Network Telescopes: Technical Report

David Moore*,†, Colleen Shannon*, Geoffrey M. Voelker†, Stefan Savage†

Abstract

A network telescope is a portion of routed IP address space in which little or no legitimate traffic exists. Monitoring unexpected traffic arriving at a network telescope provides the opportunity to view remote network security events such as various forms of flooding denial-of-service attacks, infection of hosts by Internet worms, and network scanning. In this paper, we examine the effects of the scope and locality of network telescopes on accurate measurement of both pandemic incidents (the spread of an Internet worm) and endemic incidents (denial-of-service attacks) on the Internet. In particular, we study the relationship between the size of the network telescope and its ability to detect network events, characterize its precision in determining event duration and rate, and discuss practical considerations in the deployment of network telescopes.

I. Introduction

Data collection on the Internet today is a formidable task. It is either impractical or impossible to collect data in enough locations to construct a global view of this dynamic, amorphous system. Over the last three years, a new measurement approach – network telescopes [1] – has emerged as the predominant mechanism for quantifying Internet-wide security phenomena such as denial-of-service attacks and network worms.

Traditional network telescopes infer remote network behavior and events in an entirely passive manner by examining spurious traffic arriving for non-existent hosts at a third-party network. While network telescopes have been used in observing anomalous behavior and probing [2], [3], [4], to infer distant denial-of-service attacks [5], [6], and to track Internet worms [7], [8], [9], [10], [11], to date there has been no systematic analysis of the extent to which information collected locally accurately portrays global events. In this paper, we examine the effects of the scope and locality of network telescopes on accurate measurement of both pandemic incidents (the spread of an Internet worm) and endemic incidents (denial-of-service attacks) on the Internet.

Since network telescopes were named as an analogy to astronomical telescopes, we begin by examining the ways in which this analogy is useful for understanding the function of network telescopes. The initial naming of network telescopes was driven by the comparison of packets arriving in a portion of address space to photons arriving in the aperture of a light telescope. In both of these cases, having a larger "size" (fraction of address space or telescope aperture) increases the quantity of basic data available for processing. For example with a light telescope, the further processing would include focusing and magnification, while with a network telescope, the processing includes packet examination and data aggregation. With a traditional light telescope, one observes objects, such as stars or planets, positioned in the sky and associates with the objects or positions additional features such as brightness or color. With a network telescope, one observes host behavior and associates with the behavior additional features such as start or end times, intensity, or categorization (such as a denial-of-service attack or worm infection). With astronomical telescopes, having a larger aperture increases the resolution of objects by providing more positional detail; with network telescopes, having a larger address space increases the resolution of events by providing more time detail.

II. Terminology

This section introduces the terminology we will use in our discussion of the monitoring capabilities of network telescopes for Internet research. We describe both terms relating to single nodes on the network and terms for describing networks in general.

E-mail: {dmoore,cshannon}@caida.org, {voelker,savage}@cs.ucsd.edu.
∗ - CAIDA, San Diego Supercomputer Center, University of California, San Diego.
† - Computer Science and Engineering Department, University of California, San Diego.
Support for this work is provided by NSF Trusted Computing Grant CCR-0311690, Cisco Systems University Reseach Program, DARPA FTN Contract N66001-01-1-8933, NSF Grant ANI-0221172, National Institute of Standards Grant 60NANB1D0118, and a generous gift from AT&T.

A. *Single Host Terminology*

Fundamental to the concept of a network telescope is the notion of a networked computing device, or *host*, that sends traffic collected by the telescope. While a given host may or may not be sending traffic in a uniformly random manner throughout all of a portion of the Internet address space, it does send traffic to at least one destination. Each destination, or *target* represents a choice at the host to send unsolicited traffic to that particular network location. Unless otherwise noted, a *target* should be assumed to be the result of a random, unbiased selection from the IPv4 address space. Each host chooses targets at a given *targeting rate*. While in reality, a host's *targeting rate* may vary over time, we consider only fixed *targeting rates* except as noted in Section IV-B.

Each host selecting targets for network traffic at a given targeting rate constitutes an *event*. While in many cases the actions of the host may be strongly influenced or completely controlled by a specific external influence, we reserve the term *event* to describe the actions of a single host. As with our definition of *target*, an *event* can be assumed to describe a host uniformly randomly generating IP addresses unless otherwise noted.

B. *Network Terminology*

IPv4 (Internet Protocol version 4) is currently the most ubiquitously deployed computer network communications protocol. IPv4 allows 32 bits for uniquely specifying the identity of a machine on the network, resulting in 2^{32} possible IP addresses. Portions of this IPv4 space are commonly described based on the number of bits (out of 32) set to uniquely identify a given address block. So a $/8$ describes the range of 2^{24} addresses that share the first 8 bits of their address in common. A $/16$ describes the range of 2^{16} addresses that share the first 16 bits in common, a $/24$ describes the set of 2^8 address with the first 24 bits in common, and a $/32$ uniquely describes a single address. In addition to its common usage to describe network block (and therefore, telescope) sizes, / notation is useful for quickly calculating the probability of the telescope monitoring a target chosen by a host. In general, the probability p is given by the ratio of address space monitored by the network telescope to the total available address space. For an IPv4 network of size /x the probability of monitoring a chosen target is given by: $p_x = \frac{1}{2^x}$. So for a network telescope monitoring a $/8$ network, $p_8 = \frac{1}{2^8} = \frac{1}{256}$

While the use of / notation in IPv4 networks is convenient for explanation of the properties of network telescopes, our results do not apply solely to networks with 2^{32} possible addresses. As long as the monitored fraction of total address space is known, expanding the range of address space (for example, an IPv6 network that uses 128 bit addresses), or limiting the range of address space (excluding multicast addresses from uniformly random target selection) does not matter.

III. Measuring Single Host Events

In this section, we will examine the effect of a telescope's size on it's ability to accurately measure the activity of a remotely monitored host.

A. *Detection time*

Because network telescopes are useful for their ability to observe remote events characterized by random, spontaneous connections between machines, it is useful to discern the minimum duration of an observable event for a given size telescope. Many phenomena observable via network telescopes, have a targeting rate that is either fixed or relatively constrained. For example, during a denial-of-service attack, the rate of response packets from the victim is limited by the capabilities of the victim and the available network bandwidth. Also, recent Internet worms have fallen into two categories: those which spread at a roughly constant rate on all hosts due to the design of the scanner and those which spread at different rates on hosts as limited by local CPU and network constraints. In these particular instances, and with any phenomenon observed in general, we would like to be able to identify the duration of the event. So while the raw data collected by a telescope consists of recorded targets received by the host, we treat this measurable quantity as a product of the targeting rate and the time duration. By splitting total targets into the product of rate and time, we can better understand how to adjust the interpretation of telescope data as a function of parameters changing in the real world.

A.1 Single Packet Detection

When a host chooses IP addresses uniformly randomly, the probability of detection is based on a geometric distribution. A network telescope observes some fraction of the entire IP address space. We call this fraction p, for the probability of a single packet sent by the host reaching the network telescope. If the host sends multiple packets, the number of packets seen at the telescope is described by a binomial distribution, with parameter p. However, the when a host chooses IP addresses uniformly randomly, the probability of the telescope monitoring a specific target is given by a geometric distribution.

We assume that all of the target IP address choices made by the host are independent, so when the host generates multiple packets, each packet has p odds of being sent to the telescope. Thus each target chosen by the host is by definition a Bernoulli trial. As mentioned above, we consider the number of packets that have been generated by the host as the product of the rate at which packets are sent, r, times the elapsed time, T.

The probability of seeing at least one packet in T seconds is given by $P(t \leq T) = 1-(1-p)^{rT}$. Figure 1 shows this cumulative probability of observing at least one packet from a host with a targeting rate of 10 probes per second with different network telescope sizes. Clearly telescope size has a significant impact on likelihood of observing a specific event.

The elapsed time before at least one packet of an event is observed with Z probability is given by the equation:

$$T = \frac{-1}{r \log_{\frac{1}{Z}}(1-p)} \tag{1}$$

So the expected number of packets until a first packet is seen is $\mu_N = \frac{1}{p}$ with variance $\sigma_N^2 = \frac{1-p}{p^2}$. Because we are interested in the packet rate and the time interval, we replace the absolute number of packets sent with rT and solve for elapsed time: $\mu_T = \frac{1}{rp}$.

However the probability distributions for a given targeting rate telescope size can be very wide, and the utility of a network telescope depends on knowing the specific likelihood of observing an event.

For example, rather than knowing that the expected time to observe one or more packets from a Code-Red-like host on a $/8$ is 25.6 seconds, it is more useful to know that the telescope would have a 63.2% likelihood of detecting that event in the same 25.6 second time period. In the other direction, we can compute the time required to observe this kind of event at a given likelihood. For example, to observe one or more packets from a Code-Red-like host on a $/8$ with 99.999% probability requires 4.9 minutes.

Table I summarizes the average, standard deviation and median time to see the first packet from a host choosing random IP addresses at 10 per second for several feasible network telescope sizes. Note that with a telescope monitoring a single IP address (a $/32$) the average time to observe a host at 10 addresses per second is over 13 years and the time to observe with 95% likelihood is over 40 years.

For a network telescope to have any research or operational significance, it must have good odds of actually seeing events. It must be able to make accurate quantitative analysis of events: count them, classify them, etc.

In practice events monitored by a network telescope do not last forever; most end after a period of time. Because the odds of a telescope observing an event is relative by that event's entire duration, there is a tradeoff between the likelihood of detection and practical limits imposed by real world events. While the desirable confidence in observing a given event varies depending on experimental design, clearly a telescope providing a 95% chance of observing your target event is a more useful tool than one offering a 5% chance. Figure 2 shows the time required to observe with 95% likelihood the first packet from a host choosing random IP addresses at various fixed rates.

Looking at two different IP address fractions p_1 and p_2, we can compare the times T_1 and T_2 needed for a given desired probability of detecting at least one packet P at rate r:

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

$$T_1 = T_2 \frac{\ln(1-p_2)}{\ln(1-p_1)} \tag{2}$$

Note that while one might think the detection time would scale linearly and directly with the amount of address space, it doesn't due to the increasing mass in the head of the distribution. While in practice, the difference from linear scaling

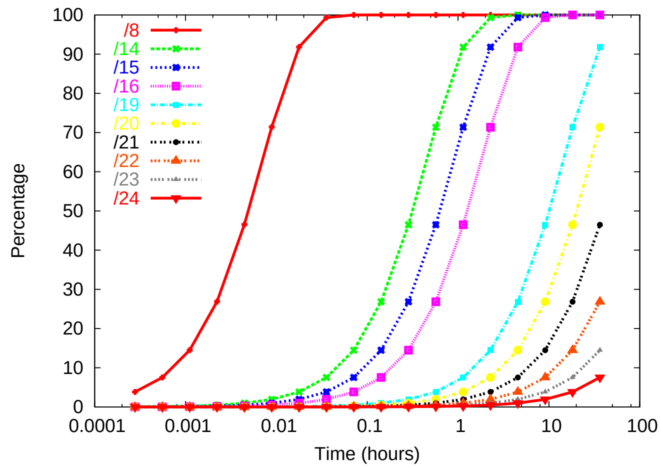


Fig. 1. Probability of observing at least one packet from a host choosing random IP addresses at 10 per second.

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 I

Statistics on the times when different network sizes would see their first packet from a sender choosing random IP addresses at 10 per second. The 95h percentile column shows the event duration for which the network telescope would monitor 95% of events, while the 5th percentile column shows the event duration for which the network telescope would *miss* 95% of events.

is slight, it's a somewhat non-intuitive consequence worth noting. In particular, the difference is more prominent in comparing larger telescopes: a /1 is more than twice as good as a /2; a /2 takes 2.41 times as long to detect one or more packets at the same confidence level as a /1.

For a more realistic example, a $/24$ would require 65664 times as long as a $/8$ to detect at least one packet with the same likelihood, slightly worse than the multiplicand derived from strict scaling by the relative fraction of their address spaces (65536).

A.2 Multiple Packet Detection

Because a network telescope can receive traffic due to a wide variety of causes, including packet corruption or misconfiguration, the ultimate utility of a telescope is predicated upon the ability to distill significant events (a denial-of-service attack or an Internet worm, for example) from background traffic. Observing multiple packets from a single source increases our confidence that the telescope has monitored a significant event. Thus in practice a telescope often needs to receive k or more packets from a given event in order to rule out a variety of irrelevant causes and develop an accurate classification. So depending on the characteristics of the event to be monitored and the experimental design, a

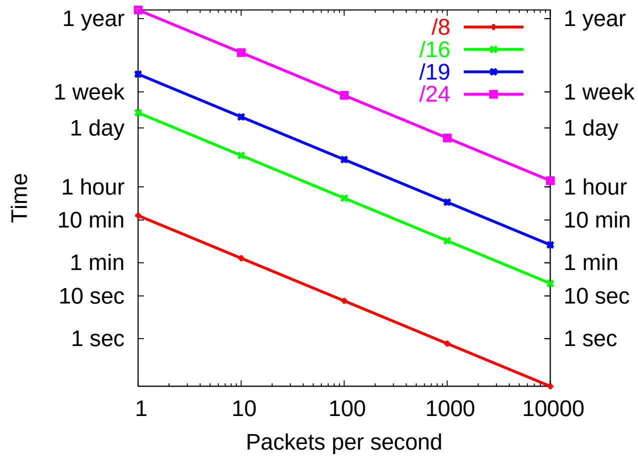


Fig. 2. Time required to observe with 95% likelihood the first packet from a host choosing random IP addresses at various fixed rates.

threshold k packets can be selected such that the probability of seeing k or more packets out of N transmitted is:

$$P(\text{saw} \geq k) = 1 - \sum_{y=0}^{k-1} \binom{N}{y} p^y (1-p)^{N-y} \tag{3}$$

So, for example, the probability of seeing at least 100 packets on a /8 telescope from a DoS attack at 500 pps lasting 1 minute would be:

$$N = 1000\text{pps} \cdot 60\text{sec} = 30000\text{packets}$$
$$k = 100\text{packets}$$
$$p = 2^{-8}$$

$$P = 1 - \sum_{y=0}^{99} \binom{30000}{y} (2^{-8})^y (1 - 2^{-8})^{30000-y}$$
$$P = 95.2\%$$

Whereas on a /16, the probability of seeing at least 1 packet is 36.7% – so your odds of observing the event at all are low. Even if the attack lasted 5 minutes, there is only a 89.9% chance that a /16 telescope would see at least 1 packet. When such a small number of packets are observed, it is impossible to extract the magnitude or volume of the original event with any confi dence.

B. Start and End Time Precision

To correlate remote security events, it is important to know the start and end time of the event, as well as the event's duration. When a network telescope observes some packets from an event, the time of the fi rst packet monitored by the telescope is an upper-limit on the start time of the event. Likewise, the time of the last packet observed provides a lower-limit for the end time. This is because the host may take some time after the start of the event to randomly select an address visible to the telescope. Similarly, after we've observed the last packet, the host may continue to choose targets that fall outside of the telescope's address range for some time. While it is impossible to know the exact real start and end times of an event, we can compute the likelihood of those times being in particular ranges around the observed times. So while we may not know the exact time that an event began, but we can know that with 99% probability it was not earlier than some reasonable period of time before our fi rst observed packet. For example, with a $/8$ network telescope, there is a 99% confi dence that the event began no earlier than two minutes before our fi rst observed packet for an event choosing IP addresses at a targeting rate of 10 per second.

If IP addresses are chosen independently and fall into the range monitored by the telescope with probability p, then the probability that exactly N targets were chosen outside of the telescope's range before the first observed target is given by $(1-p)^N$.

As seen in Section III-A.1, the number of IP addresses chosen (or amount of time, if the rate is known) between the start of an event and the time at which a particular telescope observes one packet is described by a geometric distribution. However, since the address choices are independent Bernoulli trials, if a telescope has observed a packet at a given time, the possible event start times (less than or equal to the packet observation time) is also described by a geometric distribution.

If the targeting rate is known and fixed, conversion of target distribution to a time probability distribution is straightforward - simply divide by the rate. The time distributions allow assignment of confidence intervals to the duration of the event before the first observation. Using this same approach, confidence intervals can be assigned for the total duration of the event. Consequently, for events with a constant targeting rate, the distribution of the number of packets sent after the last observed packet is the same as the distribution for sent packets before the first observed.

For example, using a fixed rate of the host choosing 10 targets per second, Table I shows that on average, the onset of an event occurred within 1.3 minutes of a /8 network telescope receiving its first packet.

C. Estimating Event Targeting Rate

In addition to obtaining bounds on the possible start and end times of an observed event, one would also like to estimate a host's actual targeting rate from targets observed by the telescope. Both the response rate of a denial-of-service attack victim and the scan rate of a host spreading an Internet worm are significant real-world targe If the duration of an event and the total number of targets are known, then the targeting rate can be determined exactly. With an estimate in lieu of either or both known values - the duration and total number of targets - an estimate of the targeting rate can be computed.

If during an event a host chooses N total targets, the number of targets observed by a network telescope is given by a binomial distribution with parameter p. The expected number of observed targets is given by $\mu = Np$ with variance $\sigma^2 = Np(1-p)$. When N is large, the binomial distribution can also be approximated by a normal distribution. When $p << 1$, and $\sigma^2 \approx Np$ the distribution is also approximated by a Poisson distribution.

To find a distribution for the total number of targets, N, given the observation of n targets, an inverse problem needs to be solved. While $P[n|N]$ is given by the binomial distribution, determining $P[N|n]$ requires knowing $P[N]$ and $P[n]$, which are not known *a priori*. However, since when N is large, the distribution is normal and for large telescopes, the observed n will also be large, allowing the estimation $\hat{N} = \frac{n}{p}$.

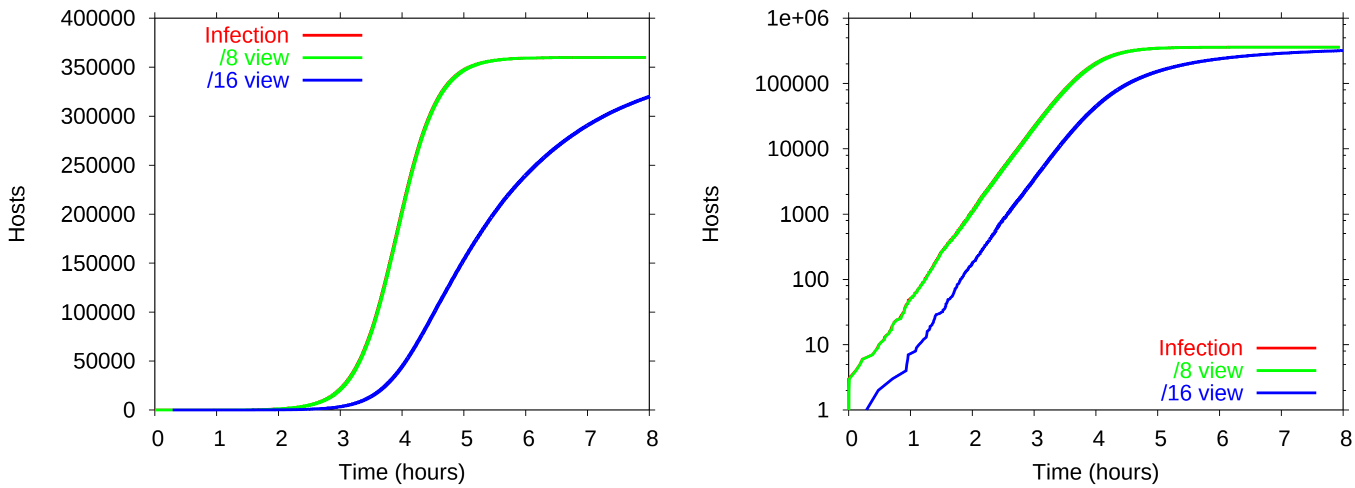
D. Obfuscation of low rate or short duration events

As detailed in Section III-A, the size of the address range monitored by a network telescope influences the scope and duration of events it can accurately monitor. With any practically sized network telescope, there are events which generate an insufficient number of targets to be detected with a reasonable likelihood on that telescope. For example, an event of only 10 targets has less than a 4% chance of having one or more targets observed on a large $/8$ telescope. Therefore, for any given telescope, there will be events which choose IP addresses at a sufficiently low rate or whose duration is sufficiently short, that the event is unlikely to be detected.

E. Flow Timeout Problem

At a network telescope, a remote denial-of-service attack appears as a stream of packets with fields implying that a particular destination is under assault. However, there is a conceptual disconnect between the high-level notion of an "attack" and the low-level observed "stream of packets". When grouping packets into an attack, the use of packet header fields (such as the victim IP address or the victim port number) are straightforward, however the time that the packets are observed is also important. While two packets arriving one second apart from the same victim are likely from the same attack. However, two packets, with none in between, arriving a week apart are probably not the same attack. This problem of deciding whether to group or split a sequence of packets is generally known as the flow timeout problem.

In the general case of tracking events with network telescopes, there are three main constraints influencing the decision of when to split a given stream of packets: do not split an event at a reasonable packet rate because of probability of observation at the telescope; allow sufficient time to prevent splitting because of event features (short duration pulsing



(a) Linear infection scale.

(b) Logarithmic infection scale.

Fig. 3. Comparison of actual host infections to observed host infections with $/8$ and $/16$ network telescopes, using a simulation of 360,000 vulnerable hosts with CodeRed-like worm at 10 scans per second. Note that the curve observed by the $/8$ lies on top of the actual infection curve and that the $/16$ curve is distorted from the expected logistic shape of a random constant spread worm.

of the DoS attack tools or Internet worm behavior; reboots of the DoS or Internet worm victim); and keep the time as short as possible to prevent grouping of unrelated events. Of these three concerns, the latter two can be subjective and are domain-specific variables for experiment design independent of telescope size. Thus we will examine only the need to not split a single attack into two attacks as a measurement artifact caused by an inopportune combination of random addresses.

At first this problem seems as though it might lend itself to the geometric distribution solution described in Section III-A.1 for event detection and Section III-B for event start and end time bounds, in that those approaches compute the odds of seeing a "gap" of k target choices missing the telescope address space at a specific location in the stream of target choices, namely at the start or end. However, when N is much larger than k there are multiple possibilities for a gap of size k to arise at *any* location in the sequence. So for long-lived stream of target choices, we wish to compute the odds of seeing a run of target failures of length $\geq k$ out of N. The longest expected gap run is $\log_{\frac{1}{1-p}} Np$.

A previous denial-of-service study [5] used a flow timeout of five minutes on a $/8$ telescope to look at week-long datasets. If there were week-long events in those datasets with targeting rates $\sim 0.0008 < r < \sim 8.4$ (targets per second), it is expected that there would be a gap in the observation of more than five minutes, causing the single event to be reported as two. Flooding denial-of-service attacks, are unlikely to have targeting rates that low. However, on a $/16$ with the same five minute timeout for a week-long dataset, events with targeting rates $\sim 0.1 < r < \sim 2150$ (targets per second) are expected to be split. Since denial-of-service attacks often occur in this range, either a longer timeout or larger telescope would be needed to prevent artificial splitting. Using a one hour timeout on a /16 telescope allows significantly more viable targeting rates of $\sim 0.1 < r < \sim 127$ (targets per second), although the threshold at which artificial flow timeouts become a problem is still an order of magnitude larger than a /8 telescope with a five minute timeout.

Note that the targeting rate of the Code-Red worm infected hosts were in the range of 5 to 10 targets per second, so even with a $/8$ monitor using a five minute flow timeout could lead to incorrectly splitting week-long infection events.

IV. Measuring Population Parameters

While most significant measurement results from network telescope data depend on accurate analysis of single host events, many incidents affect a large number of hosts in a related, aggregated manner. In this section we explore the effects of network telescope features on incidents that affect an entire population of hosts contemporaneously.

Qualitatively, the effects of network telescope size on the aggregate population measurements of an incident can be seen in Figure 3 which shows the effect of network telescopes on the observed shape of the host infection curve for a

N	size of the total vulnerable population
$S(t)$	susceptibles at time t
$I(t)$	infectives at time t
β	contact rate
T	time when 50% infective, $I(T) = N/2$

TABLE II
COMPONENTS OF THE SI MODEL.

random constant spread worm.

A. Start and End Time Precision

As with single host events, it is not possible to determine the exact start or end time for a widespread incident. However, it is possible to extrapolate from the time probability distributions and confidences described in Section III-B to determine the a ranges of time in which the incident has a high probability of beginning and ending. For incidents composed of events that are uniform with respect to targeting rate, calculation of the start and end times of the incident are straightforward. The onset probability distribution of the event with the packet received earliest at the telescope becomes the onset probability distribution for the entire event. Similarly, the end of the incident is described by the probability distribution of the event with the packet received by the telescope at the latest time.

B. Non-Uniformity of Scanning Rate

As mentioned in the previous section, many incidents are composed of events with varying targeting rates. Diversity of targeting rates across events can be caused by incident characteristics (a denial-of-service attack tool set to run at different packet rates for different victims, or an Internet worm that attempts to modulate its speed based on some volume of previously infected hosts) or by host specific causes (high CPU load on the machine due to user activity, low bandwidth upstream connection, congestion on local links, competition with network traffic from other applications, variations in operating system or application specific software). While for the Code-Red Internet worm, a relatively uniform targeting rate was observed [7], for the SQL Slammer worm, targeting rates as high as 26,000 probes per second were observed, far in excess of the average of approximately 4,000 probes per second [10].

C. Using Packet Rates/Count Instead of Host Counts for Model Fitting

Random constant spread worms, such as CodeRed, are well described using the classic SI epidemic model that describes the growth of an infectious pathogen spread through homogeneous random contacts between *Susceptible* and *Infected* individuals. This model, described in considerably more detail in [12], has been used to determine the characteristics of real-world worm spread [8], [7], [10]. Using the terms defined in Table II, the number of infected individuals at time t is:

$$I(t) = N\frac{e^{\beta(t-T)}}{1+e^{\beta(t-T)}} \tag{4}$$

where T is a constant of integration fixing the time where half of the population is infected.

This result is well known in the public health community and has been thoroughly applied to digital pathogens as far back as 1991 [13]. To apply this result to Internet worms, the variables simply take on specific meanings. The population, N, describes the pool of Internet hosts vulnerable to the exploit used by the worm. The susceptibles, $S(t)$, are hosts that are vulnerable but not yet exploited, and the infectives, $I(t)$, are computers actively spreading the worm. Finally, the contact rate, β, can be expressed as a function of worm's scan rate r and the targeting algorithm used to select new host addresses for infection. In this paper, we assume that an infected host chooses targets randomly, like Code-Red v2, from the 32-bit IPv4 address space. Consequently, $\beta = r\frac{N}{2^{32}}$, since a given scan will reach a vulnerable host with probability $N/2^{32}$.

Two primary model-fitting approaches have been taken to tracking and understanding the spread of Internet worms. Namely: counting the number of packets arriving at a telescope in specific time buckets [8], [10], and counting the number of observed infected IP addresses [7], [14], [15], [8]. To understand the advantages and disadvantages of these approaches, we examine how accurately each can measure certain worm properties with varying sizes of network telescopes.

When using small telescopes or measuring worms with highly non-uniform scan rates, using cumulative counts of unique IP addresses seen over the lifetime of the worm, can improve the estimate of infected hosts and allow better fitting to the model. For example, hosts infected with Slammer on 56Kbps dial-up modems would each scan at rate of 17 per second. Using a $/16$ telescope, only 61% of these modem hosts would be seen by the telescope in an hour window, assuming the hosts are transmitting continuously over the entire hour. Code-Red scanned at a rate below 10 per second. At those rates, a $/16$ telescope would see less than 43% of the hosts in an hour window; a $/14$ would see 89% in the same hour, while a $/19$ would only see 81% of the hosts in an entire day. However, using cumulative counts may lead to hiding of the effects of computers being repaired or removed from the network, and may also cause over-estimation due to the DHCP effect [7].

In the SI model where once a host is infected it continues to be infected for all time, the total number of unique infective hosts in a time window ($t_a \leq t \leq t_b$) is simply the number infected at the end time, $I(t_b)$. However, the total number of scans in the time window is given by the sum of the scans sent by each individual infective at each timestep during the window:

$$\begin{aligned} \text{scans}(t_a, t_b) &= \int_{t_a}^{t_b} rI(t)\,dt \\ &= \frac{rN}{\beta} \ln\left(\frac{1+e^{\beta(t_b-T)}}{1+e^{\beta(t_a-T)}}\right) \end{aligned} \tag{5}$$

When using a network telescope, equation (5) must be modified to scale by the expected number of scans to arrive on our part of the network:

$$\begin{aligned} \text{telescopescans}(t_a, t_b, p) &= p \cdot \text{scans}(t_a, t_b) \\ &= \frac{prN}{\beta} \ln\left(\frac{1+e^{\beta(t_b-T)}}{1+e^{\beta(t_a-T)}}\right) \\ &= 2^{32} p \ln\left(\frac{1+e^{\beta(t_b-T)}}{1+e^{\beta(t_a-T)}}\right) \end{aligned} \tag{6}$$

Therefore, when fitting network telescope data of host counts, either bucketed or cumulative, we use $I(t)$ from equation (4) and when fitting bucketed scan counts, we use $\text{scans}(t_{i-1}, t_i)$ from equation (6). Recalling that $\beta = \frac{rN}{2^{32}}$, we note that $\text{scans}(t_{i-1}, t_i)$ always occur r and N together as a multiplicative pair. This implies that scan count data alone provides no possible information about either r or N, only about rN.

In Figures 4 and 5, we examine the model parameters selected by least-squares fit to a simulated CodeRed-like worm as would be observed by different sized telescopes. While least-squares fitting against the number of hosts observed matches well out to a $/17$ telescope, fitting against scans matches well out to a $/24$. However, this is with clean simulated data. Initial results of fitting to host data for measured, real-world worm events suggests much worse fits.

D. *Correcting for Distortions in Infected Host Counts*

While model fitting to the number of scans, as discussed above, works well across a wide range of telescope sizes, that approach is unable by itself to provide information about the number of hosts infected. Using the time at which a telescope first observes a scan from an infected host as an estimate of the infection time of that host, a continuous time series of the cumulative number of observed infected hosts can be determined. Similarly, by using time buckets and counting the number of unique hosts observed as scanning in each bucket, the number of observed infected hosts at different times can be determined when the bucket size is appropriately chosen. However, unlike the observed scan data over time which is accurate across many telescope sizes due to the mass-action effect of all of the infected hosts, the observed infected host counts over time can be greatly distorted by a network telescope. Figure 3 shows the distortion effect of network telescopes on the observed shape of the host infection curve for a random constant spread worm.

The continuous SI model tells how many actual infected hosts there are at any time, $I(t)$. The expected observed count of hosts with a telescope, $O(t)$, can be computed from $O'(t)$:

$$O'(t) = \int_{-\infty}^{t} I'(t-x)\textit{Prob}[\text{delay} = x]\, dx \tag{7}$$

$$\vdots$$

$$O'(t) = \int_{-\infty}^{t} I'(t-x) pre^{-prx}\, dx \tag{8}$$

$$O(t) = \int_{-\infty}^{t} O'(y)\, dy \tag{9}$$

$$O(t) = pr \int_{-\infty}^{t} \int_{-\infty}^{y} I'(y-x) e^{-prx}\, dx\, dy \tag{10}$$

Also, because geometric distribution is memoryless, we can compute $I(t)$ from $O(t)$ similarly:

$$I'(t) = \int_{t}^{\infty} O'(x)\textit{Prob}[\text{delay} = x - t]\, dx \tag{11}$$

$$I(t) = \int_{-\infty}^{t} I'(y)\, dy \tag{12}$$

$$I(t) = \int_{-\infty}^{t} \int_{y}^{\infty} O'(x)\textit{Prob}[\text{delay} = x - y]\, dx\, dy \tag{13}$$

In [16], Zou et al. present a technique to reconstruct I_t (the discrete time version of $I(t)$) from measured values of O_t (although they refer to O_t as C_t). They provide an estimation $\hat{I}t$ as:

$$\hat{I}_t = \frac{C_{t+1} - (1-p)^{\eta} C_t}{1 - (1-p)^{\eta}} \tag{14}$$

where η is the number of scans by individual host in the discrete time step, $\eta = r \cdot (\text{bucket size})$. Note that C_t is the total seen from the beginning of time not just those seen in the last bucket, i.e., it is cumulative.

Note this approach is very nice in that it allows you to estimate I_t only one timestep behind what you have observed with C_t, allowing online and incremental processing. However the approach depends on the hosts remaining infected indefinitely; the model does not allow for repair or deactivation of infected machines. Also, this model requires that the scan rate, r, remain constant over time. Real events clearly do exceed these constraints. However, the advantages appear to outweigh the disadvantages.

V. Practical Considerations

In this section we first discuss practical issues in the use of a network telescope to detect and measure events on the Internet, and then discuss various approaches for addressing with these issues.

A. Practical concerns

One of the most important assumptions in the use of a network telescope is that IP addresses are being chosen randomly with no bias towards or away from the telescope's address space. In practice, IP addresses are often not chosen uniformly from the entire IPv4 address space for a variety of reasons:

1. narrowing of address space (cutting out certain regions), but otherwise selecting uniformly;
2. biasing (or weighting) some regions more heavily than others (and also biases which are source dependent - i.e., worm trying nearby addresses);
3. bugs or biases in underlying PRNG (may also be source dependent if bad PRNG is seeded in a local manner).

While in some situations, these three cases are indistinguishable, we treat them separately. We consider the first two causes as deliberate design choices and the third as a bug. In general, it is difficult to understand or correct for any of these biases without access to the code. For example, scanning at a given rate on a narrower address space is indistinguishable from scanning at a proportionally larger rate on the entire address space.

With the assumptions of uniformity (no bias) and independence in the random selection algorithm for IP addresses, the probability p of seeing a packet at a network telescope is the ratio of the size of the telescope to the size of the entire selectable address space. When the selection algorithm is not uniform, the probability p may be determined from the operation of the algorithm itself. Any particular non-uniform selection algorithm will cause measurements with a telescope to appear to have been taken with a telescope of different size.

Another assumption in the use of a network telescope is that the observed targeting rate accurately represents the true targeting rate of hosts. In practice, a network telescope can underestimate the targeting rate for a number of reasons:

1. An aggressive, wide-spread event may generate enough traffic to overload the network used by the telescope, resulting in traffic dropped within the network before it reaches the telescope.
2. Recording events in detail for future analysis requires processing and storage of event traffic. Data collection and analysis hardware may be limited in the ability to process or store event traffic during peak load, or have the capacity to store all event traffic during long events. Such under-provisioning limits accurate characterization of events during peak load, or produces gaps in analysis over time.
3. Internet routing instability may prevent traffic from particular hosts from reaching the address space observed by the telescope.
4. A network telescope that passively monitors event traffic can characterize the network dynamics of an event, but without actively responding to event traffic it is limited in its ability to classify the event in detail.
5. Traffic observed by a network telescope is a combination of multiple events from numerous causes, such as multiple simultaneous denial-of-service attacks, active worms, and direct port scans of the address space observed by the telescope. Accurately differentiating all of these simultaneous events is a challenging problem. If unable to differentiate, a telescope will improperly characterize the events.

B. Potential solutions

One solution to a number of the above problems is the use of a *distributed network telescope*. The discussion so far has assumed a telescope monitoring a contiguous range of address space. A distributed network telescope combines smaller telescopes observing different regions of the network address space into a single, larger network telescope. A distributed telescope can take the form of a small number of relatively large contiguous address regions such as the telescope used by Yegneswaran, et al. [17], a heterogeneous volunteer distributed telescope [4], or a very large number of individual addresses such as hosts in a large-scale, wide-area peer-to-peer network [18].

A distributed telescope has a number of benefits. It increases the observed address space, thereby improving detection time, duration precision, and targeting rate estimation over any of the individual telescopes. Observing multiple, disparate regions of the address space can help reveal and overcome targeting bias in the event. And multiple network telescopes can aggregate their resources to effectively combine their network capacity and monitor resources to better handle peak loads.

However, using a distributed network telescope comes at a cost. To correlate events across distributed telescopes over time, the telescope clocks need to be synchronized. The accuracy of analyses involving time therefore depends upon the accuracy of the time synchronization. Real-time analysis using the distributed telescope requires the collection of distributed data sets for analysis. And analyses of distributed telescopes will have to incorporate the network characteristics of the individual telescopes. The individual telescopes will have different reachability characteristics from hosts to the networks, individual networks will vary in their capacity to handle event traffic load, and network characteristics like delay may cause hosts to interact in different ways with different telescopes (e.g., the rate at which hosts make TCP connections).

Another kind of telescope is an *anycast network telescope*. With an anycast telescope, multiple locations advertise routes for the same network address range prefix. An anycast telescope has many of the advantages and disadvantages of a distributed telescope, except that it does not observe a larger address range. The primary advantage of an anycast telescope over a distributed telescope is that, by advertising multiple locations for an address prefix, the telescope provides multiple routes for event traffic and that traffic will often take shorter, likely better routes to the telescope depending upon the relative location of a host and anycast telescope location in the network. As a result, the anycast telescope will distribute traffic among multiple sites and distribute load, and event traffic will often reach a telescope site more quickly.

A third telescope variation is a *transit network telescope*. A transit telescope monitors event traffic to a set of network telescope address ranges, but does the monitoring within the transit network rather than at the network edges. The key advantages of a transit telescope are that it can potentially observe traffic to a very large number of address ranges, and these address ranges can be observed centrally without the time synchronization and data distribution problems of a standard distributed telescope. However, a transit telescope also has a number of disadvantages. A transit telescope only observes all traffic to telescope address range prefixes that fall within its own network address space and its single-homed customer networks. For all other address ranges, the telescope only observes event traffic routed through its transit network. Without global estimates of overall traffic distributions, accurately characterizing events is difficult. Given these tradeoffs, a transit telescope is potentially useful for detecting the occurrence of events but not their detailed characterizations.

Finally, another kind of telescope is a *honeyfarm telescope*. Rather than passively observing event traffic, a honeyfarm telescope actively responds to some or all of the event request traffic using honeypots. Using honeyfarm telescopes introduces a number of new challenges. Since telescope address ranges can be quite large (e.g., a $/8$ prefix corresponds to 16 million addresses), honeyfarms need to decide which addresses will actively respond to traffic. Further, since many events occur simultaneously, events can occur at high rates, and event traffic is intermingled with uncorrelated background traffic, honeyfarms need to decide which traffic to respond to maximize their utility. Lastly, active response adds additional load on the network, which may interfere with event traffic, exacerbate overload conditions during event peak loads, or even be unacceptable if network cost depends on traffic load.

VI. Conclusion

In this paper we examine the utility of network telescopes for detecting and characterizing globally occurring events based on unsolicited traffic sent to a *network telescope*, a range of monitored network addresses. Network telescopes provide a unique view of global network security events that are difficult to observe via traditional single-node or end-to-end measurements. We describe the effects of telescope size on recording single host activity, including characterizing visible events, extrapolating the true start and end times of host activity and examining the factors that influence artificial flow timeouts in telescope data. We also describe the effect of telescope monitoring on infectious model curve fits to the resulting data. Finally, we detail the practical concerns involved with interpreting telescope data, including classifying monitored events, compensating for non-uniform target selection, and background network activity. We introduce next-generation network telescope designs including distributed, anycast, and transit telescopes, as well as the use of honeyfarms.

VII. Acknowledgments

We would like to thank Brian Kantor, Jim Madden, and Pat Wilson of UCSD for technical support of the Network Telescope project. Vern Paxson, Stuart Staniford and Nicholas Weaver provided valuable discussion and ideas. Finally, we thank Bill Cheswick for asking "how big of a telescope do you need" every time we saw him.

References

[1] David Moore, "Network Telescopes: Observing Small or Distant Security Events," Aug. 2002, `http://www.caida.org/outreach/presentations/2002/usenix_sec/`.

[2] Steven M. Bellovin, "There be dragons," in *Proceedings of the Third Usenix UNIX Security Symposium*, 1992, `http://www.research.att.com/~smb/papers/dragon.pdf`.

[3] Steven M. Bellovin, "Packets found on an Internet," *Computer Communications Review*, vol. 23, no. 3, pp. 26–31, July 1993, `http://www.research.att.com/~smb/papers/packets.pdf`.

[4] DShield: Distributed Intrusion Detection System, ," `http://www.dshield.org`.

[5] David Moore, Geoffrey M. Voelker, and Stefan Savage, "Inferring Internet Denial-of-Service Activity," in *Usenix Security Symposium*, Washington, D.C., Aug 2001, `http://www.caida.org/outreach/papers/2001/BackScatter/`.

[6] Jose Nazario, "Trends in denial of service attacks," `http://www.usenix.org/events/sec03/wips.html#slot8`.

[7] David Moore, Colleen Shannon, and Jeffery Brown, "Code-Red: a case study on the spread and victims of an Internet worm," in *ACM Internet Measurement Workshop 2002*, Marseille, France, Nov 2002, `http://www.caida.org/outreach/papers/2002/codered/`.

[8] Stuart Staniford, Vern Paxson, and Nicholas Weaver, "How to 0wn the Internet in Your Spare Time," in *Usenix Security Symposium*, Aug. 2002, `http://www.icir.org/vern/papers/cdc-usenix-sec02/`.

[9] Dug Song, Rob Malan, and Robert Stone, "A Snapshot of Global Internet Worm Activity," `http://www.arbornetworks.com/downloads/snapshot_worm_activity.pdf`.

[10] David Moore, Vern Paxson, Stefan Savage, Colleen Shannon, Stuart Staniford, and Nicholas Weaver, "Inside the Slammer Worm," *IEEE Security and Privacy journal*, Aug 2003, `http://www.computer.org/security/v1n4/j4wea.htm`.

[11] Jose Nazario, "The blaster worm: The view from 10,000 feet," `http://monkey.org/~jose/presentations/blaster.d/`.

[12] Herbert W. Hethcote, "The Mathematics of Infectious Diseases," *SIAM Review*, vol. 42, no. 4, pp. 599–653, 2000.

[13] Jeffrey O. Kephart and Steve R. White, "Directed-graph epidemiological models of computer viruses," in *IEEE Symposium on Security and Privacy*, 1991, pp. 343–361.

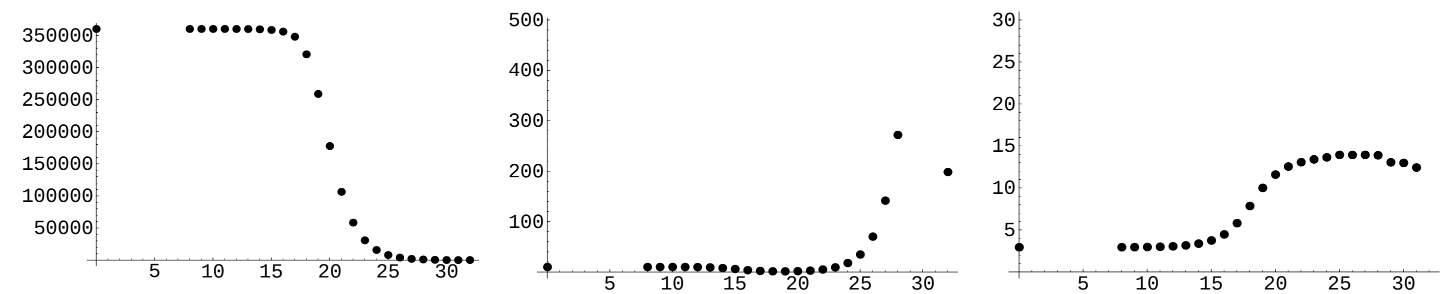
[14] Cliff Changchun Zou, Weibo Gong, and Don Towsley, "Code red worm propagation modeling and analysis," in *Proceedings of the 9th ACM conference on Computer and communications security*. 2002, pp. 138–147, ACM Press.

[15] Zesheng Chen, Lixin Gao, and Kevin Kwiat, "Modeling the Spread of Active Worms," in *INFOCOM*, Apr. 2003, `http://www-unix.ecs.umass.edu/~lgao/paper/AAWP.pdf`.

[16] Cliff C. Zou, Lixin Gao, Weibo Gong, and Don Towsley, "Monitoring and Early Warning for Internet Worms," Tech. Rep. TR-CSE-03-01, Mar. 2003, `ftp://gaia.cs.umass.edu/pub/Zou03_monitorworms.pdf`.

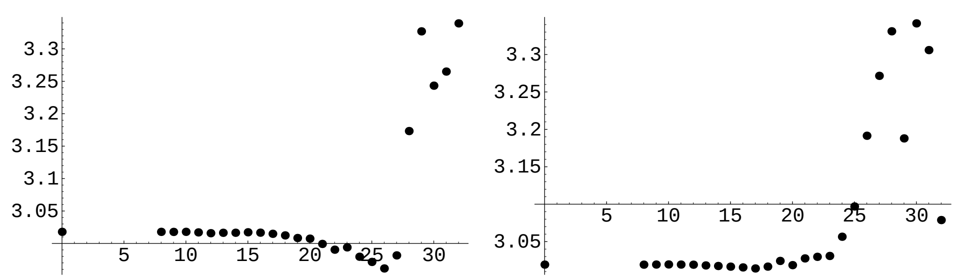
[17] Vinod Yegneswaran, Paul Barford, and Dave Plonka, "On the Design and Utility of Internet Sinks for Network Abuse Monitoring," Tech. Rep., 2003.

[18] Joel Sandin, "P2P Systems for Worm Detection," DIMACS Large Scale Attacks Workshop presentation, Sept. 2003, `http://www.icir.org/vern/dimacs-large-attacks/sandin.ppt`.



(a) N (b) r (c) T

Fig. 4. Least-squares fit of parameters N, T, and r using equation (4) to hourly bucketed counts of unique observed hosts seen at telescopes of given $/sizes$ for 24 hours using a simulation of 360,000 vulnerable hosts with a Code-Red-like worm at 10 scans per second.



(a) β (b) T

Fig. 5. Least-squares fit of parameters β, and T using equation (6) to hourly bucketed scan counts seen at telescopes of given $/sizes$ for 24 hours using a simulation of 360,000 vulnerable hosts with a Code-Red-like worm at 10 scans per second.