machines or custom applications for the implementation of a proper telescope are needed. These machinery can result to enormous amount of money and for efficient economics the idea of a telescope cannot be properly implemented. Moreover, as it was observed, telescopes are mostly used among academia and for research purposes to detect various phenomena. As it was mentioned before, CAIDA for instance, was the pioneer for the detection and proper analysis of CodeRed propagation phenomenon.

From one point of view, the implementations and the algorithms must be used for efficiency and for proper characterization and categorization of the traffic. Furthermore, specialized equipment must be used for extended results. From another point of view, network telescopes can be simplified for extended use on ISPs and for organizations or enterprises that have an extended network and already compensate a certain amount of money for proper infrastructure. So, network telescopes need further research in order to be simplified but not loose efficiency and usage. Additionally, small telescopes have visibility problems which must be solved, for efficient implementations. For instance, if a company needs to implement a small network telescope as additional security their network must have a lot of spare resources in order to characterize a darknet of /8 network. Consequently, a small telescope observing the traffic to main servers can be extremely useful and must generate results and conclusions at the same rate as a large telescope of /8 network.

Furthermore, it was observed that topologies from passive to active are becoming complex and many parameters for proper implementation must be considered. Parameters like Virtual Servers, virtual hosts, services, interactivity, sensor's implementation and sufficient information needed, complex systems in order honeyfarms not be compromised and architectural complexity such as, "where the sensor could be placed". On the other hand, there is not an extended research to avoid complex matters and always the researchers mention that even a machine hosting a virtual machine can be compromised or an attacker can fingerprint the topology and the sensors. Hence, there must be an extended research for honeyfarms and virtual machines in order to avoid efficiently fingerprinting techniques.

Moreover, as it can be observed in topologies, there was not an extended research for certain implementations such as transit and anycast telescopes. Furthermore, there is only one distributed system which monitors the activity on the Internet. This deficiency, it could also be researched in order the academic community to have different results from same topologies. For instance, the academic community can observe the propagation model of a pandemic incident with greater visibility, if there are two distribute systems correctly implemented, one in Asia and one in CAIDA organization. With this implementation researchers might have the possibility to observe incidents, in particular incidents such as CodeRed, the beginning of the propagation without delays and approximations in durations and start/end precision times.

Furthermore, it was observed that researchers experiment through emulations, meaning that they implement a network telescope and emulate its model for pandemic and endemic incidents. Of course, in some occasions the telescope is already implemented and detects with efficiency all the incidents. From another point of view, researchers must experiment with real case scenarios in order to have safe results. Particularly, nematodes(beneficial worms) can be used in order a network telescope to monitor and analyze information from real occasions. Furthermore with nematodes, which are not referred to this project, a scientist or an administrator can observe the deficiencies in their network and examine the visibility of a telescope. Hence, this model can be used to small network telescopes. If nematodes are implemented in small networks using small telescopes, the supervisor of this implementation can monitor, for instance a month's data, traffic, various information and store them, and intelligent implementations compare the data of one month with the data observed real time. This implementation could help small telescopes to monitor and analyze data, and conclude to safe results faster than before. Of course, this implementation can help also all types of telescopes.

Nematodes, beneficial worms, have the ability to scan and analyze the vulnerabilities found on a network. Therefore, if a network telescope observes the activity of a nematode and store the data, the darknet can compare the results with real time occasions. In particular, if a network telescope can monitor the activity of beneficial worms, store the data observed through the propagation strategies, analyze it and result to conclusions, there might be a possibility a network telescope to increase its visibility and effectiveness. Furthermore, network telescopes might increase effectiveness because of the monitoring results, meaning the activity observed can be used for comparing this information with the primary results of incidents in a network. For instance, if the network telescope observe a small spike through the activity, the administrator with the help of intelligent systems can compare the activity observed and the activity monitored in one minute. Therefore, there must be a possibility that this spike is an incident. So, the administrator, in order to have safe results from this activity, can activate defensive mechanisms and isolate crucial parts of the network, to avoid catastrophic events. Moreover, nematodes can be used in conjunction with network telescopes and researchers can observe the ability of nematodes to cast away pandemic incidents. Furthermore, nematodes and telescopes can be used in a different way. When a telescope monitors an incident, it can send an alarm to the administrator for information, and the telescope through intelligent systems compare the results. Consequently, the network telescope after comparison will decide whether nematodes can be released or not.

In conclusion, nematode (beneficial worm) is a controllable worm which can be used for network protection. Furthermore, it can search the network for deficiencies and offer efficiency to the infrastructure. Lastly, nematodes named by primitive worm-like organisms, that often used to cast aside other epidemics[1].

Appendix A

Abbreviations and Glossary

NOTE FOR THE READER
For the effective usage from the part of the reader, Abbreviations and Glossary are combined. After the abbreviation like IP, the user will find the full explanation of the word like INTERNET PROTOCOL and after that, the description of the word

A

AAWP, Analytical Worm Propagation model: has the ability to model propagation of active worms of random scanning[16]

Active Telescope/Monitoring: An active telescope responds to incoming packets and tries to establish communication with the attacker[34]

Anycast Telescope: use multiple locations for the proper advertisement of routes at the same /x network. A telescope of this category does not monitor large ranges such as a distributed monitor, but share some of the advantages and disadvantages[64]

Address (Netwrok or IPv4 address): IPv4 addresses are logical addresses consisted of 32 bits long and have 256 possible combinations, but 0 represents the local address and 255 the broadcasts[85]

ACK, Acknowledgment flag: a control character indicating that a packet has been received without an error. In certain net- work architectures, ACK is used for a frame that sends such an acknowledgment[29]

B

Backscatter: is the excess of DoS attack. This excess derives from SYN flood which is a stream of TCP/SYN packets sent to the target. When attacker sends TCP/SYN packets from spoofed sources, the SYN packet received from the victim is for synchronization on a new connection. If the source was not found, the target will try allocation of new structures for the connection. So, if the target replies with SYN/ACK becomes backscatter[34]

BGP, Border Gateway Protocol: In the Internet TCP/IP protocol suite, a protocol for routing packets between networks that use different protocols. This type of protocol is known as an exterior gateway protocol (EGP)[29]

C

CIDR, Classless Interdomain Routing: a routing strategy that was developed as a partial solution to two difficulties that have developed as the number of networks connected to the Internet has grown very large.[29]

D

(D)DoS, (Distributed)Denial-of-Service: the attacker spoofs IP addresses randomly and floods the targets with requests. The target responds believing that the sender is a legitimate user[59]. Distributed attacks rely on this technique, but also consumes network resources and relies on large hosts assistant[4]

Distributed Telescope: A distributed network telescope is the combination of telescopes for the purpose of monitoring different ranges of addresses into a large one. Can take the form of contiguous ranges such as a heterogeneous distributed system or an area of P2P networks[64]

DHCP, Dynamic Host Configuration Protocol: On a TCP/IP-based network, DHCP is used to get information about a client host's (i.e., a network node's) configuration from a DHCP server, which is a specially designated network node[29]

DNS, Domain Naming System: DNS is the distributed naming service used on the Internet. The DNS can provide a machine's IP address, given domain names for the machine[29]

E

Endemic Incidents (see (D)DoS)

Egress Monitor: egress monitor is used for monitoring outgoing traffic in order to infer potential behavior from worms[95]

F

FTP, File Transfer Protocol: In the TCP/IP (or Internet) protocol suite, a file transfer protocol. FTP is an application layer protocol that uses the services of the TCP protocol at the transport layer to move the files[29]

Flood: In a network, the uncontrolled propagation of discovery or other packets[29]

Firewall: A firewall is a network component that provides a security barrier between networks or network segments. Firewalls are generally set up to protect a particular network or network component from attack, or unauthorized penetration, by outside invaders[29]

G

Gateway: In the context of local-area networks (LANs) and mainframe connections, a gateway is a hardware and/or software package that connects two different network environments[29]

H

Honeyfarm, Honeynet or Hybrid system: collection of honeypots monitored by a network telescope[69]

HTTP, Hypertext Transfer Protocol: HTTP is the primary protocol for requesting and providing documents on the Internet's World Wide Web (WWW)[29]

HMS, Host Motion Sensor: The IMS will be monitoring the range of addresses and the activity will be proxied to the HMS for in depth analysis of the incidents. Furthermore, in oder to avoid false positives and scaling issues combination with a pre-filtering installation could be examined[5]

I

IMS, Internet Monitor Sensor: The Internet Motion Sensor has the ability to detect from distributed networks of /24 nets around the global network[38]. Consists of 60 network telescopes from 18 organizations, enterprises and networks from academia in 3 continents world wide. Furthermore, observes approximately over 17 million addresses[52]

IP, Internet Protocol: is an address for a station or other device on the Internet. This type of address consists of 4 bytes, which are represented as decimal values separated by periods, as in 123.45.67.89. In order to ensure uniqueness, IP addresses are assigned in part by the Internet Assigned Numbers Authority (IANA)[29]

ICMP, Internet Control Message Protocol: In the TCP/IP protocol suite, a protocol used to handle errors at the network layer. ICMP is actually part of the IP, which is the network layer protocol in the TCP/IP suite[29]

IRC, Internet Relay Chat: A service that extends Talk capabilities to allow multi party conversations[29]

Ingress Monitor: The ingress monitors can be placed on routers locally or on passive topologies for logging traffic[95]

J

K

L

M

N

Network Telescope(Darknet/BlackHole/Internet Sink/Telescope): A Network Telescope considered to be a portion of routed IP addresses, with no or little legitimate traffic exists[64]

O

P

Pandemic (see also Worm)

Passive Telescope/Monitoring: Through passive monitoring, a telescope can observe the packets and keep logs and discard them, but without interacting with the attacker[34, 77]

Port Address or Name: A port address is a bus or memory address associated with a particular hardware port. There will generally be at least enough storage allocated at the port address to handle data being written or read at the port. A port name can be used instead of an address to refer to a port. The port name is presumably easier to remember than an address[29]

Payload: In ATM network terminology, the payload is the data portion of an ATM cell, or packet. This cell consists of a ?ve-octet header and a 48-octet payload. More generally, payload refers to the data portion of a packet (for example, of an IP packet, or datagram)[29]

(IP) Packet Header Fields: An IP packet consists of a header and data, known as a payload. The payload can be up to 64 kilobytes (KB) and must be at least 512 bytes[29]

Q

R

Router: A router gets a packet from a node or from another router and passes this packet on to a destination specified in an embedded (network layer) packet, which is known as an NPDU (network-layer protocol data unit)[29]

RPC, Remote Procedure Call: which makes it possible to call an application or function on any machine, just as if the resource were local or even part of the application[29]

S

SI, Epidemic Model: illustrates the growth of infectious pathogen propagating through similar random contacts between Susceptible and Infected hosts[65]

SYN, Synchronization: In bisynchronous, or bisync, communication, a special (SYN) character is used to establish synchronization for an entire data block. Both sender and receiver must be sychronized. The receiver must acknowledge the receipt of each block with alternating ACK characters[29]

SNMP, Simple Network Management Protocol: used to control network- management services and to transfer management-related data[29]

SMTP, Simple Mail Transfer Protocol: In the TCP/IP protocol suite, an application layer protocol that provides a simple electronic-mail service. SMTP uses the services of the TCP protocol at the transport layer to send and receive messages[29]

Susceptible: vulnerable host, but not yet exploited[65]

Stateful responder: will keep each connection's state if it is active[34]

Stateless responder: will design an application response based on previous packets. Furthermore, filtering might be needed for effective monitoring of the traffic[34]

T

Transit Network Telescope: monitors IP ranges, but from within the transit network and not from the edges of /x network. It can observe large ranges of addresses and can manage to monitor centrally and do not have synchronization and distribution problems[64]

TCP, Transmission Control Protocol: provides connection- and stream-oriented, transport-layer services. TCP uses the IP to deliver its packets[29]

TTL, Time To Live: Originally, this field indicated the number of seconds the packet was allowed to travel in a network before being destroyed[29]

U

UDP, User Datagram Protocol: provides connectionless transport-layer service. UDP also uses the IP to deliver its packet[29]

V

VM, Virtual Machine

W

Worm, Internet Worm: is a self-replicating/propagating program which exploits vulnerabilities without human intervention in order to infect hosts[56]

X

Y

Z

Bibliography

[1] Dave Aitel, *Nematodes beneficial worms*, September 2005.

[2] L. Andersson and L. Zhang, *Report from the iab workshop on unwanted traffic march 9-10, 2006 draft-iab-iwout-report-00.txt*, Tech. report, Network Working Group, Internet-Draft IETF, March 2006.

[3] Michael Bailey, Evan Cooke, Timothy Battles, and Danny McPherson, *Tracking global threats with the internet motion sensor*, NANOG 32,, Sept 7 2004.

[4] Michael Bailey, Evan Cooke, Farnam Jahanian, Jose Nazario, and David Watson, *The internet motion sensor: A distributed blackhole monitoring system*, Proc. of network and distributed system security symposium (ndss'05), Electrical Engineering and Computer Science Department University of Michigan and Arbor Networks, February 2005.

[5] Michael Bailey, Evan Cooke, Farnam Jahanian, Niels Provos, Karl Rosaen, and David Watson, *Data reduction for the scalable automated analysis of distributed darknet traffic*, USENIX Association Internet Measurement Conference, 2005.

[6] Michael Bailey, Evan Cooke, David Watson, Farnam Jahanian, and Jose Nazario, *The blaster worm: Then and now*, IEEE Security and Privacy Magazine, vol. Volume: 3, July-Aug 2005, pp. pages: 26–31.

[7] George Bakos, *Sqlsnake code analysis*, 2002.

[8] AT&T Labs Research Bellovin, *Icmp traceback messages*, Network Working Group, Internet Draft, March 2000.

[9] Steven M. Bellovin, *There be dragons*, in Proceedings of the Third Usenix UNIX Security Symposium, 1992, pp. pp. 1–16.

[10] Steven M Bellovin, *Packets found on an internet*, Computer Communications Review, vol. 23:3, July 1993, pp. pp. 26–31.

[11] Chen Bo, Bin Xing Fang, and Xiao Chun Yun, *Adaptive method for monitoring network and early detection of internet worms*, Lecture Notes in Computer Science, vol. Volume 3975/2006, ch. Surveillance and Emergency Response, pp. 178–189, Springer Berlin / Heidelberg, Wednesday, May 10, 2006.

[12] Y. Bouzida, F. Cuppens, and S. Gombault, *Detecting and reacting against distributed denial of service attacks*, Communications, 2006 IEEE International Conference on, vol. 5, June 2006, pp. 2394–2400.

[13] Claffy K C, *Internet traffic characterization*, Ph.D. thesis, UC San Diego, 1994.

[14] Martin Casado, Tal Garfinkel, Weidong Cu, Vern Paxson, and Stefan Savage, *Opportunistic measurement: Extracting insight from spurious traffic*, Tech. report, 4th Workshop on Hot Topics in Networks (HOTNETS-IV)., November 2005.

[15] Shigang Chen and S. Ranka, *Detecting internet worms at early stage*, Selected Areas in Communications, IEEE Journal on **23** (2005), no. 10, 2003–2012.

[16] Zesheng Chen, *Worm propagation models*, Mathematics Awareness Month (2006).

[17] Zesheng Chen, Lixin Gao, and Kevin Kwiat, *Modeling the spread of active worms*, in INFOCOM, Apr. 2003.

[18] Zesheng Chen and Chuanyi Ji, *Optimal worm-scanning method using vulnerable-host distributions*, nternational Journal of Security and Networks: Special Issue on Computer and Network Security **vol. 2, no. 1/2** (2007), Zesheng Chen and Chuanyi Ji International Journal of Security and Networks: Special Issue on Computer and Network Security, vol. 2, no. 1/2, 2007.

[19] Evan Cooke, Michael Bailey, Farnam Jahanian, and Richard Mortier, *Dark oracle: Perspective-aware unused and unreachable address discovery*, 3rd Symposium on Networked Systems Design and Implementation (NSDI '06) (San Jose, CA), May 8-10 2006.

[20] Evan Cooke, Michael Bailey, Z. Morley Mao, David Watson, Farnam Jahanian, and Danny McPherson, *Toward understanding distributed blackhole placement*, WORM '04: Proceedings of the 2004 ACM workshop on Rapid malcode (New York, NY, USA), ACM Press, 2004, pp. 54–64.

[21] Evan Cooke, Farnam Jahanian, and Danny McPherson, *The zombie roundup: Understanding, detecting, and disrupting botnets*, Proc. of Workshop on Steps to Reducing Unwanted Traffic on the Internet (SRUTI'05), July 2005.

[22] James Cowie, Andy T Ogielski, BJ Premorey, and Yougu Yuany, *Global routing instabilities triggered by code red ii and nimda worm attacks*, Tech. report, Renesys Corporation Hanover, NH 03750, December 2001.

[23] Weidong Cuiy, Vern Paxsonz, and Nicholas C. Weaverz, *Gq: Realizing a system to catch worms in a quarter million places*, Tech. report, INTERNATIONAL COMPUTER SCIENCE INSTITUTE, 1947 Center St. Suite 600 Berkeley, California 94704-1198, September 7 2006, University of California, Berkeley, CA International Computer Science Institute, Berkeley, CA.

[24] Tina Darmohray and Ross Oliver, *"hot spares" for dos attacks*, ;login:, Nov 2000.

[25] D Dean, M Franklin, and A Stubblefield, *An algebraic approach to ip traceback*, Network and Distributed System Security Symposium Conference Proceedings: 2001, 2001.

[26] T. Diibendorfer and B. Plattner, *Host behaviour based early detection of worm outbreaks in internet backbones*, Enabling Technologies: Infrastructure for Collaborative Enterprise, 2005. 14th IEEE International Workshops on (2005).

[27] Thomas Dubendorfe, *Flow-level traffic analysis of the blaster and sobig worm outbreaks in an internet backbone*, Switzerland ETH Zurich,, 2005, DIMVA 2005, Wien, Austria.

[28] Michalis Faloutsos, *Public real data repositories and measurement tools*, Tech. report, ACM Press ,ACM SIGCOMM Computer Communication Review,Volume 36 , Issue 2 (April 2006) ,Pages: 37 - 40, 2006.

[29] Werner Feibel, *Encyclopedia of networking*, 2nd ed., no. 0-7821-1829-1, Network Press, 1996.

[30] Virtual Center for Network and Security Data, *Predict workshop,newport beach, ca*, Sept. 27 2005.

[31] Jerome Francois, Radu State, and Olivier Festor, *Tracking global wide configuration errors*, Tech. report, Management of Dynamic Networks and Services Laboratoire Lorrain dInformatique et de ses Applications de Lorraine Nancy, France, 2006.

[32] Mark Fullmer and Steve Romig, *The osu flow-tools package and cisco netflow logs*, Proceedings of the 2000 USENIX LISA Conference. (New Orleans, LA), 2000, pp. 291–303.

[33] Carl G., Kesidis G., Brooks R.R., and Suresh Rai, *Denial-of-service attack-detection techniques*, Internet Computing, IEEE **10** (2006), no. 1, 82–89.

[34] Julia Grace and Claire OShea, *Network telescopes*.

[35] Guofei Gu, Monirul Sharif, Xinzhou Qin, David Dagon, Wenke Lee, and George Riley, *Worm detection, earlywarning and response based on local victim information*, ACSAC, Dec 2004.

[36] Burch H and Cheswick B, *Tracing anonymous packets to their approximate source*, In Proceedings of the 2000 USENIX LISA Conference. New Orleans (LA), 2000, pp. 319–327.

[37] Uli Harder, M.W. Johnson, J.T. Bradley, and W.J. Knottenbelt, *Observing internet worm and virus attacks with a small network telescope*, Tech. report, International Conference, PASM 2005, Proceedings of the 2nd Workshop on Practical Applications of Stochastic Modelling,, July 2005,pp.113126.

[38] Warren Harrop and Grenville Armitage, *Greynets: A definition and evaluation of sparsely populated darknets*, August 22-26 2005, Centre for Advanced Internet Architectures, Swinburne University of Technology Melbourne, Australia.

[39] Herbert W. Hethcote, *The mathematics of infectious diseases*, SIAM Review, vol. vol. 42, 2000,, pp. pp. 599653 ,http://www.math.uiowa.edu/ hethcote/PDFs/2000SiamRev.pdf.

[40] Alefiya Hussain, John Heidemann, and Christos Papadopoulos, *A framework for classifying denial of service attacks*, ACM Sigcomm (Karlsruhe, Germany), Aug 25-29 2003.

[41] P Ferguson Cisco Systems Inc and D Senie Amaranth Networks Inc, *Network ingress filtering: Defeating denial of service attacks which employ ip source address spoofing*, May 2000.

[42] Chuanyi Ji and Zesheng Chen, *Importance-scanning worm using vulnerable-host distribution*, Global Telecommunications Conference, 2005. GLOBECOM '05. IEEE **3**.

[43] Xuxian Jiang and Dongyan Xu, *Collapsar: A vm-based architecture fornetwork attack detention center*, Tech. report, August 9-13 2004.

[44] Jeffrey O Kephart and Steve R White, *Directed-graph epidemiological models of computer viruses,*, in IEEE Symposium on Security and Privacy, 1999, p. 343361.

[45] Peter Komisarczuk, Christian Seifert, Dean Pemberton, and Ian Welch, *Grid enabled internet instruments*, Tech. report, Victoria University of Wellington, School of Mathematics, Statistics and Computer Science, Wellington, New Zealand, March 2007.

[46] Ramana Rao Kompella, Sumeet Singh, and George Varghese, *On scalable attack detection in the network*, Tech. report, University of California, San Diego. Internet Measurement Conference 2004, 2004.

[47] E. Kranakis, D. Whyte, and P.C. van Oorschot, *Detecting intra-enterprise scanning worms based on address resolution*, Computer Security Applications Conference, 21st Annual (2005).

[48] Balachander Krishnamurthy, *Mohonk: Mobile honeypots to trace unwanted traffic early*, Tech. report, AT&T LabsResearch, 2004.

[49] Abhishek Kumar, Vern Paxson, and Nicholas Weave, *Exploiting underlying structure for detailed reconstruction of an internet-scale event*, ACM IMC, Oct 2005, New Orleans, LA.

[50] Abhishek Kumar, Vern Paxson, and Nicholas Weaver, *Exploiting underlying structure for detailed reconstruction of an internet-scale event*, Tech. report, Georgia Institute of Technology, ICSI, October 19, 2005.

[51] L. Li, I. Hamadeh, S. Jiwasurat, G. Kesidis, P. Liu, and C. Neuman, *Emulating sequential scanning worms on the deter testbed*, Tech. report, Pennsylvania State University, University Park.

[52] Bailey M., Cooke E., Jahanian F., Myrick A., and Sinha S., *Practical darknet measurement*, Information Sciences and Systems, 2006 40th Annual Conference on, no. 10.1109/CISS.2006.286376, March 2006, pp. 1496 – 1501.

[53] Danny McPherson and Barry Greene, *Isp security: Deploying and using sinkholes*, June 2003.

[54] D. Moore and C. Shannon, *The spread of the witty worm*, Security & Privacy Magazine, IEEE **Vol. 2** (2004), no. No 4.

[55] David Moore, *Network telescopes: Observing small or distant security events*, Cooperative Association for Internet Data Analysis - CAIDA San Diego Supercomputer Center, University of California, San Diego, August 8 2002.

[56] ______, *Network telescopes overview: What is a "network telescope"?*, 2003.

[57] David Moore, Vern Paxson, Stefan Savage, Colleen Shannon, Stuart Staniford, and Nicholas Weaver, *The spread of the sapphire/slammer worm*, Tech. report, CAIDA, 2003.

[58] ______, *Inside the slammer worm*, Tech. report, IEEE Security and Privacy, 1(4):33-39, July 2003.

[59] David Moore and Colleen Shannon, *Network telescopes: The flocon files*, 2004.

[60] ______, *The spread of the code-red worm (crv2)*, March 30 2006.

[61] David Moore, Colleen Shannon, Douglas J. Brown, Geoffrey M. Voelker, and Stefan Savage, *Inferring internet denial-of-service activity*, Tech. Report 2, New York, NY, USA, 2006.

[62] David Moore, Colleen Shannon, and Jeffery Brown, *Code-red: a case study on the spread and victims of an internet worm*, Tech. report, in ACM Internet Measurement Workshop 2002, Marseille, France, Nov 2002.

[63] David Moore, Colleen Shannon, Geoffrey M. Voelker, and Stefan Savage, *Internet quarantine: Requirements for containing self-propagating code*, Proceedings of the 2003 IEEE Infocom Conference (San Francisco, CA), April 2003.

[64] David Moore, Colleen Shannon, Geoffrey M. Voelkery, and Stefan Savagey, *Network telescopes: Technical report*, Tech. report, Cooperative Association for Internet Data Analysis (CAIDA), July 2004.

[65] David Moore, Geoffrey M. Voelker, and Stefan Savage, *Quantitative network security analysis*, Tech. Report Tel: (858) 534-5160 Fax: (858) 534-5117, CAIDA/SDSC and CSE Department University of California, San Diego, 9500 Gilman Drive, MS 0505 La Jolla, CA 92092-0505, Dec 4 2002, dmoore@caida.org.

[66] Jose Nazario, *The blaster worm: The view from 10,000 feet*, http://monkey.org/jose/presentations/blaster.d/, 2003.

[67] Spatscheck O and Peterson L, *Defending against denial of service attacks in scout*, In Proceedings of the 1999 USENIX/ACM Symposium on Operating System Design and Implementation, 1999, pp. 59–72.

[68] Ruoming Pang, Vinod Yegneswaran, Paul Barford, Vern Paxson, and Larry Peterson, *Characteristics of internet background radiation*, IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement (New York, NY, USA), ACM Press, 2004, pp. 27–40.

[69] Vern Paxson, *Addressing the threat of internet worms*, ICSI Center for Internet Research and Lawrence Berkeley National Laboratory, Feb 2005.

[70] Honeynet Project, *Know your enemy: Honeynets*, Nov 2002.

[71] Niels Provos, *A virtual honeypot framework*, In Proceedings of the 13th USENIX Security Symposium (SanDiego,CA, USA), August 2004, p. pages114.

[72] Chen P.T., Laih C.S., Pougetand F., and Dacier M., *Comparative survey of local honeypot sensors to assist network forensics*, Systematic Approaches to Digital Forensic Engineering, 2005. First International Workshop on, no. 0-7695-2478-8, IEEE, IEEE, 7-9 November 2005, pp. On page(s): 120– 132.

[73] Moheeb Abu Rajab, Fabian Monrose, and Andreas Terzis, *Worm evolution tracking via timing analysis*, Tech. report, Johns Hopkins University The 3rd Workshop on Rapid Malcode (WORM), 2005.

[74] ______, *Fast and evasive attacks: Highlighting the challenges ahead*, vol. Volume 4219/2006, Lecture Notes in Computer Science, no. 978-3-540-39723-6, malware collection and analysis Malware Collection and Analysis, pp. 206–225, Springer Berlin / Heidelberg, September 21 2006.

[75] Joel Sandin, *P2p systems for worm detection,dimacs large scale attacks workshop presentation*, DIMACS Large Scale Attacks Workshop presentation, Sept 2003,.

[76] sandvine, *Million dollar home page ddos*, Tech. report, sandvine, 2006-01-16.

[77] Christian Seifert and Ian Welch andPeter Komisarczuk, *Taxonomy of honeypots*, Tech. report, VICTORIA UNIVERSITY OF WELLINGTON TeWhareWanangaoteUpokooteIkaaMaui, June 2006, TechnicalReportCS-TR-06/12.

[78] Giuseppe Serazzi and Stefano Zanero, *Computer virus propagation models*, Tech. report, Dipartimento di Elettronica e Informazione, Politecnico di Milano, Via Ponzio 34/5, 20133 Milano, Italy,, 2001.

[79] S. Shakkottai and R. Srikant, *Peer to peer networks for defense against internet worms*, Tech. report, Workshop on Interdisciplinary Systems Approach in Performance Evaluation and Design of Computer & Communications Systems, Oct 2006, Pisa, Italy, Oct 2006.

[80] Sumeet Singh, Cristian Estan, George Varghese, and Stefan Savage, *Automated-worm fingerprinting*, Tech. report, Department of Computer Science and Engineering University of California, San Diego, 2006.

[81] ______, *The earlybird system for real-time detection of unknown worms*, Tech. report, University of California San Deigo, Department of Computer Science, Technical Report CS2003-0761, August 2003.

[82] Dug Song, Rob Malan, and Robert Stone, *A snapshot of global internet worm activity*, November 13 2001.

[83] Stuart Staniford, Vern Paxson, and Nicholas Weaver, *How to 0wn the internet in your spare time*, How to 0wn the Internet in Your Spare Time, Aug 2002, pp. http://www.icir.org/vern/papers/cdc–usenix–sec02.

[84] Robert Stone, *Centertrack: An ip overlay network for tracking dos floods*, In Proceedings of the 2000 USENIX Security Symposium, 2000istics and Prevalence, pp. 199–212.

[85] Greg Tomsho, Ed Tittel, and David Johnson, *Guide to networking essentials*, ed 3rd ed., Course Technology, no. ISBN: 0619130873, Thomson, 25 Thomson Place, Boston, Massachusetts, 02210, 2003.

[86] Jean-Pierre van Riel and Barry Irwin, *Inetvis, a visual tool for network telescope traffic analysis*, Tech. report, Department of Computer Science Rhodes University Grahamstown, South Africa, 6140, 2006/01/25.

[87] Nicolas Vanderavero, Xavier Brouckaert, Olivier Bonaventure, and Baudouin Le Charlier, *The honeytank : a scalable approach to collect malicious internet traffic*, Tech. report, Computing Science and Engineering Department, Universite catholique de Louvain, 2004.

[88] Yegneswaran Vinod, Barford Paul, and Ullrich Johannes, *Internet intrusions: Global characteristics and prevalence*, Tech. report, In Proceedings of ACM SIGMETRICS, June, 2003.

[89] Michael Vrable, Justin Ma, Jay Chen, David Moore, Erik Vandekieft, Alex C. Snoeren, Geoffrey M. Voelker, and Stefan Savage, *Scalability, fidelity, and containment in the potemkin virtual honeyfarm*, SOSP '05: Proceedings of the twentieth ACM symposium on Operating systems principles (New York, NY, USA), ACM Press, 2005, pp. 148–162.

[90] A.D. Wood and J.A. Stankovic, *Denial of service in sensor networks*, Computer **35** (2002), no. 10, 54–62.

[91] Jianhong Xia, Lixin Gao, and Teng Fei, *Flooding attacks by exploiting persistent forwarding loops*, Proceedings of the USENIX/ACM Internet Measurement Conference, October 2005.

[92] Jianhong Xia, Sarma Vangala, Jiang Wu, Lixin Gao, and Kevin Kwiat, *Effective worm detection for various scan techniques*, Tech. report, Journal of Computer Security, vol. 14, no. 4, pp. 359-387, 2006.

[93] Vinod Yegneswaran, Paul Barford, and Dave Plonka, *On the design and use of internet sinks for network abuse monitoring*, vol. Volume 3224/2004, Lecture Notes in Computer Science, no. issn: 978-3-540-23123-3, attack and alert analysis Recent Advances in Intrusion Detection, pp. 146–165, Springer Berlin / Heidelberg, October 01 2004.

[94] Cliff C. Zou, Don Towsley, and Weibo Gong, *On the performance of internet worm scanning strategies*, Tech. report, Department of Electrical & Computer Engineering -Department of Computer Science, Univ. Massachusetts, Amherst Technical Report: TR-03-CSE-07, 2003.

[95] Cliff Changchun Zou, Lixin Gao, Weibo Gong, and Don Towsley, *Monitoring and early warning*, (2003).

[96] Cliff Changchun Zou, Weibo Gong, and Don Towsley, *Code red worm propagation modeling and analysis*, in Proceedings of the 9th ACM conference on Computer and communications security, 2002, pp. 138147, ACM Press.