Practical Darknet Measurement

Michael Bailey, Evan Cooke, Farnam Jahanian, Andrew Myrick, Sushant Sinha
Department of Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, MI 48109-2122
{mibailey, emcooke, farnam, andrewmy, sushant}@umich.edu

Abstract

The Internet today is beset with constant attacks targeting users and infrastructure. One popular method of detecting these attacks and the infected hosts behind them is to monitor unused network addresses. Because many Internet threats propagate randomly, infection attempts can be captured by monitoring the unused spaces between live addresses. Sensors that monitor these unused address spaces are called darknets, network telescopes, or blackholes. They capture important information about a diverse range of threats such as Internet worms, denial of services attacks, and botnets. In this paper, we describe and analyze the important measurement issues associated with deploying darknets, evaluating the placement and service configuration of darknets, and analyzing the data collected by darknets. To support the discussion, we leverage 4 years of experience operating the Internet Motion Sensor (IMS), a network of distributed darknet sensors monitoring 60 distinct address blocks in 19 organizations over 3 continents.

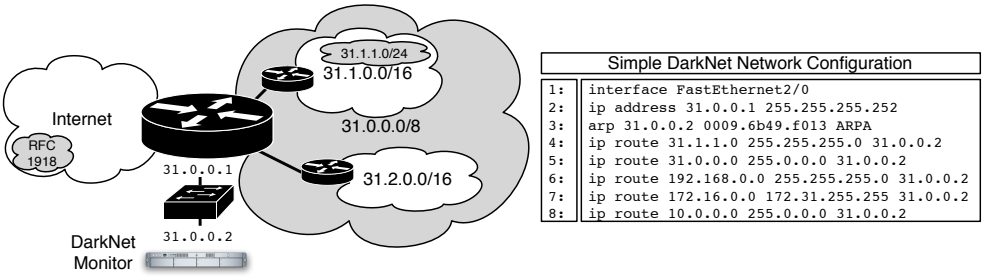
I. Introduction

Monitoring packets destined to unused Internet addresses has become an increasingly important measurement technique for detecting and investigating malicious Internet activity. Since there are no legitimate hosts or devices in an unused address block, any observed traffic must be the result of misconfiguration, backscatter from spoofed source addresses, or scanning from worms and other network probing. Systems that monitor unused address space have been called darknets [8], network telescopes [11], blackhole monitors [17], Sinkholes [9], or background radiation monitors [13], and capture important information about a diverse range Internet threats such as denial of service attacks [12], random scanning worms [3], [10], [15], [16], and botnets [7].

In this paper, we describe and analyze the important measurement issues associated with deploying darknets, configuring darknets, and analyzing the data collected by darknet monitors. The goal is to provide a general overview of darknet measurement and give researchers with the information needed to deploy and analyze the data from darknet monitoring systems. Our approach does not focus on a particular architecture and is meant to be complementary to existing work [1], [11], [19], [21].

We begin by describing how to setup a darknet and how to configure the network to forward traffic destined for unused addresses to a monitoring system. Next, we analyze data from different sized darknets to assess the storage and network resources required for darknet measurements. We next discuss how the placement of a darknet within address space and the surrounding network topology influences the visibility of monitoring systems. We also describe how visibility is impacted by the response to incoming packets. In particular, we show how no response, a SYN-ACK responder, an emulated operating system and application-level response, and a real honeypot host response represent a spectrum of interactivity that can provide additional intelligence on network events and threats. Finally, with this understanding of how to deploy and configure darknet monitors, we describe different methods of identifying important events in data collected by darknet monitors.

To inform our analysis we use data from the globally deployed Internet Motion Sensor (IMS) distributed darknet monitoring system. The IMS consists of 60 darknet blocks at 18 organizations including broadband providers, major service providers, large enterprises, and academic networks in 3 continents. It monitors over 17 million addresses which represents more than 1.25% of all routed IPv4 space.



Simple DarkNet Network Configuration

```
1: interface FastEthernet2/0
2: ip address 31.0.0.1 255.255.255.252
3: arp 31.0.0.2 0009.6b49.f013 ARPA
4: ip route 31.1.1.0 255.255.255.0 31.0.0.2
5: ip route 31.0.0.0 255.0.0.0 31.0.0.2
6: ip route 192.168.0.0 255.255.255.0 31.0.0.2
7: ip route 172.16.0.0 172.31.255.255 31.0.0.2
8: ip route 10.0.0.0 255.0.0.0 31.0.0.2
```

Fig. 1. A sample configuration which illustrates the three major darknet deployment models; capturing traffic out-bound to reserved space (lines 6-8), traffic destined to a statically configured, unused subnet (line 4), or capturing all unused space within an allocation. (line 5)

II. Darknet Deployment

The deployment of a darknet monitoring system requires an understanding of the topology of the local network. Since a darknet monitor observes traffic to unused addresses, the upstream router or dynamic host configuration server must be instructed to forward undeliverable packets to the monitor. In this section we highlight some of the important challenges associated with configuring the network and then discuss how to provision adequate storage and network resources for a darknet system.

A. Configuration

There are three general techniques for forwarding packets to a darknet monitoring system. The simplest approach is to configure the monitoring box to send ARP replies for each unused address to the router. This works well when the darknet is well-defined and spans a few addresses or when access to the upstream router is not possible. However, it is far less efficient with thousands or millions of monitored addresses. A more scalable approach is to configure the upstream router to statically route an entire address block to the monitor. This idea is illustrated in line 4 of the router configuration in Figure 1. This figure depicts a darknet monitoring setup in which the monitor is connected to a switch which is then connected to an upstream router. The use of a static route illustrated in the figure is simple but requires that darknet address block be specifically set aside for monitoring. A more flexible approach is to route all packets destined to locations that do not have a more specific address configured (and would thus be dropped) to the monitoring system by means of a blackhole (also called a fall-through route). Thus, if an organization is allocated a /8, then it could create a static route to the darknet monitor for the entire /8. Packets to valid addresses will hit more specific prefixes and only packets to unused addresses will fall through to the /8 route. This idea is also illustrated in Figure 1 and is similar to adding a route to prevent flooding attacks against persistent loops [20].

The setup thus far has assumed monitoring of unused addresses that are both globally addressable and reachable. It is also possible to monitor unused and non-routable addresses [5]. For example, RFC 1918 addresses are often used within service providers and enterprises for local systems and unused addresses in these ranges can also be monitored by darknet systems. Lines 6-8 of the router configuration in Figure 1 demonstrate how to setup static fall-through routes for the three major RFC 1918 ranges.

B. Resource Provisioning

Understanding the storage and network requirements of a darknet is critical to correctly provision the monitoring system as the amount of incoming traffic can be quite large. These requirements are typically dependent on the number of addresses monitored. To provide a general overview of the data rates observed at darknets of different sizes, we measured the packets per day per IP for various sized darknet blocks. The results are shown on the left of Figure 2. Note that the darknets that monitored less addresses tended to receive more packets per day per IP than the larger darknets. We explore these differences in more detail in the next section. On average, we found that a small /24 sensor is likely to see a sustained rate of 9 packets per second, a moderately sized /16 monitor will see roughly 75 packets per second, and a large /8 monitor over 5,000 packets per second. An important caveat that biases these results is that the /24 monitors actively responded to certain incoming connections. We found that the traffic was between 1.1 to 16 times greater on average than nearby passive /24 monitor. Details on the response are described in the next section. Another consideration is that the average rates can be deceptive because traffic routinely bursts two to three orders of magnitude above the sustained rate. For example, one IMS sensor has had a sustained rate of 9 packets per seconds over the last 2.5 years with a daily low of .6 packets per second and a daily high

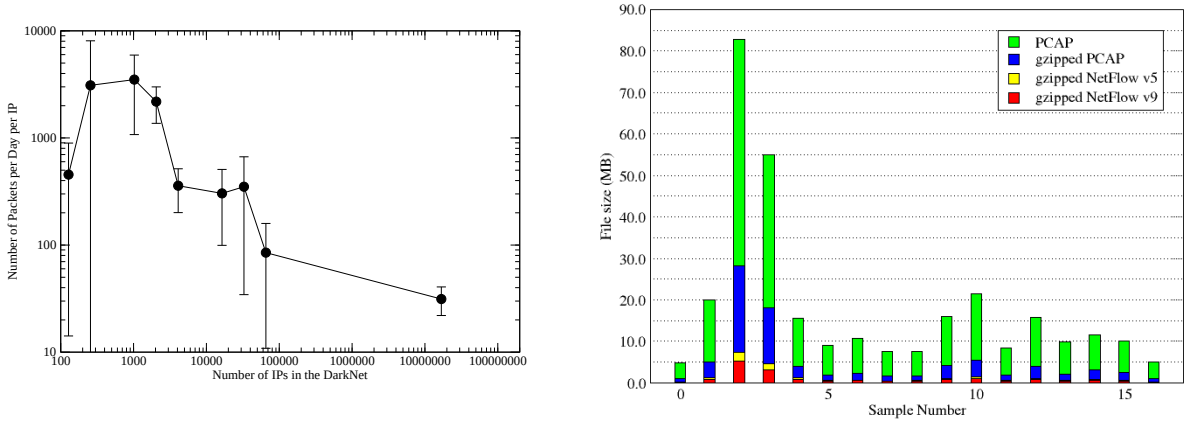


Fig. 2. The provisioning requirements for various blocks. On the left the number of packets seen per day per IP for various sized blocks. On the right the size on disk for various representations of of darknet Traffic

of 290 packets per second. The average packet size was approximately 100 bytes. The corresponding bandwidth requirements for an average /24 sensor is 7 Kbps, 60 Kbps for a /16 monitor, and 4 Mbps for a /8 monitor.

The storage format used to log incoming packets also has a large impact on the storage requirements. Common formats for collecting network traces like `pcap` and `NetFlow` are well suited for collecting darknet data. To better quantify the actual storage requirements based on different darknet sizes, we analyzed the bytes required for different storage formats (in raw and zipped format) at /16 darknet monitor over a 17 hour period. The results are shown by hour on the right side of Figure 2. The plot shows that `pcap` tends to compress very well and so keeping data files in gzip format can reduce storage requirements by more than a factor of two. The figure also shows that while flow-based representation lose important data like a the payload, they do provide excellent data reduction. There was nearly 15:1 compression when converting `pcap` to `Netflow v9`.

These measurements demonstrate that a /16 monitor can record a few months worth of data on commodity hardware with a single disk. Furthermore, by compressing or converting data into flow-based formats the storage requirements can be reduced by a factor 2 to 15.

III. Darknet Visibility Considerations

Before deciding exactly what addresses to allocate to a darknet it is important to understand how the placement of a darknet impacts what it observes. It has been shown that the malicious and misconfigured activity observed by two different but equally sized darknets is almost never the same [6]. These differences tend to depend on two

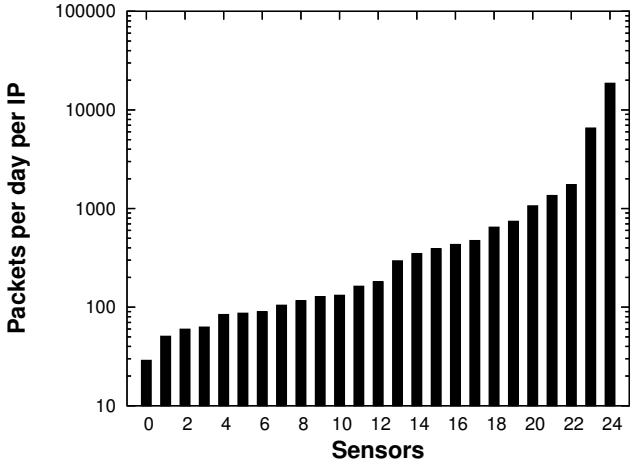


Fig. 3. Packet per day per IP at 25 different IMS darknet monitors.

important factors: the placement of a darknet, and the way in which a darknet responds to incoming packets. In this section we provide a brief overview of these two influences and describe how they impact darknet visibility.

A. Placement

Evaluating the placement of a darknet involves understanding several topological factors. One of the biggest influences on visibility appears to be vicinity to live hosts. That is, proximity in IP address space to live hosts [6]. Figure 3 shows the average packets per day per IP for 25 of the IMS sensor blocks. The values are normalized per IP to make different sized blocks comparable. The values range from 10 packets per day to more than 10,000 packets per day per IP. Of note, the darknets that observe the most

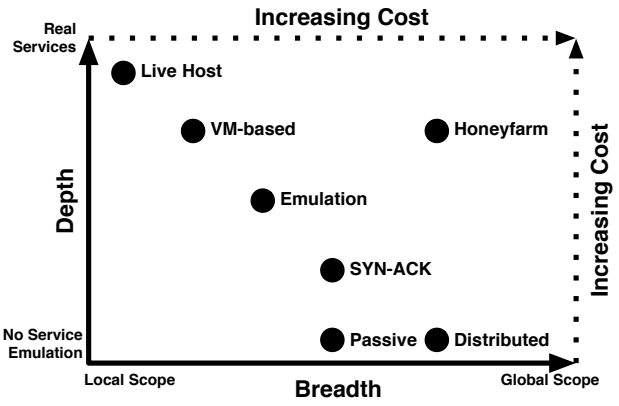


Fig. 4. Illustration of the tradeoffs between the number of addresses a darknet can monitor (breath) and the accuracy of the responses from a darknet as compared to a real host (depth). Additional resource costs are incurred attempting improve breath or depth.

traffic also tend to be smaller (/24) and are located in live networks near hosts.

The concrete reasons behind these traffic differences are still not well understood but it appears to be related to targeting behavior [6]. In particular, the preference for nearby addresses by malware and misconfigured applications. For example, Internet worms like Blaster [3] and Nimda [4] have a strong preference for nearby addresses. Another factor appears to be targeting by botnets and other attackers [7]. By targeting specific ranges of addresses that are known to contain vulnerable hosts, attackers can increase the number of systems they are able to compromise.

Another important darknet placement consideration is the location within a network. If a darknet monitor is placed behind a firewall or other infrastucture protection or filtering device, it will likely not observe externally sourced threats. On the other hand, darknets within the network can also provide important visibility into locally-scoped threats within a network. Ideally, a darknet deployment that includes monitors deployed both inside and outside network perimeters should have the greatest potential visibility.

B. Service configuration

The visibility provided by darknets is also heavily dependent on how a darknet responds to incoming packets. The simplest action is not to respond at all. A passively configured darknet simply records all the packets it observes and no further action is taken. This reveals the address of the host sending the packet and other header information. However, it may not reveal critical data like the exploit used in an attack or the details of misconfigured application requests. For example, all valid TCP transactions require a three-way handshake that must be completed before any application-level data is exchanged. This means that a passive darknet will not observe application-level data from hosts that attempt to connect via TCP.

An active response to incoming packets on a darknet can be used to collect additional application-level information to better understand an exploit attempt and better understand the intentions of an attacker. A simple but effective active response technique is to respond to TCP SYN packet with TCP SYN-ACK packets [1]. A single stateless response packet provides at least the first data packet on a TCP session and helps uniquely identify complex threats like the Blaster worm.

The first data payload may not provide enough information, so more complex responses can be used to elicit additional information. One method of generating these responses is to emulate the behavior of a real host [14], [21]. An emulated host can masquerade as a large variety of operating system and application combinations. An emulator provides the flexibility to emulate just enough of an application to acquire the needed information. However, one danger is that a malicious attacker could identity an emulated host and avoid a darknet monitor or sent it false information. The simple way to reduce the impact of this fingerprinting problem is to use a real host (i.e. a honeypot) instead of an emulated host [18].

A real host can provide complex information on an attacker and help profile the behavior, however, it can be very resource intensive. The cost of running a real host is significant and limits the number of possible monitored addresses from thousands to just a handful. One way to regain scalability is to use a pool of quickly recyclable virtual machines [19] so that multiple virtual hosts can be executed on a single physical system. Another method is to filter the connections before they reach the end hosts so that only the newer and more interesting connection are investigated [2].

Together, these different response techniques form a spectrum of interactivity that provide additional information from darknets. Figure 4 visually depicts this spectrum along the y-axis labeled depth. We define *depth* as a measure of the accuracy of the responses from a darknet when compared to the responses from a real host. On the x-axis is *breadth*. We define *breadth* as a measure of scope or number of addresses a darknet monitor can observe. This figure demonstrates the tradeoffs between scalability and fidelity and associated resource cost incurred in attempting to achieve additional breadth or depth.

A final and critical configuration decision when running a darknet with real or emulated hosts is what operating

Network /Mask	Type of Organization	Unique Hosts	TCP Implment-ations	HTTP Config-urations
A/16	university	5512	352	241
B/16	university	1289	156	73
C#1/17	webfarm	11342	256	862
C#2/19	webfarm	2438	93	293
D#1/19	webfarm	1859	118	221
D#2/19	webfarm	1652	137	208

TABLE I. Different TCP implementations and HTTP server configurations across different production networks.

systems and applications should be run/emulated. This is a very important consideration because it can be the difference between quickly identifying a new threat or missing it because the correct service was not running. Choosing appropriate services to run is far more complex than it might first appear. We conducted a survey of the services running on the University of Michigan's engineering campus and found a wide variety of operating system and application combination. A table of the different TCP stack and HTTP server implementations is shown in Table III-B. In the table, network A/16 contained 5512 scannable hosts and we found 252 unique TCP implementations, 241 unique HTTP configurations, and 1210 combinations of TCP and open and closed port configurations. This diversity means that choosing the right services to run is a complex problem for which their is currently no simple solution.

IV. Analysis of Darknet Data

Darknets can produce vast quantities of high-dimensional measurement data. Making sense of this data can be a daunting task. An entire study was dedicated to understanding the traffic observed in darknets [13]. In general, darknet traffic can be classified into four main areas:

- Infection attempts by worm, botnet, and exploit tools.
- Misconfigured application requests and responses (e.g. DNS).
- Backscatter from spoofed denial of service attacks.
- Network scanning and probing.

Generating these classifications can be complex due to scalability constraints from the huge amount of darknet data that must be processed. One method of reducing this effort is to filter the data into a smaller set of more manageable events. One simple yet powerful method is to cluster the data by source address [13], [2]. A single address may contact many different destination addresses

```
All Sources
  Total IP Packets:   57178687
  Total TCP Packets:  20173121
  Total UDP Packets:  34238463
  Total ICMP Packets: 2747341
  Unique Source IPs:  95888
  Statistics on Top 10 TCP Source IPs:
    Source IP   TCP  Pkt Cnt Top 3 Dest Ports
    X.X.56.81   229695       tcp/445:221043 tcp/8080:8652
    X.X.153.156 219602       tcp/445:219593 tcp/80:9
    X.X.199.134 162931       tcp/443:26456 tcp/80:25899 tcp/1315:1113
```

Fig. 5. Example of an IMS report based on clustering by source address.

in the darknet but will tend to perform similar behavior at each address. For example, a host infected with the Slammer worm [10] may scan tens or hundreds of thousands of destination addresses in a single day. This generates a huge amount of total traffic however it can be compressed down to a single event by grouping all that traffic by the single Slammer source address. To get a better idea of the real-world savings consider that a certain IMS darknet received 57,178,687 IP packets in a single 24-hour period. If we instead cluster that same traffic by source address we find 95,888 unique source IPs. Thus, this simple technique provides three orders-of-magnitude savings in the number of events that must be analyzed. We leverage this technique to provide daily IMS reports to operators of potentially infected systems. A clipping from a report detail the same 24-hour period described earlier is shown in Figure 5.

A. Global and Local Darknet Events

The individual events detected in darknets can usually be further divide to two locality classes. When an event such a new attack or large increase in probing occurs, it will impact a very small number of addresses (i.e., local) or the entire Internet (i.e., global). This classification only applies to the destination of an event so a local event could originate from a different network across the Internet as long it targeted a specific destination network. Figure 6 shows examples of global and local events. The left pane of the figure shows a globally-scoped attack against the MySQL service as observed by 23 IMS sensors (each color represent a separate sensor). In the right pane is a targeted RPC-DCOM attack observed in academic network containing an IMS sensor. In general, we see this bimodal distribution across many different vectors such as payload and source addresses. The implication is that attacks and other events observed in darknets are observed at only one network or are widespread and are observed at many points around the Internet.

V. Conclusion

This paper has described the important measurement issues associated with deploying darknets, configuring darknets, and analyzing the data collected by darknet

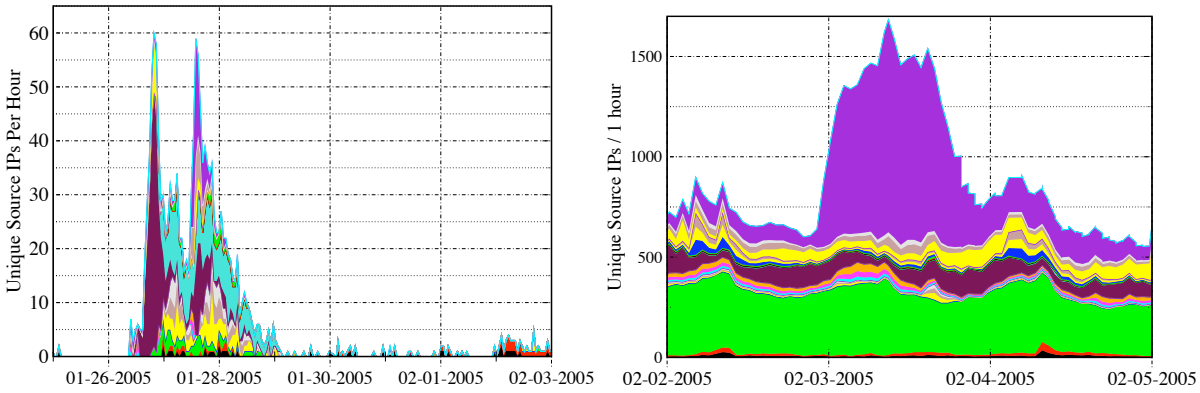


Fig. 6. The left figure is globally-scoped attack against the MySQL service as observed by 23 IMS sensors (each color represent a separate sensor). In the right figure is a targeted RPC-DCOM attack observed in academic network containing an IMS sensor.

monitors. We have attempted to provide researchers with a general overview of darknet measurement and the important details needed to deploy darknet monitoring systems. This analysis has attempted to demonstrate that building and operating a darknet monitor is a simple and productive method of gaining significant additional visibility into network threats and the state of local network and Internet as a whole.

Acknowledgments

This work was supported by the Department of Homeland Security (DHS) under contract number NBCHC040146, and by corporate gifts from Intel Corporation and Cisco Corporation. The authors would like to thank all of the IMS participants for their help and suggestions. We would also like to thank Danny McPherson, Jose Nazario, Dug Song, Robert Stone, and G. Robert Malan of Arbor Networks and Manish Karir and Bert Rossi of Merit Network for their assistance and support.

References

[1] Michael Bailey, Evan Cooke, Farnam Jahanian, Jose Nazario, and David Watson. The Internet Motion Sensor: A distributed blackhole monitoring system. In *Proceedings of Network and Distributed System Security Symposium (NDSS '05)*, San Diego, CA, February 2005.

[2] Michael Bailey, Evan Cooke, Farnam Jahanian, Niels Provos, Karl Rosaen, and David Watson. Data Reduction for the Scalable Automated Analysis of Distributed Darknet Traffic. *Proceedings of the USENIX/ACM Internet Measurement Conference*, October 2005.

[3] Michael Bailey, Evan Cooke, David Watson, Farnam Jahanian, and Jose Nazario. The Blaster Worm: Then and Now. *IEEE Security & Privacy*, 3(4):26–31, 2005.

[4] CERT Coordination Center. CERT Advisory CA-2001-26 Nimda Worm. 2001.

[5] Evan Cooke, Michael Bailey, Farnam Jahanian, and Richard Mortier. The dark oracle: Perspective-aware unused and unreachable address discovery. In *Proceedings of the 3rd USENIX Symposium on Networked Systems Design and Implementation (NSDI '06)*, May 2006.

[6] Evan Cooke, Michael Bailey, Z. Morley Mao, David Watson, and Farnam Jahanian. Toward understanding distributed blackhole placement. In *Proceedings of the 2004 ACM Workshop on Rapid Malcode (WORM-04)*, New York, Oct 2004. ACM Press.

[7] Evan Cooke, Farnam Jahanian, and Danny McPherson. The Zombie roundup: Understanding, detecting, and disrupting botnets. In *Proceedings of the Steps to Reducing Unwanted Traffic on the Internet (SRUTI 2005 Workshop)*, Cambridge, MA, July 2005.

[8] Team Cymru. The darknet project. `http://www.cymru.com/Darknet/index.html`, June 2004.

[9] Barry Raveendran Greene and Danny McPherson. Sinkholes: A swiss army knife isp security tool. `http://www.arbor.net/`, June 2003.

[10] David Moore, Vern Paxson, Stefan Savage, Colleen Shannon, Stuart Staniford, and Nicholas Weaver. Inside the Slammer worm. *IEEE Security & Privacy*, 1(4):33–39, 2003.

[11] David Moore, Colleen Shannon, Geoffrey M. Voelker, and Stefan Savage. Network telescopes. Technical Report CS2004-0795, UC San Diego, July 2004.

[12] David Moore, Geoffrey M. Voelker, and Stefan Savage. Inferring Internet denial-of-service activity. In *Proceedings of the Tenth USENIX Security Symposium*, pages 9–22, Washington, D.C., August 2001.

[13] Ruoming Pang, Vinod Yegneswaran, Paul Barford, Vern Paxson, and Larry Peterson. Characteristics of Internet background radiation. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pages 27–40. ACM Press, 2004.

[14] Niels Provos. A Virtual Honeypot Framework. In *Proceedings of the 13th USENIX Security Symposium*, pages 1–14, San Diego, CA, USA, August 2004.

[15] Colleen Shannon and David Moore. The spread of the Witty worm. *IEEE Security & Privacy*, 2(4):46–50, July/August 2004.

[16] Colleen Shannon, David Moore, and Jeffery Brown. Code-Red: a case study on the spread and victims of an Internet worm. In *Proceedings of the Internet Measurement Workshop (IMW)*, December 2002.

[17] Dug Song, Rob Malan, and Robert Stone. A snapshot of global Internet worm activity. FIRST Conference on Computer Security Incident Handling and Response, June 2002.

[18] Lance Spitzner et al. The honeynet project. `http://project.honeynet.org/`, June 2004.

[19] 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. In *Proceedings of the 20th ACM Symposium on Operating System Principles (SOSP)*, Brighton, UK, October 2005.

[20] Jianhong Xia, Lixin Gao, and Teng Fei. Flooding Attacks by Exploiting Persistent Forwarding Loops. *Proceedings of the USENIX/ACM Internet Measurement Conference*, October 2005.

[21] Vinod Yegneswaran, Paul Barford, and Dave Plonka. On the design and use of Internet sinks for network abuse monitoring. In *Recent Advances in Intrusion Detection—Proceedings of the 7th International Symposium (RAID 2004)*, Sophia Antipolis, French Riviera, France, October 2004.