Design, Deployment, and Use of the DETER Testbed

Terry Benzel, Robert Braden, Dongho Kim, Clifford Neuman
USC Information Sciences Institute
Anthony Joseph, Keith Sklower, University of California at Berkeley
Ron Ostrenga, Stephen Schwab, SPARTA, Inc.

Abstract

The DETER testbed provides infrastructure for conducting medium-scale repeatable experiments in computer security, especially experiments that involve malicious code. Built using Utah's EMULAB, the DETER testbed has been configured and extended to provide stronger assurances for isolation and containment. This paper provides information on the capabilities of the DETER testbed and discusses the lessons learned from its deployment. Our strategies for containment are described and future plans discussed.

1 Introduction

The need to defend against network attacks such as distributed denial of service, worms, and viruses requires an improvement in the state of the art of experimental evaluation of network security mechanisms. Such efforts require the development of large-scale security testbeds [8], combined with new frameworks and standards for testing and benchmarking to make the testbeds truly useful. Current impediments to evaluating network security mechanisms include lack of scientific rigor [19]; lack of relevant and representative network data [15]; inadequate models of defense mechanisms; and inadequate models of the network, background, and attack traffic data [6]. The latter is challenging because of the complexity of interactions among traffic, topology, and protocols [6,7].

Cyber-defense research has been severely limited by the lack of a public experimental infrastructure for testing new theories and new technologies in realistic scenarios. It is both unclear and unproven that technologies tested on small subnet-sized topologies modeled by a few machines will scale to realistic Internet environments.

The cyber-DEfense Technology Experimental Research (DETER) testbed [1,3] was developed to meet this challenge. The DETER testbed provides experimental infrastructure to support the development and demonstration of next-generation information security technologies. DETER provides a medium-scale facility for safe, repeatable security-related experimentation, to validate theory and simulation. The DETER testbed is implemented as an Emulab [25] cluster, using the comprehensive and powerful cluster testbed control package developed by Jay Lepreau and his colleagues at the University of Utah.

With a composition of several hundred experimental nodes, the DETER testbed provides an intermediate point between small-scale and Internet-scale experiments. Chartered to support scientific investigation, the testbed is designed for experimental repeatability, allowing experimenters to investigate, validate, and find alternative explanations for their research results and to build on the results of others. In addition to the hardware and software infrastructure needed to conduct experiments, the DETER testbed provides tools that aid the experimenters, many of which are being developed by experimenters themselves.

This paper summarizes and updates an earlier paper [1] on the DETER testbed. It presents our experience with the deployment and operation of the testbed, highlights selected projects, and discusses our plans for continued development, and expansion of the testbed facility.

2 Overview of Testbed Design

The DETER testbed is comprised of hardware -- a set of high-end PCs as experimental nodes -- and extensive control software. Flexibility and usability of the control software is critical in meeting the needs of testbed users. The expense of developing and maintaining the control software can substantially exceed the hardware cost for the testbed. Fortunately, the Utah Emulab software was available to meet most of our needs.

2.1 Testbed Requirements

A simple testbed can be constructed by manually wiring together and configuring a dedicated set of machines; however, such a testbed lacks generality and share-ability. Like Emulab, DETER belongs to the more useful class of testbeds that are *general-purpose*, *shared*, and *remotely accessible* by experimenters.

To support a large community of users, the testbed hardware can be partitioned into independent and isolated experimental testbeds, which can be used simultaneously. Just as a major particle accelerator has multiple beam-lines, so the DETER testbed supports multiple simultaneous experiments. Emulab uses high-performance VLAN-capable switches to dynamically create nearly arbitrary topologies among the nodes.

Remote accessibility for initiation and monitoring of experiments is important, but it may clash with security and containment requirements A major challenge of the

Terry Benzel, Robert Braden, Dongho Kim, Clifford Neuman, Anthony Joseph, Keith Sklower, Ron Ostrenga, Stephen Schwab, Design Deployment and use of the DETER testbed.. In Proceedings of the DETER Community Workshop on Cyber-Security and Test, August 2007, Boston.
http://clifford.neuman.name/publications/2007/200708-usecdw-deter-design-deploy/

DETER design was to allow remote access for all but the most dangerous security experiments while keeping the experiments themselves contained within the testbed.

Because DETER is intended to support security-related experiments, *containment* and *security* were basic requirements, as discussed in Section 3. Other goals for DETER were experimental *fidelity*, *repeatability*, *programmability*, and *research functionality*, as we now discuss. These goals sometimes conflict; we believe that the DETER design is an effective compromise.

- *Fidelity*

Fidelity to "real" networks, and in particular to the real Internet, is important. Dimensions to fidelity include: (1) large enough number of nodes, (2) realistic router and end-system behavior, (3) realistic heterogeneity of hardware and software, and (4) realistic mix of link bandwidth and delay.

- *Repeatability*

A central objective of DETER is to advance the science of cyber security, which requires repeatable experiments.

The dynamics of the real Internet cover a wide range of conditions, and Internet measurements vary widely in time and location. Internet topology, available bandwidth and software versions, as well as the background "attacks" and user traffic that are present, are continually evolving. It would be impossible to truly repeat a security experiment in the real Internet (even if it were prudent to conduct such experiments.)

- *Programmability*

Some DETER experiments concern new network mechanisms for monitoring, filtering, and diagnosis, which implies adding or modifying router algorithms. Router vendors are not anxious to open their platforms to experimental modifications, so the basic DETER node is programmable for this purpose. The experimenter can load specialized PC router software such as Click or Zebra [16,27]. Using software routers in DETER adds flexibility and programmability, but sacrifices fidelity. To regain some fidelity, the DETER testbed includes a small number of commercial routers that can be linked into an experimental topology.

- *Research Functionality*

The DETER testbed was built to support research in a particular topic area, security. In addition to the hardware and control software of the testbed itself, we provide a technical and social environment for security experimenters. Technical support includes a rich set of traffic and topology generators and experimental profiles, and tools for instrumentation, visualization, and analysis of results. As part of the effort, we have also developed a powerful software environment for creating, monitoring, and controlling particular kinds of security experiments.

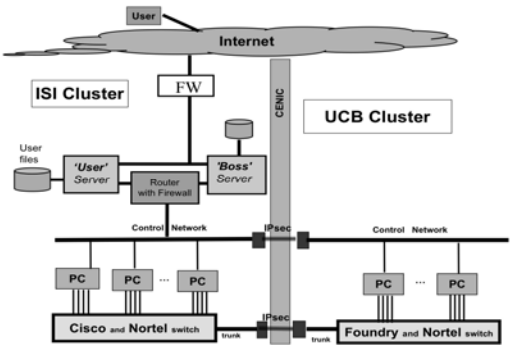

Fig. 1. Architecture of the Testbed

The social environment includes workshops and meetings to encourage collaboration and building on other's work.

2.2 Testbed Design

Figure 1 shows a simplified view of the DETER testbed architecture, based upon Emulab [25]. It shows that DETER is composed of two clusters of experimental PC nodes, at ISI and at UC Berkeley, with a common control plane. There are roughly 300 nodes in total, currently. The Emulab control software for DETER is configured to place nodes at the two sites in separate logical pools. An experiment can allocate nodes from either one or from both clusters.

These nodes are interconnected by a "programmable backplane" of high-speed Ethernet switches, trunked to form a single logical switch. Each experimental PC has four experimental interfaces and one control interface to this switch. To create the topology specified by the experimenter, the Emulab control software on the 'Boss' server allocates PC nodes to experiments and interconnects them by setting up VLANs in the switches. High-capacity switch hardware is used to avoid experimental artifacts caused by interference between VLANs.

3 Security Issues

Security is not only the object of research using the testbed; it is also a vital requirement for the testbed itself. Security for the DETER testbed is critical, and the threats are both internal and external. Internal threats come from virulent code that is tested within DETER and threatens to take control of the testbed or escape into the Internet. Additional internal threats come from experimenters who attempt to steal test data or results prior to publication. The external threats come from those who see the testbed as a tempting target for exploit; thus, infiltration protection is required.

Like any network infrastructure connected to the Internet, the DETER testbed is subject to attack; this is especially acute because a security testbed forms an attractive target. Both experiments running in DETER and the testbed

control plane must be protected. Because of DETER's mission, DETER security has a major extra component: the public Internet as well as the testbed control plane and other experiments must be protected from attack by experiments running in DETER. Whereas security in most systems is concerned only with the problem of *infiltration*, DETER is additionally concerned with the problem of *exfiltration*.

The intent of the DETER testbed is to provide containment for security experiments, to support safe experiments that present a wide range of threat levels.

The most dangerous level might be "live" testing of a contagious attack program whose attributes are completely unknown, for example an actual malicious worm or virus. The traditional approach to testing such dangerous programs has used a completely isolated laboratory consisting of dedicated systems whose disk drives and memory chips never leave the laboratory. Experimenters must be physically present in these laboratories and must be specially trained.

Not all experiments require complete isolation from the rest of the internet, and in fact, it is a goal of our effort to provide varying degrees of isolation depending on what is known about experiment that is to run. The approach of the DETER project, is to build a single safe testbed that can change its operational mode to match the threat level of the experiments. DETER provides a shared laboratory facility for those experiments whose threat level is low enough to allow sharing, but it can be reconfigured for exclusive use for more dangerous experiments. The testbed allows remote experimenter access for all but the most dangerous experiments. Another paper in these workshop proceedings [18] describes techniques under development for stronger isolation and containment of real malware whose properties are known, without the need for a complete disconnection from the rest of the internet.

The next section discusses some of the techniques used for containment of experiments that do not involve malicious code. These can be emulated worms that are not viable absent special emulating run-time hosts, as well as experiments using traffic traces and traffic generators that have the potential to consume resources but which are not themselves self propagating.

3.1 Containment

Containment addresses the need to prevent exfiltration of packets from the testbed. The worst breach of containment would be release of a previously unseen virus or worm into the public Internet. In addition to containing malicious code, the testbed contains the effects of malicious software and excessive traffic that are generated by an experiment.

The DETER testbed provides containment through several means. The first is the use of a physically separate experimental network on which the nodes of an experiment communicate. This network is unable to route packets beyond the nodes that are part of the experiment. Second, as shown in figure 1, firewalls are placed at several locations in the testbed and on the interface between the user machine and the open Internet.

The problem of containment is a little more complicated when considering dissemination of malicious code that has been running within an experiment. While active exfiltration can be prevented by the techniques just described (i.e. no route for packets to leave the testbed), malicious code can escape by hiding itself in data retrieved to the outside by an experimenter upon conclusion of an experiment.

3.2 Isolation

We provide physical link isolation to address the need to prevent experiments from interfering with one another, or for events external to the experiment or the testbed to interfere with the results of an experiment. Such interference could be unintentional, such as the overloading of a common network link by another experiment, or intentional such as from of a denial of service attack.

A programmable VLAN switch is used to map physical connections between nodes, so there is effectively no interference between links of the same or different experiments, as long as the nodes are allocated on the same switch. Because multiple switches are needed to handle the total number of nodes in the testbed, as well as to handle nodes at different physical sites, the testbed contains multiple switches connected by trunking links, some of which are used within a site, and some of which are wide area. These links may be over-subscribed by the logical links crossing them, which can cause experimental artifacts (varying performance) when a single experiment uses nodes on multiple switches. Careful monitoring of connection bandwidth is used to alert investigators if interference (or even just plain over allocation) has occurred.

3.3 Confidentiality and Integrity

The confidentiality and integrity requirements of the DETER network center around the protection of the data used by an experiment, the code and nature of the experiment, and the results of the experiment until such time as the results are published. Often an experiment will use input data such as traffic traces that are subject to non-disclosure agreements.

Confidentiality must be provided while data is resident in the staging area for an experiment (databases and file systems), in place on nodes assigned to an experiment, and while transiting the network. Confidentiality of the data while resident on the staging file system, and while it is in transit to an allocated experimental node is provided through the use of the Cryptographic File System [2].

Integrity of the data used as inputs to and produced by experiments is also critical. The integrity issue is also addressed through the use of a cryptographic file system.

3.4 Achieving Security

The security goals just described are achieved in the DETER testbed through several techniques, many of which are not unique to DETER. These techniques include:

3.4.1 Firewalls

Firewalls are deployed both externally and internally in the DETER architecture, as shown in Figure 1. The internal firewall is configured to help protect the control plane from disruptions by experiments and to prevent exfiltration of experiments. These egress filters are redundant because the topology of the testbed itself prevents such egress, but they do provide protection in case someone plugs a connector into an incorrect port.

Firewalls are also deployed extensively on the control network. Because the expected communicating pairs of machines on the control network is well constrained and the ports and protocols for such communication are known in advance, the configuration of these firewalls is very restrictive.

3.4.2 Intrusion Detection

The placement of an intrusion detection system on the network between the staging machine and the Internet, and on the control network allows us to detect traffic that should not be there. Again, because the means of interaction with experiments is well constrained, we can write relatively tight rules for anomaly detection, that have a very low rate of false positives.

We are also exploring the deployment of additional intrusion detection mechanisms to detect anomalous behavior within experiments. While activity on the control network and interface to the outside is very constrained, fewer constraints exist on activity within the experiments themselves.

To aid in detecting misbehaving experiments, we can require investigators to provide us with a characterization of the behavior of their experiment when they propose to use DETER. This characterization can be used both to determine the level of containment needed for their experiment, and to load rules into an intrusion detection system. Experiments that exceed their proposed envelope by certain bounds would be immediately suspended. If the problem was poor characterization by the experimenter, they can update their request for access and proceed, possibly under a new set of containment rules.

3.4.3 Decontaminating Nodes

Upon deallocation of an experimental node, the disks on the node can be zeroed and a new system image loaded for the subsequent experiment. This protects confidentiality of the data that was resident on the node and prevents interference with the next experiment, which could occur if changes were made to the system image.

3.4.4 Red Teaming

To help verify the security of the DETER testbed, a red team from Sandia Laboratories was contracted for a security assessment. Several exploitable vulnerabilities were found, some specific to the DETER configuration and address translation. As a result of this test, configuration changes were immediately made to the DETER testbed addressing the vulnerabilities.

3.4.5 Administration

The last technique used for protection of the DETER testbed is administrative rather than technical. Investigators seeking to use DETER must submit an application which is classified according to the potential threat the experiment poses to the testbed and to the Internet in general.

The investigator is asked to describe the potential threats resulting from a breach of containment, and explain the basis for that assessment (for example, to tell us why a simulated worm could not affect computers running beyond his or her experiment). The investigator is also asked for any confidentially or other specific security requirements.

We review proposals to assess the threat posed to the public Internet by breach of containment. We take a conservative view in our assessment of the investigators statements, considering possible errors that can be made.

3.5 Protection Domains and Federation

The DETER testbed is presently managed as a single entity, even though nodes are present both at USC and at UC Berkeley. Common policies are applied at both locations, and the interconnection between sites is accomplished through two encrypted tunnels, one on the control network and one on the experimental network.

One of the experiments run on DETER did not fit this model and required special hardware connected to the testbed through an encrypted tunnel to the user's site. In this particular case we had confidence in the investigator and allowed the departure from our normal configuration. The placement of the tunnel was such that a breach of security at the remote hardware would affect only this one experiment, although that level of containment was dependent on the proper functioning of our other defenses. We are investigating a generalization of this capability, to provide a protected *portal* from the testbed for access to special hardware running in a protected laboratory external to the DETER testbed itself. Such a model is important for support of industrial users with new types of hardware routers and security appliances.

Recently we have studied issues of federation more extensively and another paper in these proceedings [4] discusses our current approach and plans for federation.

4 Experiment Support Facilities

The DETER testbed is aimed at a relatively narrow experimental community, so it provides an opportunity to create and maintain a set of common software tools and data that support security experiments. The testbed builders and the testbed users have collaborated to build such a common set, including building blocks, a repository of complete experiments, and an integrated experimental environment.

Testbed operations assembled a library of useful building blocks, tools and data sets. The tools support measurement, data analysis, and visualization and include generators for sample topologies, network configurations, and attack and background traffic. Many of these tools were developed by DETER experimenters to ease their own work. The data sets contain standard topologies and other static information that facilitate comparative experiments using common network conditions.

The repository of complete experiments contains both very simple experiments, to help newcomers and students, and also some paradigmatic complex experiments, to serve as a basis for modification to build new experiments.

Finally, we are integrating these tools and facilities into a common experimental environment, to ease the task of creating and running a complex experiment. We refer to this environment as the "security experimenters' workbench" (SEW). The SEW will aid the assembly of a complete experiment, including topologies, generators, configurations, and monitoring tools. Built around a GUI, it provides an organized interface for monitoring and controlling execution of the resulting experiment, and it provides a powerful set of tools for analyzing the results. Finally, the SEW facilitates repeating an experiment with different parameters and algorithms. An initial model SEW, called ESVT (Experiment Specification and Visualization Tool) has been very successful [12]. Later versions will include the provision of common interface standards for experimental tools.

5 The Experimental Program

The DETER testbed has been live since March 2004. To illustrate the strength as well as the limitations of the DETER testbed for security research, this section briefly describes several research programs using DETER. We discuss DDoS, worm behavior, and BGP security experiments, as these cover a range of technical demands.

These research efforts have been aimed at experimental verification of the effectiveness and dynamics of attacks and defenses, and also at the development of methodology of these experiments. The methodology research has led to the development of libraries and tools that have been made available to other users of the DETER testbed. This work has been captured in a set of prototypical experiments (benchmarks) and associated databases of:

- topologies and topology generators
- attack & background traffic traces
- attack & background traffic generators
- special-purpose devices (e.g. meters, virtual nodes)
- metrics for scale-down, fidelity, performance, overhead.
- defenses

Many of the experiments described here were performed by researchers from the EMIST [3] project, which was funded concurrently with DETER. EMIST is a collaboration among UC Davis, Penn State, Purdue University, SRI International, ICSI, and SPARTA. The DETER user community has since expanded to include over a 100 researchers from academia and industry.

More information on many of the experiments run on deter appear in the proceedings of this workshop, and short summaries of many others appear in the proceedings of the June 2006 Deter Community Workshop. Below we very briefly describe some of the experiment areas.

5.1 DDoS Experiments on DETER

DDoS experiments on DETER have explored the dynamics and effects of DDoS attacks on complex networks. They contributed to the development of a methodological framework for analyzing the effectiveness of DDoS defense technologies [9]. This framework was refined through experiments of increasing scale and realism, using combinations of simulation, emulation on the DETER testbed, modeling, and analysis. A notational short hand was developed for describing and comparing experiments, archiving experiment descriptions, data, and results. [21]. This archive will be expanded to cover other defensive technologies and attack scenarios, and will serve as a set of resources for other DDoS experimenters, making it relatively easy for new experimenters to reuse existing software and tools to create an experiment scenario.

DDoS experiments on DETER have included:

- studies of defensive technologies, using commercial and open source software, and research prototypes;
- investigation of configuration, conduct, methodology and analysis of DDoS defense, in a rigorous setting;
- examination of two specific commercial software packages Symantec ManHunt and Network Flight Recorder (NFR) Sentivist;
- evaluation of FloodWatch [5], a traffic detection and response system using statistical profiling, as a defensive technology for defining, executing, and refining the experimental process.

5.2 Worm Behavior Experiments using DETER

Early DETER experimentation on worm behavior concentrated on modeling Internet-scale dynamics of worm propagation. Since it is not possible to perform a truly Internet-scale experiment, scale-down is critical for experiments on worm behavior.

Early worm behavior research on DETER included:

Development of two models for scanning worms [24]: the homogeneous cluster model and the heterogeneous cluster model., one representing a 1/64 scale emulation of the Internet.

A 1000 (virtual) node enterprise network simulations of the Slammer worm [13], and Witty and Blaster worms [14]: This included development of virtual nodes that model the response of subnetworks to a worm attack for the purposes of studying scale-down. This experiment also employed a visualization tool that has been integrated into the DETER testbed for use by other researchers [12].

Development of an abstract data model called the Internet Worm Propagation Data Model (IWPDM) [11] and WormGen, a safe attack generation system. In order to expand the test to a larger number of nodes, each physical node on the DETER testbed hosted four WormGen agents, using the four network interfaces on each physical node.

A number of experimenters using the DETER testbed have reported significant results through the use of the testbed. A researcher from ICSI reported recently that he discovered a problem in his worm emulation model as a result of running source models and emulated worms on the testbed.

One researcher discussed the role that fidelity plays in the accuracy of models [22]. It is expected that an increase in model accuracy resulting from the addition of nodes, and reduction of testbed artifacts will lead to increased understanding of worm behavior opportunities for new research on worm containment and prevention techniques.

5.3 Experiments with Live Malicious Code on DETER

In running early experiments with self propagating malicious code [17], we sought to force ourselves as testbed operators to put in place and exercise the protections necessary for running such code. We wanted to choose a virus or worm that was well known, and to which defenses were long since deployed outside the testbed, in case our procedures failed containment. We also wanted to run an experiment that would yield data that was useful to others, so that the experiment was not being run solely for the experiments sake.

Our choice of malicious code was the Scalper worm [5], a worm that has been circulating on the Internet for several years, and for which most machines have already been patched. As we learned when running the experiment not only was the particular worm exploit patched in recent and not so recent versions of Apache, but recent changes to FreeBSD also made the self propagation of the worm no longer viable.

Our experiments with live Malicious Code allowed us to generate realistic traffic traces for worm propagation for use as data sets for other experiments.

5.4 Early Routing Security Experiments in DETER

Preliminary research on BGP routing attacks [23,26] has:

- Evaluated the ability of security mechanisms such as Whisper/Listen, SBGP, and SoBGP to defend the Internet routing infrastructure against malicious attack.
- Demonstrated two types of BGP attacks: OASC (Origin AS Changes) and DDP (Differential Damping Penalty). The experiments provided data that could be used to compare their strength, weakness, performance, and the effectiveness of several proposed approaches to handle attacks toward the routing infrastructure.
- Examined signature- and statistics-based detection to search for anomalous BGP routing dynamics.

Experimenters proposed two approaches to managing this analysis and identifying advantages and limitations of each. Their study is currently limited by the lack of data from real environments, but the use of the DETER testbed supports examination of BGP on a larger scale.

6 Lessons Learned

We have learned that the needs of users vary more significantly than originally expected. Security experiments tend to be larger than other experiments because the effects of attacks are often not felt until a large number of end machines have been compromised. Some experiments require large numbers of nodes, many more than we can provide physically, so even with 300 nodes, support for virtual nodes was important. The use of virtual nodes, however, introduces artifacts in the experimental results that must be considered.

We also found that some experiments required the ability to employ topologies of specific commercial routers which had not been previously incorporated into DETER. This introduced new requirements for testbed design to enable investigators to plug in hardware modules, in this case, separately from the testbed and interconnected through an encrypted tunnel, and to allow those modules to be allocated only for specific experiments.

We believe that the portal concept design can be extended to other inter-testbed connections and integrated into the DETER control plane. However, it should be noted that such connections require the use of dedicated nodes, thereby reducing the total number of nodes available for experimentation. If the testbed is to support more portals then more dedicated nodes will be needed for this use.

Many of the experiments that run on DETER do not require the most secure mode of containment. Basic containment is needed to keep the effects of an error in the traffic and attack generators from causing problems beyond the confines of the testbed, but because the code to be tested is written by the investigators, the incentive for breach of containment is just not there. However, the ability to experiment with wild malicious code is important for some experimenters, and the red teaming experiments were helpful in moving us toward that ability. The ability to run large scale simulations and emulations employing a mix of physical and virtual topologies can provide a mechanism for exploring "what if" scenarios and evaluating response options in the face of critical infrastructure attacks.

7 Conclusion

The DETER testbed has been operational since March of 2004 and is used by researchers to perform experiments on worm propagation, distributed denial of service attacks, and routing and infrastructure attacks. At the time of writing, the testbed had more than 300 nodes and it has been used by commercial and academic researchers to study attacks and assess the benefit of products in development.

The testbed provides investigators with the ability to run experiments using potentially risky code, on an isolated experimental network. For most categories of experiments, control is possible remotely by connecting to a testbed user machine through the Internet. Firewalls, intrusion detection systems to monitor access, and other safeguards protect access through this control network, and physical separation from the Internet is provided on the experimental network on which the experimental nodes communicate.

Support for testbed users includes a repository of attack traffic generators, monitoring tools, topology generators, and other tools, and work is underway to integrate these tools into an experimenters' workbench which will simplify the task of getting new experiments up and running.

Over time we expect to see replicas of the DETER testbed, collectively supporting a larger user community, with varying requirements for containment, confidentiality, the number of nodes, and performance. We will explore ways to federate such independently managed testbeds to enable larger experiments to run than can be supported on a single testbed. We have experience running the DETER cluster across two sites, USC and Berkeley, however, we have managed these sites as a single domain with common security requirements. Federation of testbeds is a much more complicated problem which must take into account differences in the policies enforced at the different endpoints, differing levels of containment, diversity in the user communities and the level of trust one places in the management of the independent clusters.

The DETER testbed provides a venue for investigators to run experiments that require containment from release to the open Internet. The testbed provides an environment that makes experiments more readily repeated and validated by others, and serves as a repository for the data and hardware and software configurations used for experiments. For more information please visit http://www.isi.edu/deter.

8 Acknowledgements

This research was supported by funding from the United States National Science Foundation and the United States Department of Homeland Security under contract numbers ANI-0335298 (DETER) and CNS-0454381 (DECCOR), and by the Department of Homeland Security, and Space and Naval Warfare Systems Center, San Diego, under Contract number N66001-07-C-2001. Opinions, findings, conclusions and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation (NSF), the Department of Homeland Security, or the Space and Naval Warfare Systems Center, San Diego. Juniper Networks and Hewlett-Packard donated equipment used by the DETER testbed. Donations were also received from Sun Microsystems and Dell through their University Discount programs.

9 References

[1] Benzel, Terry, Bob Braden, Dongho Kim, Clifford Neuman Anthony Joseph and Keith Sklower Ron Ostrenga and Stephen Schwab, *Experience with DETER: A Testbed for Security Research.* Second IEEE Conference on testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCom2006), March 2006, Barcelona.

[2] Blaze, Matt, A Cryptographic File System for UNIX, *ACM Conference on Computer and Communications Security*, pp. 9-16, 1993.

[3] Members of the Deter and EMIST Team, Cyber Defense Technology Networking and Evaluation, Communications of the ACM 47(3), March 2004, pp 58-61.

[4] Faber, Ted, John Wroclawski, and Kevin Lahey, A DETER Federation Architecture In Proceedings of the DETER Community Workshop on Cyber-Security and Test, August 2007, Boston.

[5] Feinstein, L. et al. *Statistical Approaches to DDoS Attack Detection and Response*. In Proceedings of the Third DARPA Information Survivability Conference and Exhibition (DISCEX III). 1 (Apr. 2003) 303-314.

[6] Floyd, S. and Kohler, E. Internet research needs better models. *Hotnets-I* (Oct. 2002).

[7] Floyd, S. and Paxson, V. Difficulties in simulating the Internet. *IEEE/ACM Transactions on Networking 9*, 4 (Aug 2001), 392–403.

[8] Hardaker, W. et al. Justification and Requirements for a National DDoS Defense Technology Evaluation Facility. Network Associates Laboratories Report 02-052, July 26, 2002.

[9] A. Hussain, J. Heidemann, and C. Papadopoulos, “A Framework for Classifying Denial of Service Attacks,” In Proceedings of *SIGCOMM 2003*

[10] Hibler, M., L Stoller, J. Lepreau, R. Ricci, and C. Barb. *Fast, Scalable Disk Imaging with Frisbee.* Usenix 2003.

[11] Levitt, K. et al. *Using a Worm Propagation Data Model for Safe Attack Generation Systems.* In Proceedings of CCS Workshop on Worm Behavior, ACM Conference on Computer and Communications Security. (Oct. 2004).

[12] L. Li, S. Jiwasurat, P. Liu, G. Kesidis, EMIST Experiment Specification and Visualization Tool: The User Manual, October, 2004. Code at: http://emist.ist.psu.edu/ESVT2/download_esvt2.html

[13] Li., L. et al. Worm Propagation in Enterprise Networks Emulated on the DETER Test-bed. *Technical Report, School fo IST and CSE Department, Penn State University* (June 2004).

[14] Li., L., I. Hamadeh, S. Jiwasurat, G. Kesidis, and P. Liu. Emulating sequential scanning worms on the DETER testbed. In the *2nd International IEEE Conference on Testbeds and Research Infrastrucutres for the Development of Networks and Communities*, Barcelona, (March 2006).

[15] McHugh, J. Testing intrusion detection systems: A critique of the 1998 and 1999 DARPA intrusion detection system valuations as performed by Lincoln Laboratory. *ACM Transactions on Information and System Security 3*, 4 (Nov. 2000), 262–294.

[16] Morris, Robert, Eddie Kohler, John Jannotti, M. Frans Kaashoek. The Click modular router, *Symposium on Operating Systems Principles*, pp 217-231. 1999.

[17] Neuman, Clifford, Chinmay Shah, Kevin Lahey. *Running Live Self-Propagating malware on the DETER Testbed.* Proceedings of the DETER Community Workshop, Arlington VA, June 2006.

[18] Ostrenga, Ron and Stephen Schwab, Robert Braden, A Plan for Malware Containment n the DETER Testbed. In Proceedings of the DETER Community Workshop on Cyber-Security and Test, August 2007, Boston.

[19] Pawlikowski, K., Jeong, H., and Lee, J. On credibility of simulation studies of telecommunication networks. *IEEE Communications Magazine* (Jan. 2001).

[20] Porras, P., L. Briesemeister, K. Skinner, K. Levitt, J. Rowe, and Y. Ting. A Hybrid Quarantine Defense. In Proceedings of Worm’04 (October 2004).

[21] Schwab, S., B. Wilson, R. Thomas, “Methodologies and Metrics for the Testing and Analysis of Distributed Denial of Service Attacks and Defenses,” MILCOM, Atlantic City, NJ, Oct. 2005.

[22] Sewani, A., “A System for Novel Email Virus and Worm Detection,” Masters Report, University of California, Electrical Engineering and Computer Science department, August 2005.

[23] Teoh, S. et al. Combining Visual and Automated Data Mining for Near-Real-Time Anomaly Detection and Analysis in BGP. In Proceedings of CCS Workshop on Visualization and Data Mining for Computer Security, ACM Conference on Computer and Communications Security (Oct. 2004).

[24] Weaver, N. et al. Characterization, Modeling and Scale-down Simulation of Slammer's Propagation in the Internet. *In Proceedings of CCS Workshop on Worm Behavior, ACM Conference on Computer and Communications Security* (Oct. 2004).

[25] White, B., J. Lepreau, L.Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb, and A. Joglekar. An Integrated experimental environment for distributed systems and networks. In *Proceedings of the Fifth Symposium on Operating Systems Design and Implementation (OSDI02)*, (Dec. 2002). Pp 255-270.

[26] Zhang, K. et al. On Detection of Anomalous Routing Dynamics in BGP. *Networking 2004*, 259-270.

[27] The Zebra Router. http://www.zebra.org.