CERT

CERT RESEARCH ANNUAL REPORT

2007

Software Engineering Institute | Carnegie Mellon

CERT

Software Engineering Institute
Carnegie Mellon

The primary goals of the CERT® Program are to ensure that appropriate technology and systems management practices are used to resist attacks on networked systems and to limit damage and ensure continuity of critical services in spite of attacks, accidents, or failures.

CERT is part of the Software Engineering Institute (SEI), a federally funded research and development center (FFRDC) sponsored by the U.S. Department of Defense and operated by Carnegie Mellon University. The SEI advances software engineering and related disciplines to ensure systems with predictable and improved quality, cost, and schedule.

This report describes how CERT research advanced the field of information and systems security during the 2007 fiscal year.

To download a PDF version of this annual report, go to
http://www.cert.org/research/2007research-report.pdf

TABLE OF CONTENTS

Research Areas

Additional Research Activities

Researcher Activities

CERT Research Vision

This year CERT begins its 20th year of service and research. For the past 20 years we have provided support to the nation and the global community, to help create a collaborative international infrastructure dedicated to improving security across the interconnected world. The successes we have attained have been based on understanding security threats and vulnerabilities, creating practical solutions to security problems, and transitioning these solutions to government and industry.

We envision and are working toward a safer computing world. This will require more knowledgeable and better trained people building better systems that benefit from better systems management. Improved practices and technologies must be widely understood and routinely used to protect, detect, and respond to attacks, failures, and accidents on networked systems.

Our research is focused on four principal objectives:

- Embed software and system assurance techniques in all aspects of the system development life cycle.
- Improve the effectiveness of the international intrusion analysis and response team community.
- Develop an international workforce skilled in secure cyber operations.
- Improve the survivability and resiliency of critical networked information systems.

.

Accomplishing these objectives requires our own best efforts, as well as cooperation and collaboration within the community we serve. We strive to understand emerging security challenges, and work to transition disciplined improvement approaches such as the Resiliency Engineering Framework, security assessment methods, and threat modeling and mitigation techniques. We continually look forward to anticipate risks and threats to future systems.

Evolving information and communication technology, growing interconnection of complex systems of systems, and the dynamic nature of cyber threats bring increased risk, and with it the need for more powerful mitigation strategies. Our research programs are dedicated to understanding network behavior, identifying best security practices, and developing foundations for security architecture, design, and development. We foster research that provides a solid basis for addressing challenges facing practitioners and educators who are seeking to improve the current state of practice. We also address root causes of security problems through work in engineering methods and tools for developing software with fewer vulnerabilities and improved security properties.

We are researching new methods and tools to monitor and analyze the behavior of complex networks, and new forensic techniques to investigate the sources and effects of novel forms of attack. Our intent is to transition these sensors and analysis tools to the security community. The goal is improved awareness of security compromises, faster response to and recovery from attacks, and increased success in identifying and prosecuting attackers. We also focus on improved system management and enterprise risk management methods, to provide system operators and owners with better control and increased resiliency in responding to security problems.

As we begin our third decade, we will continue to promote and rely on cooperation and collaboration within the security community. We welcome participation by organizations that share our belief that the networked environment of the future can provide safe and secure computing for all participants in the global information community.

Rich Pethia
Director, CERT
Software Engineering Institute
Carnegie Mellon University

Executive Summary

The work of the CERT Program at Carnegie Mellon University's Software Engineering Institute includes technologies and methods for

- eliminating security flaws and vulnerabilities in systems
- preventing intrusions from occurring
- identifying intrusions that have occurred
- preserving essential services when systems have been penetrated and compromised
- providing decision makers with information required for network defense

We recognize the importance of multiple strategies for prevention and detection of and recovery from cybersecurity attacks, and the CERT Program has been designed to address a broad spectrum of security technology research, development, and transfer.

In our research activities, the goal is to replace informal methods with precise software and security engineering. In our technology development work, we create software and security standards, technologies, and automation. In technology transfer, we work with clients to incorporate results into key acquisition and development projects. We also provide training and materials, such as books and articles, to support technology transfer.

While all these elements are necessary to achieve success, the focus of this report is on CERT's research work. Our research agenda is driven by the need to develop theoretical foundations and engineering methods to help ensure the security of critical systems and networks. We believe the projects described in this report are essential elements of this agenda. Abstracts are provided here for our major research projects, followed in the report by more detailed descriptions of the projects. Additional research activities, publications, and technical leadership activities are also described.

Building Server Inventories and Detecting Rogue Servers in Network Flow Data

Accurately inventorying servers in a large-scale network is a daunting task. Due to their size, active probing of such networks is infeasible. Here we demonstrate a method of passively inventorying common server types based on a history of activity, naming convention, and expected service. Furthermore, we demonstrate the use of an inventory in detecting rogue servers.

Estimating Phishing Populations from Reports

We estimate the extent of phishing activity on the Internet via capture-recapture analysis of two major phishing site reports. Capture-recapture analysis is a population estimation technique originally developed for wildlife conservation, but it is applicable to any environment in which multiple independent parties collect reports of an activity. Generating a meaningful population estimate for phishing activity requires addressing complex relationships between phishers and reports. To address these relationships, we estimate populations in terms of netblocks and by clustering reported phishing records together into *scams*, which are records that demonstrate similar behavior on multiple axes. We generate population estimates using data from two different phishing reports over an 80-day period.

Insider Threat Modeling and Analysis

Insiders can be current or former employees and contractors who have or had authorized access to their organization's system and networks, and who are familiar with internal policies, procedures, and technology. They can exploit that knowledge to facilitate attacks and even collude with external attackers. CERT's research focuses on both technical and behavioral aspects of actual malicious insider acts, including espionage, theft of confidential or proprietary information, IT sabotage, fraud, and potential threats to our nation's critical infrastructures. We produce models, reports, training, and tools to raise awareness of the risks of insider threat and to help identify the factors influencing an insider's decision to act, the indicators and precursors of malicious acts, and the countermeasures that will improve the survivability and resiliency of the organization.

Network Traffic Visualizations for Portal Environments

Grasping global changes in network activity is hindered by perspective. Using raw data dumps provides insight into specific activity, but in many cases context is lost. Visualization of flow data provides analysts with a compact way of demonstrating changes in the network, as well as an expectation of future activity. However, a single visualization is unable to sufficiently capture phenomena at every resolution. Therefore, we present three visualizations that provide insight into sensor health, global connectivity, and host-level activity.

Resiliency Engineering Framework

Organizations tend to manage operational risk activities such as security, business continuity, and IT operations in a reactive posture encumbered by over-reliance on heroes, stove-piped organizational structures, and poorly defined and measured goals. This affects the organization's operational resiliency and its ability to manage operations in constantly changing and complex risk environments. The objective of CERT's resiliency engineering research is to define the convergence of these operational risk activities in a process improvement model—the CERT Resiliency Engineering Framework—and provide a roadmap for organizations to objectively measure and improve their resiliency engineering processes. With the focus provided by the REF, organizations can redirect scarce resources and improve capabilities for adapting to change and yet-to-be-realized disruptions with agility and confidence.

Spam Detection at the ISP Level

The current state of practice in defending against unwanted email (spam) involves use of endpoint-centric solutions such as Bayesian filtering. As spam volume rises and the spammer community shifts behavior to avoid detection, we expect the effectiveness of these solutions to decrease. We have developed a method for detecting sources of spam rather than attempting to detect individual spams or to trace spammers. The method is a payload-agnostic, flow-based mechanism for detecting hosts that send suspected spam. It distinguishes normal email use from uses that are in a spammer's interest. Our results are empirically validated using traffic logs from active hosts on approximately 5% of the Internet public address space. During 2007, we constructed and validated the method. During 2008, we plan to initiate pilot implementations.

SQUARE: Requirements Engineering for Improved System Security

Through the SQUARE project, CERT researchers have developed an end-to-end process for security requirements engineering to help organizations build security into the early stages of the production life cycle. The SQUARE methodology consists of nine steps that generate a final deliverable of categorized and prioritized security requirements. The process has been baselined, and transition to real-world clients has shown that the methodology can be incorporated into industry practice. A prototype tool and educational and training materials have been developed for SQUARE.

STAR*Lab: A Software Development Laboratory for Security Technology Automation and Research

STAR*Lab is an internal software development laboratory that CERT has established to create theory-based prototype automation to challenge problems in security and software engineering. STAR*Lab is engaged in the Function Extraction project and is ready to expand the technology in four projects: Computational Security Attributes, Software Correctness Verification, System Component Composition, and Flow-Service-Quality Engineering, as described below.

*STAR*Lab Function Extraction for Software Assurance: Engineering Automation for Computing Software Behavior*

STAR*Lab recognizes the importance of software assurance to national defense. Software assurance depends on knowing and verifying the complete behavior of software because behavior that is not known can contain errors, vulnerabilities, and malicious content. To help address this need, STAR*Lab is conducting research and development on the emerging technology of function extraction (FX). The goal of this project is to compute the behavior of software with mathematical precision to the maximum extent possible. Computation of the behavior of malicious code is of particular interest, to help analysts quickly determine intruder objectives and develop countermeasures.

*STAR*Lab Computational Security Attributes: Engineering Automation for Software Security Analysis*

In the current state of practice, security properties of software systems are often assessed through labor-intensive human evaluation. These *a priori* evaluations can be of limited value in the dynamics of system operation, where threat environments can change quickly. This project focuses on automated analysis of the security properties of software. The goal is to develop foundations to help transform security engineering into more of a computational discipline.

*STAR*Lab Software Correctness Verification: Engineering Automation for Software Assurance*

In the current state of practice in software engineering, no practical means exists for automated, large-scale correctness verification of software with respect to intended behavior. As a result, much time and energy is devoted to inspection and testing activities that can provide only limited evidence of correctness. The objective of this project is to develop a proof-of-concept prototype of a function verification system that will analyze the correctness of programs.

*STAR*Lab System Component Composition: Engineering Automation for Understanding System Behavior*

Modern systems are characterized by large-scale heterogeneous networks with many components that must be correctly integrated to achieve mission objectives. System integration today is a complex, labor-intensive process that can take months or even years for large systems. The objective of this project is to develop a proof-of-concept prototype of a component composition system that will help determine the net effect of combining components in network architectures for faster integration.

*STAR*Lab Flow-Service-Quality (FSQ) Engineering: Foundations for Developing Network-Centric Systems*

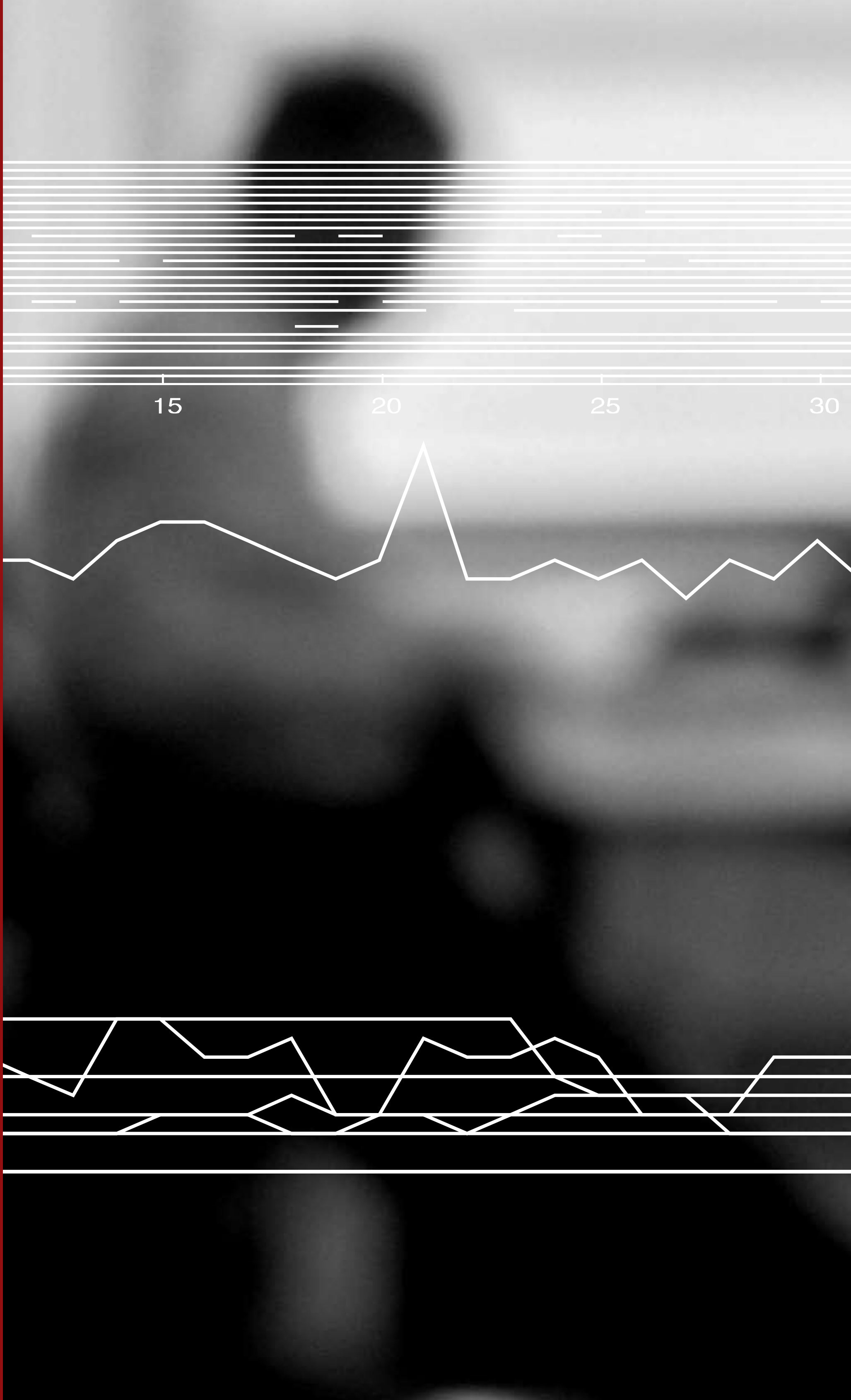
Large-scale, network-centric systems are often characterized by massively distributed components, lack of global visibility, uncertain function and quality, ever-changing boundaries and user groups, permanent risk of intrusion and compromise, and extensive asynchronous operations. The objective of the FSQ project is to create rigorous engineering foundations for development and evolution of such systems under intellectual control. Flow structures, a key element of FSQ engineering, refine mission objectives into designed compositions of network components while accommodating the operational uncertainties of the underlying systems.

Survivability Analysis Framework

The Survivability Analysis Framework (SAF) was developed through research initially focused on assurance analysis methods applicable to increased demands for system interoperability, integration, and survivability. The complexity that arises in networked systems of systems is an aggregate of technology, scale, scope, and operational and organizational issues. The research effort has expanded to address analytical capabilities for service-oriented architectures (SOA) and the integration of shared services with organizational missions. The SAF process analyzes mission threads in end-to-end workflows and identifies potential stresses that could limit completion of the workflows. This approach permits analysis of interrelations among people, processes, and technology to identify critical patterns for system simplifications and reductions in potential mission failures.

Building Server Inventories and Detecting Rogue Servers in Network Flow Data

Jeff Janies
412-268-8225



Building Server Inventories and Detecting Rogue Servers in Network Flow Data

Problem Addressed

Rogue servers are hosts that provide a public service without the network administrator's consent. They are commonly the result of a misconfiguration or an exploit. For instance, the presence of an SMTP service on a machine that is not designated as a mail server provides an avenue by which malicious parties could relay spam messages or use the host as a stepping-stone to compromise others. Furthermore, the utilization of a compromised host to relay spam messages damages the reputation of the surrounding netblock by increasing the probability of the netblock being blacklisted.

In order to detect rogue server activity, a network administrator must have ground truth with regard to which hosts actually provide a service of interest. However, in the case of large-scale networks (on the order of one or more /8 CIDR blocks), establishing ground truth is daunting. Large-scale networks have a variety of network administration groups and policies, making centralized understanding of the network topology difficult. Furthermore, the configuration of each network can vary greatly, giving rise to unpredictable trends in host behavior and bandwidth utilization.

One method of mitigating the inevitable confusion such a diverse network landscape generates is to develop inventories of internal hosts' services. Such an inventory provides the administrator with a level of expectation for each host in the inventory. However, inventories are difficult to generate and maintain. Classically, active probing has played a key role in producing such inventories, but this method does not scale to large networks. Opposed to active probing, we propose a passive, payload-agnostic method for discovering servers of various types.

Research Approach

We have examined four server types: mail, DNS, VPN, and web. We base our methods on the principle of stability in core network infrastructure components. Legitimate servers have a high likelihood of continual server-like behavior and a propensity towards performing as a server. Additionally, administrators tend towards structured naming conventions. As a result, our classification strategy is twofold. First, we classify according to *administrative* characteristics. We define an administrative characteristic as any attribute associated with a host that provides insight into the administrator's intent for the host. Here we focus on host name and DNS records. Second, we classify according to *behavioral* characteristics, i.e., observable activity of the host.

Using both of these characteristics, we develop an initial inventory for each type of service. This initial inventory comprises all easily identifiable servers. Even though a majority of legitimate servers are easily detected with this strategy, there are servers with sparse communication patterns. Therefore, we augment the inventory with conclusions drawn from historical evidence of server activity.

Since historical evidence requires long-term trending of individual server behavior, developing historical evidence is a more intensive process than measuring administrative and behavioral characteristics. However, historical evidence is a very strong indicator of legitimate activity. For instance, if a host lacks a history of sending and receiving communications using SMTP, it is less likely that the host is performing as an authorized mail server and more likely to be a compromised or misconfigured host. Conversely, a history of mail server activity provides evidence for the host's inclusion in the inventory. To limit both the processing time and storage requirements of a historical record for each suspected server, we only investigate servers that we did not detect with administrative and behavioral characteristics.

Combining these two methods, we have an iterative process that generates accurate server inventories. Once the inventory is established, the analyst can assume that all un-inventoried hosts providing a particular service have a high likelihood of doing so illegitimately. Furthermore, by inventorying, we gain a better understanding of the actual topology of the network and are better equipped to deal with emerging threats.

Expected Benefits

We expect that having accurate inventories will provide analysts with the ability to better manage rogue server detection. First, it allows the analyst to filter the inventoried hosts from investigation, thus decreasing the number of hosts examined. Second, the level of expectation for each inventoried host is established and deviations are detectable.

2007 Accomplishments

Of the four services investigated, we expand here on mail server inventories and their use in detecting rogue servers. A mail server is a host that listens and participates in meaningful communication on port 25. Since the characteristics of the communication are different, we distinguish between hosts running listening services on port 25 (SMTP listeners) and hosts that communicate with external mail servers (SMTP talkers). In general, SMTP listeners are primarily associated with incoming mail, while SMTP talkers, like relays, are associated with outgoing mail.

We begin by developing an initial inventory of SMTP listeners based on the administrative and behavioral characteristics. We test these characteristics against hosts that have complete TCP/IP sessions on port 25. Assume that a complete TCP/IP session contains more than 120 bytes and more than three packets per direction, and that the OR of the TCP/IP flags for the whole session contains SYN, ACK, and FIN. If a host's name contains the substrings "smtp," "mx," or "mail," we assume the host is intended to function as an SMTP server. However, if the host's name does not contain these substrings, we evaluate it based on the ports that the host utilizes. By inspection, we have observed that 85% of the TCP/IP flows from a legitimate server are associated with mail protocols, such as SMTP, POP3, IMAP, and ports associated with authentication. If the percentage of mail server activities versus total activities exceeds this threshold, we add the host to the inventory. We then augment the inventory by adding observed SMTP talkers (hosts with communications to destination port 25) that have names consistent with mail servers.

To account for varying usage over time, we assume that given a large enough observation window, we will see all legitimate servers servicing requests. For the purposes of the initial inventory, we use a window size of seven days. This observation window accounts for weekly seasonal trends in usage but not the existence of rarely used mail servers.

Figure 1 shows the activity of the initially detected mail servers on nine independently administered subnets over the course of one month. The top part of the figure is an existence plot, where each y-value represents a unique host and each subnet is grouped by color. The lines along the x-axis represent observed activity with a minimum resolution of one day. If mail server activity was observed on a given day, the activity is represented by a line. Otherwise, the area is blank. The existence plot is accompanied by a time series (the lower part of Figure 1) that summarizes the number of observed servers per subnet. This time series demonstrates a relative consistency in number of active mail servers.

As the figure shows, a majority of the servers have highly seasonal components to their activity, with two exceptions. First, all of the servers on two subnets disappear after approximately a week. This is the result of the network sensors going completely offline. However, the subnets are included here as examples of the regularity that mail servers demonstrate. For every day that we can observe, these servers are active. Second, one subnet has eight servers that appear only once. These hosts are legitimately listening on port 25 and fell victim to vulnerability scans. As a result, we removed them from the inventory. Excluding these exceptions, we see that any seven-day observation period will contain nearly all of the servers listed.

Once we have established this initial inventory of mail servers, we develop a daily archive of un-inventoried hosts' activities to and from port 25. Over time, we manually expand the inventory by adding observed SMTP talkers that have a history of activity. We note that many non-mail servers may still communicate via SMTP using local Message Transfer Agent (MTA), appearing as an SMTP talker in our inventory. Therefore, we take liberty with the term *SMTP talker* to include any host that uses port 25 consistently and for explainable reasons.

By augmenting the inventory with servers discovered through historical evidence, we greatly increase the number of inventoried hosts, in some cases as much as 50 percent. This is primarily due to our distinction of SMTP talkers and listeners and the fact that we evaluate only the characteristics of listeners in the creation of the initial inventory. However, we found that after examining one month of historical information, the number of unique new talking mail servers decreased to nearly zero for all subnets examined. Thus, we can expect that the number of hosts in the inventory will approach a stable number.

2008 Plans

In the coming year, we will expand on this work in two ways: developing rogue server detection techniques for other server types and expanding our inventory generation methods. We will develop rogue server detection techniques for DNS, VPN, and web services. We will also be expanding the inventories to include other client/server models, such as FTP.

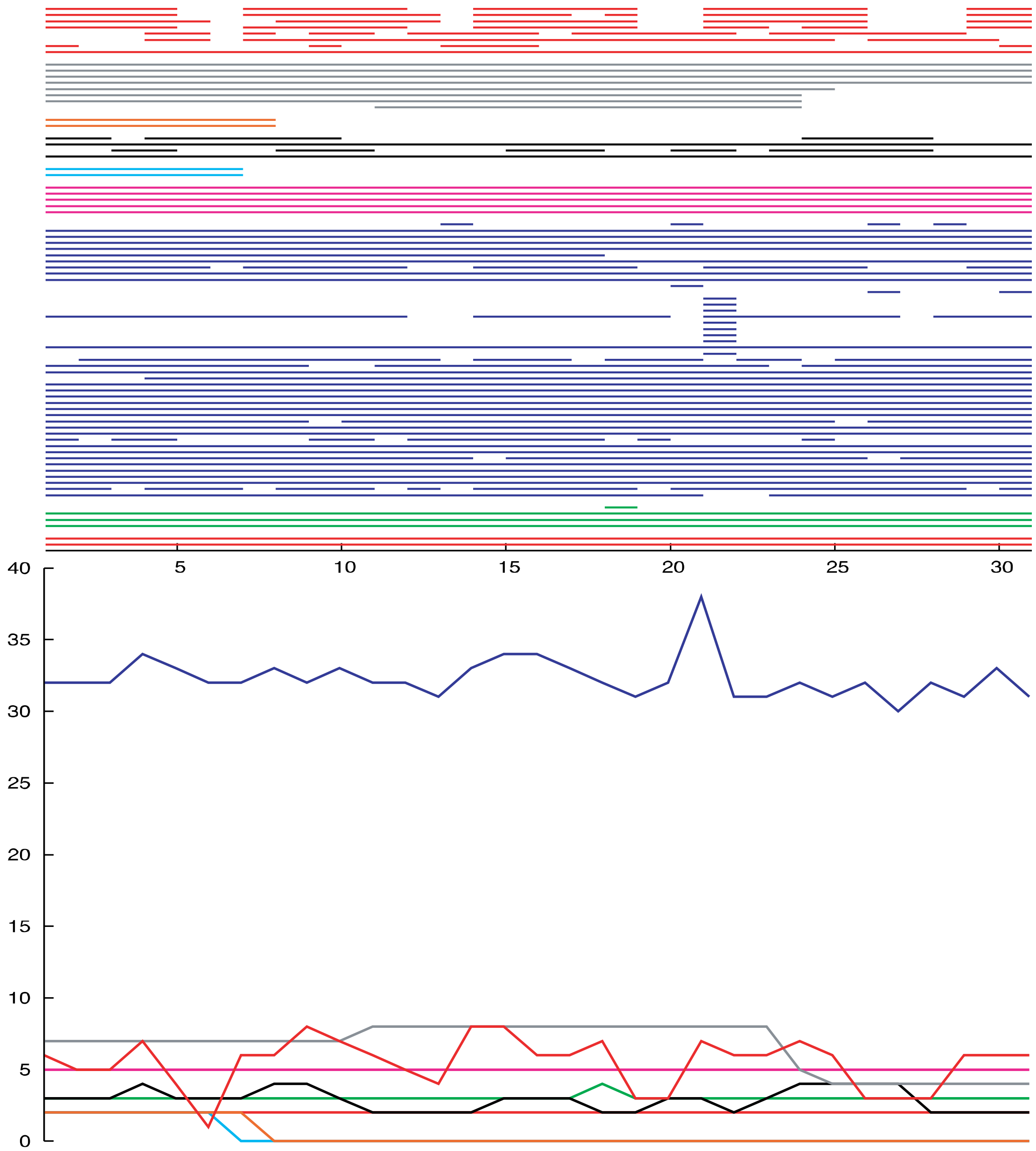
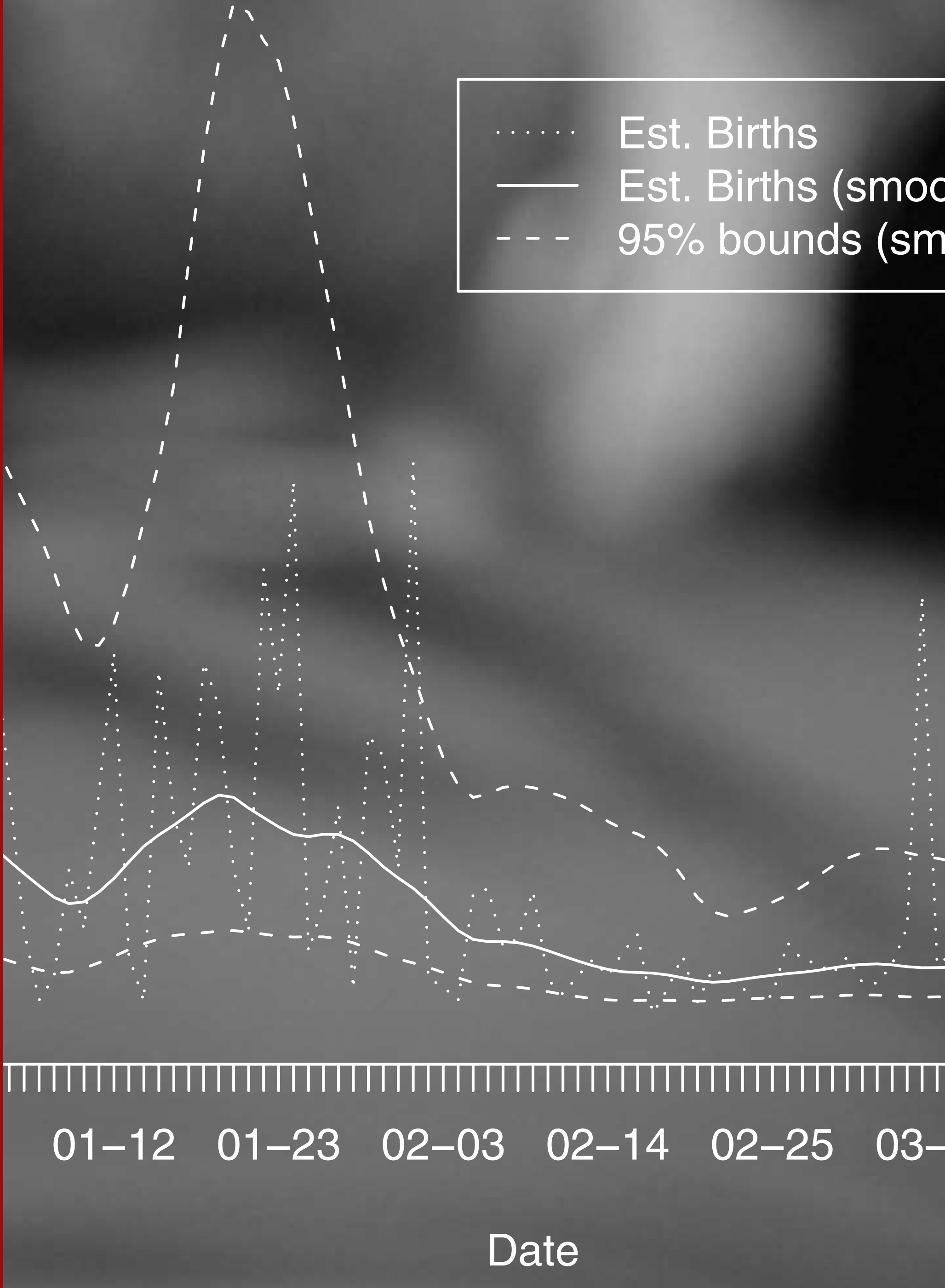


Figure 1: Observed Behavior of Known SMTP Servers Located on Nine Independent Networks

Estimating Phishing Populations from Reports

Rhiannon Weaver
412-268-8511

M. Patrick Collins
412-268-9124



Estimating Phishing Populations from Reports

Problem Addressed

We estimate the extent of phishing activity (fraudulent websites designed to capture personal information) on the Internet by collating information from multiple phishing reports. We see this population estimation work as a necessary component for understanding the extent of the threat posed by phishing and other hostile activities to the public health of the Internet. In particular, the CERT Network Situational Awareness (NetSA) group studies phishing as part of a broader examination of *uncleanliness*, which is the likelihood of a particular IP address being used to launch attacks. Since uncleanliness values are calculated by collating reports of hostile activity, we use population estimates to determine how complete our coverage of a particular threat is.

Research Approach

In a phishing attack, a perpetrator attempts to gain access to a victim's financial or other private information by impersonating a website that the victim trusts, such as a bank, or an eBay or PayPal site. By using a spam email, the perpetrator prompts the victim to visit the site and enter their legitimate username and password. Thus, the act of phishing requires creating and *publicly* advertising a fraudulent site. This necessity gives us an opportunity to collect publicly available information on phishing sites and to use this information to estimate the overall phishing population with a statistical method called *capture-recapture estimation*.

Online watchdog sites and commercial providers routinely collect information on advertised phishing sites. This information is reported as a set of records, where each record contains the following fields:

- **report date:** the date on which the phish was reported to the source
- **target**: the institution that the phish impersonated
- **url**: the URL that victims were instructed to click on in the phishing email
- **address**: the IP address of the site hosting the phish (i.e., the IP address of the web server that hosts the URL)

Capture-recapture estimation uses records from multiple sources to estimate population sizes. Essentially, each of the watchdog sites and commercial providers are independently "trolling a pond," looking for phishes and reporting records as they come across them. The amount of *overlap*, that is, the number of unique records that are reported by more than one source, gives us an idea of how large the population is and how much of the population is being reported. A large overlap between sources indicates good coverage of the population, so that most of the phish in the pond are accounted for, while a small overlap indicates that many phish in the pond are being overlooked.

Capture-recapture estimation has been used effectively in many different applications, although in practice there are complexities about both the population and the sources that must be accounted for. This is the case for the phishing population study, where our population estimates must be robust enough to handle the following:

- **Heterogeneity**: differences among phishers or reporting methods that affect the probability that a record will be reported (i.e., "captured"). This may include the sophistication or subtlety of a phishing site or the different reporting and classification methods used by distinct sources.
- **Locality**: the tendency of individuals to cluster within the population and of sources to troll only parts of the whole pond. In phishing, an example of such a tendency would be for multiple phishers to use the same host address.
- **Open populations**: *births* and *deaths*, in this context meaning a change in the true population occurring during the time the estimation takes place. In phishing, such phenomena would include the introduction or take-down of phishing sites.
- **Trap effects**: effects from reporting that affect the probability of subsequent reports for a particular target. For example, phishers may demonstrate less interest in a host once it has been identified by a source.

Estimation of phishing populations is further complicated by the ambiguous relationship between URLs, the machines hosting those URLs, and the phishers controlling those machines. In phishing circles, "bullet-proof hosting" sites will sometimes employ a technique, called "fast flux," that rapidly cycles DNS services through a cache of IP addresses to avoid blacklisting. On the other hand, sophisticated phishers, such as the "Rock Phish" group working out of Russia, will often register many different domains to create hundreds of URLs that point to the same server, housing several sites impersonating different targets.

To account for these complexities, we take two general approaches:

1. Before applying capture-recapture techniques, we build a set of *scams* out of the observed records. This is to control locality in the population and to attempt to account for fast flux or Rock-Phish-like relationships as described above. A scam is a construct created from a cluster of multiple, similar records. For example, a large set of similar URLs with the same target, reported at the same time, would be clustered as a single scam.
2. We *stratify* the population by dividing scams into subgroups, based on descriptive characteristics and time, to reduce the effects of heterogeneity of the population, open populations, and trap effects on the population estimates.

Expected Benefits

Measuring and reporting the population of phishers can help provide indicators of malicious activity on the Internet as a whole. There is also benefit in understanding what percentage of a population goes unseen by public watchlist sites and measuring how populations grow and change with time. Once fully developed for phishing, the method can also be generalized to study other populations of interest, such as spammers, botnets, and malware.

2007 Accomplishments

In 2007, we performed an exploratory capture-recapture study using two sources. The source identified as "netcraft" in the study was collected by NetCraft LLC.[1] The source identified as "castlecops" was prepared from data collected from the Phishing Incident Reporting and Termination Squad.[2] We compiled records from these two sources over the period of January 1, 2007 through March 21, 2007.

We implemented a very simple clustering technique for grouping records into scams based on equal targets, proximity in IP space, and similarity in URL in terms of the levenshtein distance between the two strings.[3] We matched these scams across the two lists, resulting in 18,593 total scams, 506 of which were reported on both lists. We used the results to study both the population of /24 netblocks housing scams and the population of scams themselves.

Phishing by Netblock

We used the number of scams observed per /24 netblock, as well as the average lifetime reported for scams on each block, to characterize four strata of activity for netblocks. *Isolated* netblocks were observed to have a small number of short-lived scams over the 12-week study. *Persistent* netblocks had a small number of scams reported, but the scams had a long duration. *Bursty* netblocks had a large number of relatively short-lived scams, while *Corrupted* netblocks exhibited a large number of long-lived scams. We also broke the Isolated and Persistent strata into two sub-strata by time, due to observation of a distinct drop in activity among those strata from February 7th onward. Figure 1 shows an example of observed scam activity for each of these strata.

We hypothesized that persistently compromised netblocks, such as those in the Corrupted spectrum, may make up a small percentage of the overall population; but due to their activity level, they will be more visible to source reports. Conversely, netblocks that are compromised for short periods may be more difficult to find. Our results, summarized in Table 1, seem to support this claim. Capture rates indicate that the two lists capture about 80% of the population of /24 netblocks, with rates going down in the Isolated and Persistent strata in the latter part of the study.

Phishing by Scam

We explore scam populations by both date and type. Figure 2 shows capture-recapture estimates of the number of scam "births" per day, over the 12 weeks in the study. Because of the naïve clustering method and the small overlap ratio, the estimates fluctuate greatly from day to day, with high variability. A smoothed line, with smoothed 95% confidence bounds, is shown in the figure, and was used to obtain an estimate of a population of approximately 88,000 scams, indicating a capture rate of only 21%. Capture rates were worse in January when activity was high and stabilized to about 31% in February and later weeks.

Table 2 breaks down capture rates for scams by type. We use the scam size (measured by the number of records clustered together to create the scam) and the scam target type to define several strata, again split according to the February 7th divide. *Parts* indicates that while the scam comprised few records, records for the target type typically show up in related bursts. *Isolated* indicates small scams with unusual targets (the target was not a bank, nor was it eBay or PayPal). *Kits* indicate large scams of all types. We report population estimates as percentages of the 88,000 scams indicated by the longitudinal study.

A striking feature of Table 2 is that bank kits, while comprising approximately 43% of all records in the population, account for only 3.75% of scams. This indicates the level to which the sophisticated multi-homed or multi-domain phishing scams color the view that we have into the population based on counting individual records alone.

2008 Plans

In the upcoming year, we plan to refine our clustering techniques and to compile records from more available sources, facilitated by our relationship with the Anti-Phishing Working Group (APWG). While we presented an exploratory study in this report based on simple capture-recapture estimation within strata, in the future we plan to implement a fully Bayesian statistical approach in order to refine our modeling techniques. We plan to operationalize reporting of our population estimates as a web feed.

[1] http://www.netcraft.com

[2] As described by PIRT, "PIRT is a global phishing termination operation launched by CastleCops and Sunbelt Software. PIRT is operated at www.castlecops.com, a volunteer security community focused on making the Internet a safer place."

[3] Levenshtein distance counts the number of insertions or deletions of characters needed to transform one string into the other.

Reference

Weaver, R. & Collins, M. P. "Fishing for Phishes: Applying Capture-Recapture Methods to Estimate Phishing Populations." Presented at the APWG 2007 eCrime Researchers Summit, October 4–5, 2007, Pittsburgh, PA. http://www.cert.org/netsa/publications /ecrimes07-collins-weaver-fish-for-phish.pdf

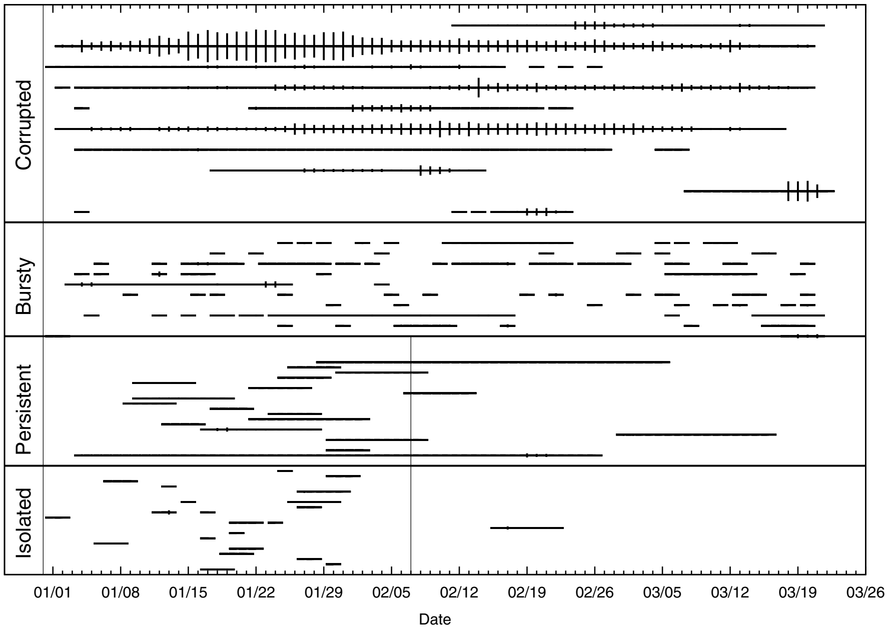


Figure 1: Examples of Scam Activity per Representative /24 Netblock for Six Strata

The length indicates the amount of time the host was observed in phishing activity, with the height of the bar indicating the number of active scams observed.

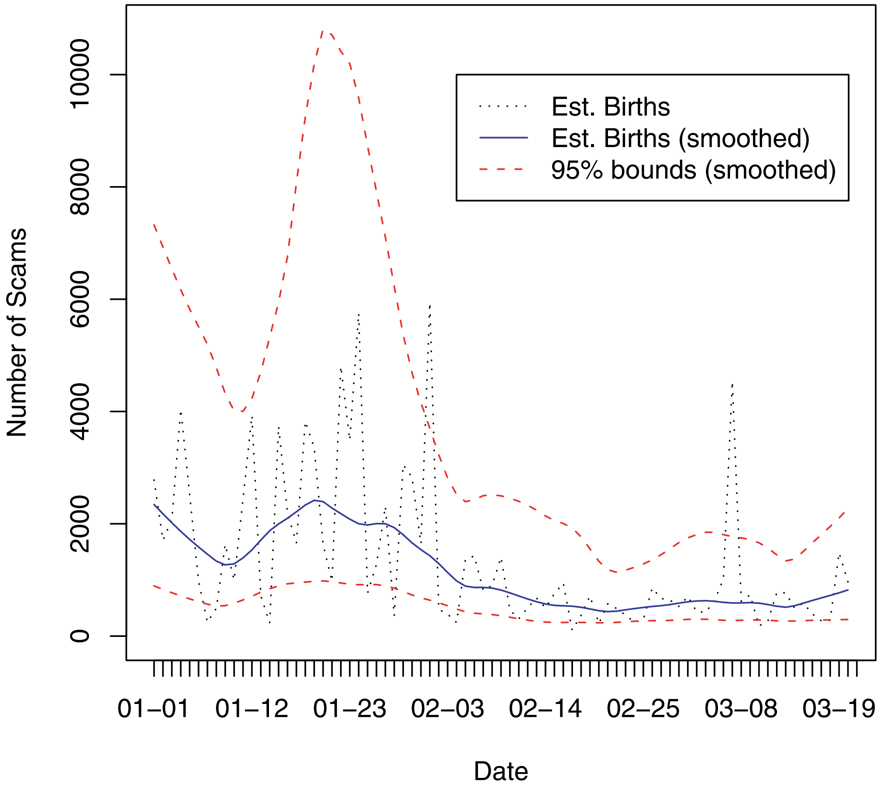


Figure 2: Estimate of the Number of Scam Births per Day over the 12 Weeks in the Study

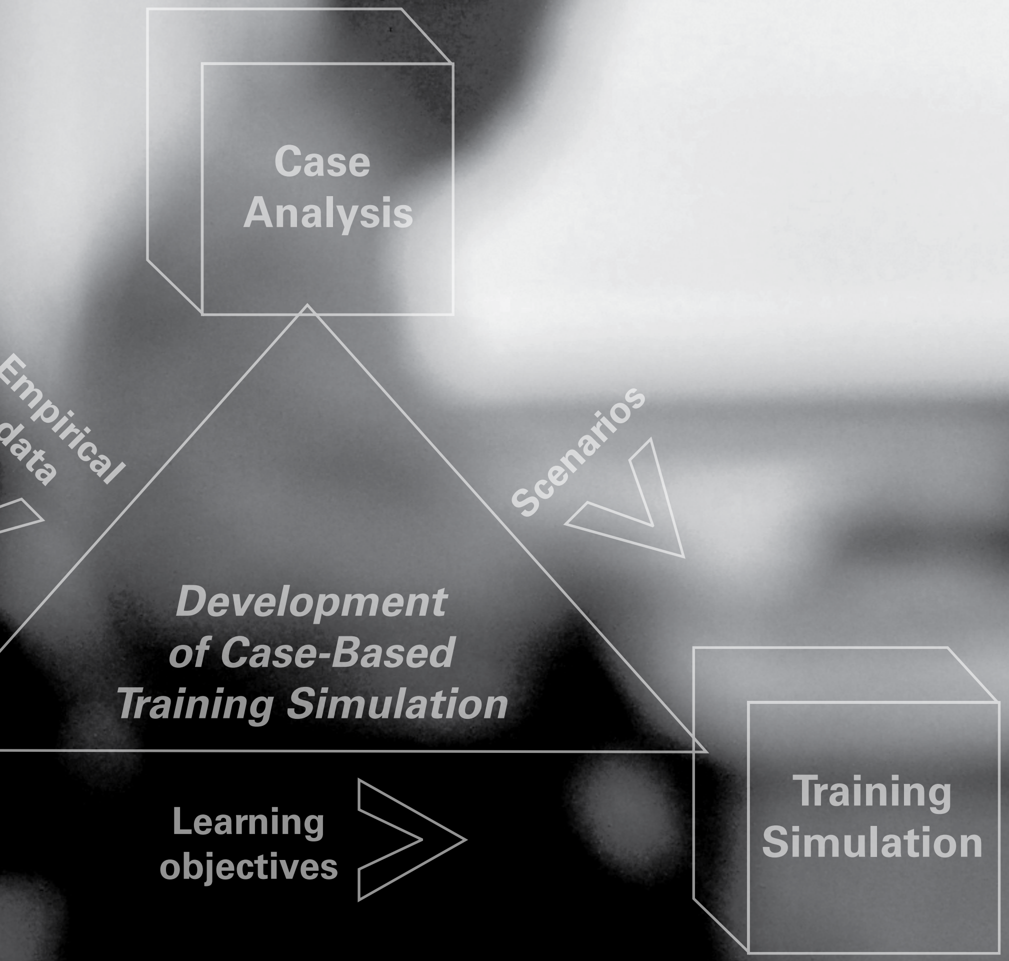
	Characteristics			Capture Profile			Population Estimate		
	Date Range	Scams/Block	Avg. Lifetime	netcraft	castlecops	Both	Population	Rate	95%CI
Isolated-1	01/01-02/06	< 10	≥ 3 days	1,410	25	24	1,469	0.96	(0.71,0.99)
Isolated-2	02/07-03/21	< 10	≥ 3 days	279	48	20	670	0.45	(0.27,0.66)
Persistent-1	01/01-02/06	< 10	> 3 days	2,669	227	207	2,927	0.92	(0.85,0.95)
Persistent-2	02/07-03/21	< 10	> 3 days	726	210	111	1,374	0.60	(0.50,0.70)
Bursty	01/01-03/21	≥ 10	≥ 3 days	117	43	43	117	1.00	(0.89,1.00)
Corrupt	01/01-03/21	≥ 10	> 3 days	116	93	88	123	0.98	(0.94,1.00)
Total				5,317	646	493	6,680	0.82	(0.62,0.91)

Table 1: Population Estimates of /24 Netblocks

	Characteristics			Capture Profile					Population Estimate			
	Date	Recs/Scam	Type	Recs	Scams	netcraft	castlecops	Both	%Scams	≈%Recs	Rate	95%CI
Parts	01/01-02/06	< 5	bank	5,649	4,255	3,766	565	76	35.66	16.56	0.15	(0.10,0.22)
		< 5	ebay	3,365	2,629	2,592	58	21	9.12	4.08	0.37	(0.18,0.60)
		< 5	paypal	2,481	1,986	1,892	135	41	7.94	3.47	0.32	(0.19,0.48)
Isolated		< 5	other	1,815	1,427	1,307	148	28	8.80	3.91	0.21	(0.11,0.35)
Kits		≥ 5	bank	18,447	552	165	410	23	3.75	43.83	0.19	(0.09,0.34)
		≥ 5	ebay	2,417	71	69	10	8	0.11	1.31	0.82	(0.30,1.00)
		≥ 5	paypal	1,765	76	59	39	22	0.13	1.06	0.72	(0.42,0.95)
		≥ 5	other	346	47	26	32	11	0.10	0.26	0.62	(0.26,0.94)
Parts	02/07-03/21	< 5	bank	4,195	3,250	2,949	369	68	20.38	9.20	0.20	(0.14,0.29)
		< 5	ebay	1,121	879	827	82	30	2.88	1.28	0.39	(0.22,0.59)
		< 5	paypal	1,661	1,251	1,083	225	57	5.45	2.53	0.29	(0.19,0.42)
Isolated		< 5	other	1,521	1,181	1,055	185	59	4.21	1.90	0.36	(0.24,0.50)
Kits		≥ 5	bank	6,230	269	124	166	21	1.25	10.13	0.27	(0.14,0.48)
		≥ 5	ebay	134	19	11	12	4	0.04	0.10	0.56	(0.13,1.00)
		≥ 5	paypal	506	77	36	69	28	0.11	0.25	0.87	(0.59,1.00)
		≥ 5	other	210	35	16	28	9	0.06	0.13	0.70	(0.27,1.00)

Table 2: Population Estimates of Scams by Type

Insider Threat Modeling and Analysis



Dawn M. Cappelli
412-268-9136

Andrew P. Moore
412-268-5465

Randall F. Trzeciak
412-268-7040

Insider Threat Modeling and Analysis

Problem Addressed

Insiders include current and former employees and contractors who have or had authorized access to their organization's systems, data, and networks. Insiders are familiar with internal policies, procedures, and technology and can exploit that knowledge to facilitate attacks and even collude with external attackers. Consequences of malicious insider incidents include financial losses, operational impacts, damage to reputation, and harm to individuals. The actions of a single insider have caused damage to organizations ranging from a few lost staff hours to negative publicity and financial damage so extensive that businesses have been forced to lay off employees and even go out of business. Furthermore, insider incidents can have repercussions extending beyond the affected organization, disrupting operations or services critical to a specific sector, or resulting in issuance of fraudulent identities that create serious risks to public safety and national security.

CERT's ongoing insider threat research[1] provides comprehensive analysis of the insider threat problem, including espionage, insider IT sabotage, theft of confidential or sensitive information, and fraud. CERT's technical security expertise has been augmented by experts in the areas of psychology, sociology, insider threat, espionage, cyber crime, and specific domains such as the financial industry. Our research has shown that to detect insider threats as early as possible or to prevent them altogether, members of management, IT, human resources, security officers, and others in the organization must understand the psychological, organizational, and technical aspects of the problem, as well as how to coordinate their actions over time. As of December 2007, CERT's insider threat team has collected and coded over 250 insider threat cases in a series of databases. These databases contain information regarding insider motivation, planning, technical preparatory actions, technical details of the incident, detection, and more. CERT staff have gained additional contextual information via interviews with convicted insiders, victim organizations, investigators, and prosecutors. CERT's research includes both classified and unclassified projects; this article focuses only on the unclassified research.

In January 2002, the CERT Program and the United States Secret Service (USSS) National Threat Assessment Center (NTAC) started a joint study combining NTAC's expertise in behavioral psychology with CERT's technical security expertise to provide in-depth analysis of approximately 150 insider incidents that occurred in critical infrastructure sectors in the U.S. between 1996 and 2002.[2] Four reports have been published to date as part of the study: one analyzed malicious insider incidents in the banking and finance sector, one analyzed insider IT sabotage attacks across all critical infrastructure sectors, one pertained to the information technology and telecommunications sector, and one focused on the government sector. In addition, in 2007 CERT collected and coded approximately 100 additional cases of insider compromise that occurred since 2002, updating the previous work with NTAC.[3]

CERT's insider threat modeling, referred to as MERIT (Management and Education of the Risk of Insider Threat),[4] uses the wealth of empirical data collected by CERT to convey the "big picture" of the insider threat problem—the complexity of the problem, relative degree of risk, and unintended consequences of policies, practices, technology, insider psychological issues, and organizational culture over time. As part of MERIT, we are developing a series of models and associated tools that can be used to communicate the risks of insider threat and help organizations understand how to mitigate those risks [1,3,5]. In addition, we have developed a workshop to raise awareness of management, security staff, IT departments, and human resources personnel about behavioral and technical indicators and ways to decrease risks.

Research Approach

Our understanding of the insider threat problem has evolved from our initial, detailed data collection and analysis of insider cases to a more comprehensive view of the problem through a series of group modeling sessions. Our data collection process has been guided by the development of codebooks that detail behavioral, organizational, and technical aspects of the cases. All data is stored in databases, facilitating ongoing, evolving analysis for multiple purposes. For instance, we have published reports detailing motive, planning behaviors, technical actions, and detection of insider threats. We also could use the database to evaluate functionality in currently available tools versus actual cases, for example.

Our approach to group modeling is based on the system dynamics methodology. System dynamics is a method for modeling and analyzing the holistic behavior of complex problems as they evolve over time. Group model building involves bringing together individuals with a range of domain expertise to build dynamic theory based on case study research. Over the years, we have brought together

[1] CERT's public work in the insider threat area is comprehensively documented on our web page at http://www.cert.org/insider_threat/.

[2] This study was funded by the USSS, as well as the Department of Homeland Security, Office of Science and Technology, which provided financial support for the study in fiscal years 2003 and 2004.

[3] This work was funded by Carnegie Mellon University's CyLab.

[4] MERIT is funded by Carnegie Mellon University's CyLab. Our modeling and comparison of insider IT sabotage and espionage was funded by the DoD Personnel Security Research Center (PERSEREC).

behavioral scientists, psychologists, and historians to complement CERT's technical understanding of the problem to build system dynamics models that capture the nature of the insider threat problem and how it evolves over time [3]. Our initial focus of the group modeling effort was in the area of insider IT sabotage—an insider's use of information technology to direct specific harm at an organization or an individual [4], as well as cases of espionage against the U.S. [1].

The system dynamics approach helped to structure and focus the team's discussion. This was particularly important since members of the team, by necessity, came from the different disciplines of psychology and information security. By identifying the primary variables of interest, the influences between these variables, and the feedback loops that are so important for understanding complex behavior, the team found itself able to communicate much more effectively. The group modeling process enabled the team to step back and consider the "big picture" at times and focus on individual concepts at other times. The models also provided a concrete target for validation through mapping to observables exhibited by the real-world cases. The linkage to observables relates behaviors recognized as important for early detection to actions managers can take to better identify and understand an evolving insider threat. This approach helps ensure that recommendations made as a result of the modeling effort are actionable.

Expected Benefits

The ultimate effect of business policy and technical decisions on insider threat risks is complex and often counterintuitive, and can result in significant losses and operational impacts due to insider cyber attack. This work identifies and validates policies, practices, and technologies—helping decision-makers better understand insider threat risks and the effects of decisions on the promotion or mitigation of that risk. The results of our work will empower organizations to develop comprehensive, efficient, and justifiable defenses against insider threats along with the organizational understanding and support needed to maintain a strong security posture over time. Broad application of concepts developed will enable organizations across the U.S. and abroad to significantly reduce their risk and losses due to insider attacks. The capability that will result from this work promotes the security, survivability, and resiliency of all government, military, and commercial critical systems. The ultimate beneficiaries will be organizational stakeholders and, where the U.S. critical infrastructures are better protected, the general public as a whole.

2007 Accomplishments

In 2006 CERT conducted a series of insider threat education and awareness workshop pilots involving managers representing information technology, human resources, legal council, risk management, and physical security across a wide range of organizations [5]. Based upon feedback from those pilots, we re-organized the workshop in 2007 so that the most critical domain concepts were presented prior to introducing the system dynamics model to the participants. We simplified the notation used to present the MERIT model and focus on the feedback relationships between model variables. We developed and presented the resulting insider threat education and awareness workshop with the following structure:

- overview of empirical research on insider threat
- interactive discussion of the instructional case of insider threat
- general observations from the case data
- system dynamics model: problem, prevention, and mitigation
- recommendations for countering the threat

While the system dynamics model of insider IT sabotage is useful within the workshop context to convey dynamic interrelationships between important domain concepts, we have found that it is most effective to introduce primitive concepts that may be unfamiliar to workshop participants gradually before introducing the model, e.g.,

- *personal predisposition* - a personal characteristic historically linked to a propensity to exhibit malicious insider behavior, and
- *access path* - a sequence of one or more access points that lead to a critical system.

When the participants are comfortable with such concepts, we then introduce dynamic interrelationships, such as when an insider's personal predispositions leads to an escalation of disgruntlement, a rash of observable behavioral precursors, followed by the creation and use of access paths unknown to the organization to execute an attack. This approach helps ensure that workshop participants are not overwhelmed with too many new concepts, both modeling and domain, at the same time.

Rather than simply presenting concepts as a series of definitions and examples, the workshop engages participants in interactive discussion based on instructional cases. Concrete case examples, with well-posed questions, allow participants to bring in their own experiences and insights, making the learning experience informative and engaging. Since the sensitivity of insider threat case data precludes the use of actual cases for training, we developed a fictional case scenario that is representative of a preponderance of the actual cases.

We also developed an instructional case that illustrates insider threats during the software development life cycle. We believe that the use of instructional cases provides a coherent and well-grounded basis for training on the important issues relevant to insider threat. The case scenarios are representative in character (but not necessarily detail) of many of the actual cases that we have seen.

The MERIT workshop is only a first step toward producing more effective training on insider threat risk awareness and mitigation. As shown in Figure 1, CERT is also attempting to bring the benefits of training simulation, or serious games, to bear on the challenge of insider threat education. In collaboration with Carnegie Mellon's Entertainment Technology Center, we built a proof-of-concept training simulation, called MERIT-*Interactive,* that immerses players in a realistic business setting from which they (1) make decisions regarding how to prevent, detect, and respond to insider actions and (2) see the impacts of their decisions in terms of key performance metrics.[5] MERIT-*Interactive* provides a team-oriented, role-playing experience using model-based simulation of critical aspects of insider threat risk management in a realistic organizational context. Team orientation is critical because organizations typically do not identify these problems at the individual manager level, nor are solutions generally implemented solely in one department, because the problem is enterprise wide. Role-playing is also crucial because solutions generally require collaboration among multiple stakeholders.

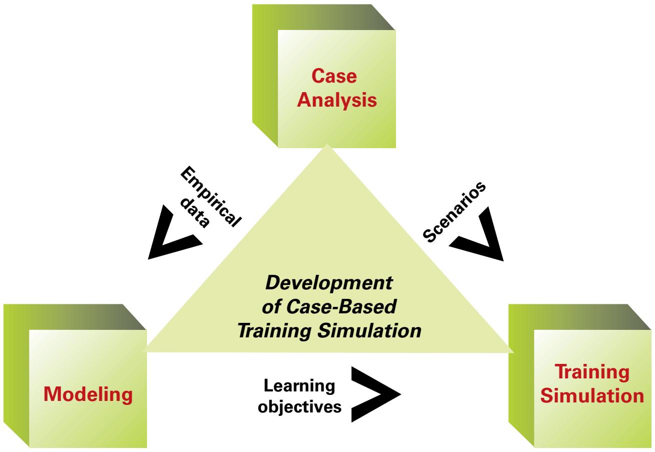


Figure 1: Using Simulation in Insider Threat Education

In the future, we plan to revise our previously developed MERIT system dynamics model to serve as a back-end engine for the game. This approach should facilitate transferring of insights from the model into lessons taught by MERIT-*Interactive*. Experiments will be needed to assess the extent to which players are learning the important lessons regarding the insider threat domain. We believe that MERIT-*Interactive* will ultimately help decision-makers and stakeholders in organizations better understand insider threat risks and the effects of decisions on the promotion or mitigation of that risk.

2008 Plans

Ongoing and future research into insider threat at CERT includes two areas: a broader study of specific types of insider threats, and the development of an insider threat risk assessment diagnostic.[6] This section summarizes both of these areas of work.

Broader Study of Insider Threats

The library of assets produced by the MERIT project provides a collection of tools that have been very effective in transitioning our knowledge of insider IT sabotage to an international audience of security experts, IT practitioners, all levels of government and business managers, and law enforcement. Insider threat workshop participants appreciate the interactive nature of the initial discussions and the use of the model to interrelate important, but complex, insider IT sabotage domain concepts. They have one primary suggestion for improvement: they need to understand insider fraud and theft of confidential or sensitive information in the same depth that the workshop provides for insider IT sabotage. Our empirical data collection shows that insider theft of information using IT, including crimes such as identity theft and corporate espionage, is a significant problem in today's privacy-conscious and competitive corporate world. Likewise, insider fraud using IT is a significant problem in industry, especially in the banking and finance sector. Case data collected on these two types of crimes bolsters the need for modeling and analysis, as there are significant differences in those crimes as compared to insider IT sabotage, especially in motivation, insider characteristics, and the technical nature of the malicious activity [2].

[5] MERIT-*Interactive* was funded by Carnegie Mellon University's CyLab.

[6] Both 2008 projects are funded by Carnegie Mellon University's CyLab.

The primary objective of our broader study is to extend MERIT to include a comprehensive pattern analysis and transition mechanism for *all* types of insider threat, including fraud, theft of confidential or sensitive information, and IT sabotage. Outputs of this project will include a complete package of empirically based insider threat system dynamics models, as well as a comprehensive insider threat workshop that includes in-depth analysis and interactive discussion of all three types of insider crimes. We expect that participation in the workshop will empower corporate and government personnel to develop comprehensive, efficient, and justifiable defenses to insider threats along with the organizational understanding and support needed to maintain a strong security posture over time.

In addition to analyzing previously collected insider fraud and theft of information cases, for the past year we have been collecting and analyzing insider compromises that have occurred since 2002. A focus of this broader analysis will be to determine how the insider threat is evolving as well as to generate a larger dataset on which to base findings. We plan to update our *Common Sense Guide to Prevention and Detection of Insider Threats* based on the results of this work [2].

Insider Threat Risk Diagnostic Instrument

The objective of this project is to build a comprehensive diagnostic instrument that is empirically based and that can be used by organizations to assess their insider threat risk, with the ultimate goal of improving the resiliency and survivability of the organization. The insider threat risk assessment diagnostic will enable organizations to gain a better understanding and manage the risk of insider threat. It will merge technical, organizational, personnel, and business security and process issues into a single, actionable framework. As in our past projects, the project team includes psychological and technical expertise. The instrument will be structured to encompass all stakeholders in the fight against insider threat: management, information technology, human resources, and physical security.

We will build a pilot instrument based on approximately 250 insider threat cases in CERT's case library and will continue to expand our library with recent cases for inclusion in this research. We are seeking opportunities for collaboration with external organizations on this project. Collaboration opportunities range from reviewing the instrument to confidential sharing of insider case and/or best practice information for inclusion in the instrument. In return for participation, we will offer those organizations opportunities to pilot the insider threat risk assessment diagnostic. Following each pilot, we will provide the organization with a confidential report on the findings of the pilot, and suggestions for improvement. As with all of our insider threat research, all collaborations will remain confidential and no references will ever be made to any organizations and/or individuals.

Bibliography

[1] Band, S. R., Cappelli, D. M., Fischer, L. F., Moore, A. P., Shaw, E. D., & Trzeciak, R. F. *Comparing Insider IT Sabotage and Espionage: A Model-Based Analysis* (CMU/SEI-2006-TR-026). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2006. http://www.cert.org/archive/pdf/06tr026.pdf

[2] Cappelli, D. M., Moore, A. P., Shimeall, T. J., & Trzeciak, R. F. *Common Sense Guide to Prevention and Detection of Insider Threats: Version 2.1*. Report of Carnegie Mellon University, CyLab, and the Internet Security Alliance, July 2006 (update of the April 2005 Version 1.0). http://www.cert.org/archive/pdf/CommonSenseInsiderThreatsV2.1-1-070118.pdf

[3] Cappelli, D. M., Desai, A. G., Moore, A. P., Shimeall, T. J., Weaver, E. A., & Willke, B. J. "Management and Education of the Risk of Insider Threat (MERIT): Mitigating the Risk of Sabotage to Employers' Information, Systems, or Networks." *Proceedings of the 24th International Conference of the System Dynamics Society.* Nijmegen, Netherlands, July 2006. http://www.albany.edu/cpr/sds/conf2006/proceed/proceed.pdf

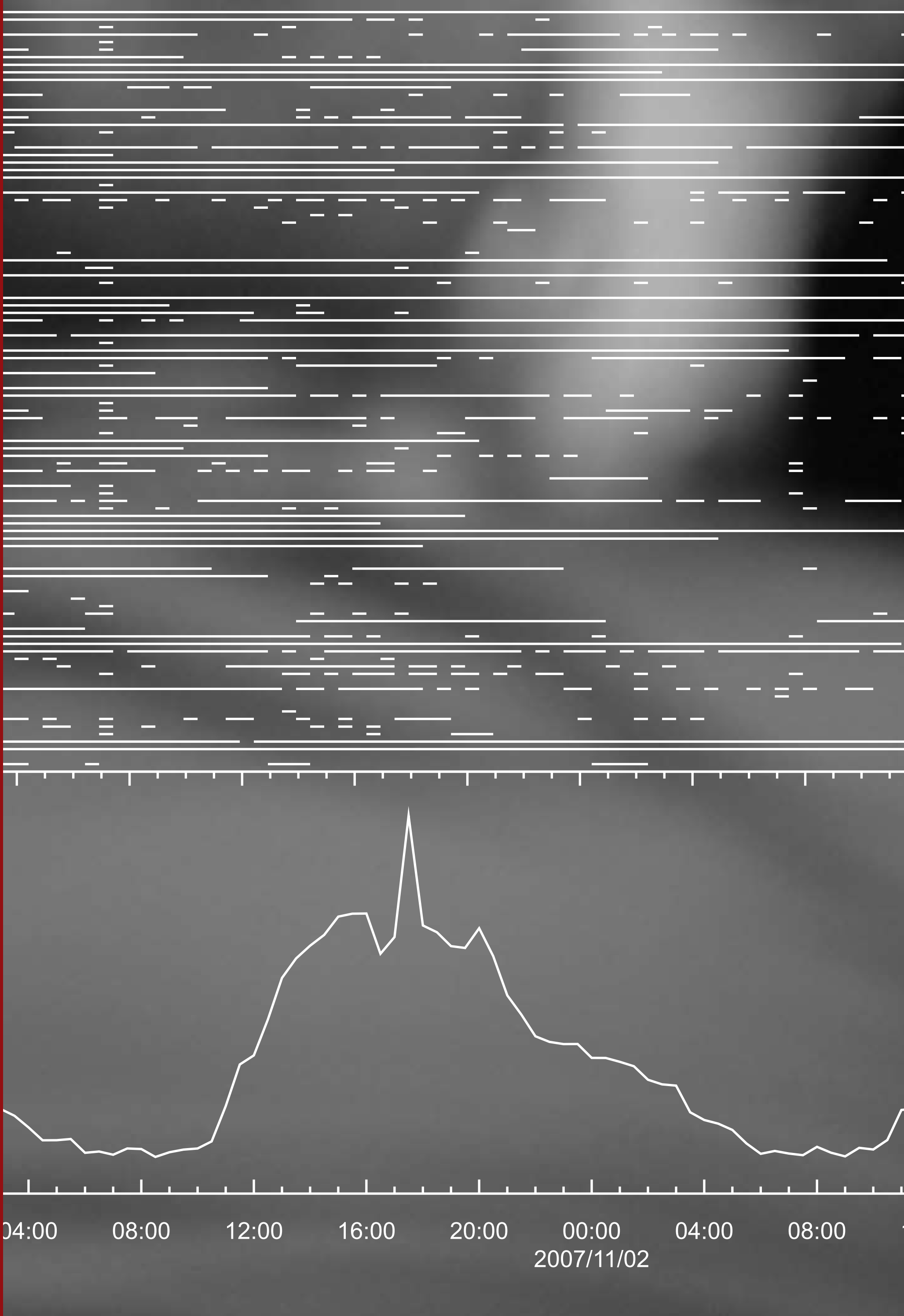
[4] Keeney, M. M., Kowalski, E. F., Cappelli, D. M., Moore, A. P., Shimeall, T. J., & Rogers, S. N. *Insider Threat Study: Computer System Sabotage in Critical Infrastructure Sectors*. Joint SEI and U.S. Secret Service Report, May 2005. http://www.cert.org/archive/pdf/insidercross051105.pdf

[5] Moore, A. P., Cappelli, D. M., Joseph, H., Shaw, E. D., & Trzeciak, R. F. "An Experience Using System Dynamics Modeling to Facilitate an Insider Threat Workshop." *Proceedings of the 25th International Conference of the System Dynamics Society.* July 2007. http://www.systemdynamics.org/conferences/2007/proceed/papers/MOORE349.pdf

Jeff Janies
412-268-8225

John Prevost
412-268-9201

Network Traffic Visualizations for Portal Environments



Network Traffic Visualizations for Portal Environments

Problem Addressed

An integral part of network situational awareness is representing a network's phenomena, trends, and state in a concise and informative manner. Data visualization is a key tool in bridging the gap between data collection and analysis. Visualization allows analysts to easily infer patterns of activity, to isolate and test specific variables in network traffic, and to concisely convey their findings to others.

Three of our group's areas of interest are providing global views of networks' connections, representing data collection sensors' uptime (*sensor health*), and representing the activities of hosts of interest. Developing visualizations for these areas of interest greatly improves situational awareness capabilities. For instance, by understanding how a subnet interacts with the Internet, we gain insight into naturally occurring trends. Understanding a sensor's current state ensures a measure of data integrity. Finally, visually expressing hosts' activities in an unambiguous way allows for ease of comparing behavior and quickly gaining insight about a host (or collection of hosts).

Research Approach

We present here three visualizations that individually fulfill the needs of the areas of interest described above. In order to represent the profile of Internet hosts communicating with a given subnet, we make use of a Hilbert curve. Individual sensor behaviors are presented using a dense graphic containing many useful pieces of information. Finally, we use a visualization that we call an *existence plot* to represent the activity of a finite number of hosts in a low-resolution display.

Hilbert Curves

Displaying summaries of Internet host behavior is more challenging than displaying summaries of host behavior in an internal network. In general, networks of individual organizations tend to be small and to exhibit very simple network hierarchies. It is often sufficient to map IP space directly onto a single axis of a graph. Hierarchies only split the internal IP space into a small number of distinct sections, each of which may be represented explicitly on the axis. The Internet, however, is extremely large and contains multilevel hierarchies of varying size. As a result, simple techniques for mapping IP addresses onto the *x*- or *y*-axis of a plot fail, since they cannot adequately account for the resolution required both in volume and hierarchical structure.

Mapping from linear to planar position on a Hilbert space-filling curve provides a number of features that are helpful for our task, as shown by Munroe [4] and Irwin et al. [1]. First, this mapping preserves locality. If two addresses are adjacent in IP space, they are adjacent on the plane after being mapped onto the curve. More than that, values near each other linearly are clustered on the plane based on a two-bit prefix. In practice, this means that any CIDR block you care to choose is represented by either a square or a rectangle after being mapped onto the Hilbert curve. In addition to spreading addresses out enough that small networks are distinguishable, this mapping keeps Internet address hierarchies of any size organized. Because of this, visual examination of Internet traffic mapped onto a Hilbert curve can identify features that would be invisible on a more traditional planar mapping.

Our current renderings using a Hilbert curve work by grouping hosts into the desired bin size, and then mapping the individual bins onto the Hilbert curve and using a heatmap to represent information on the volume of traffic from that bin (see Figure 1). In our experience, it is practical to use this approach for bin sizes between 8-bit and 24-bit prefixes of IP addresses. On the low end of that scale, the resolution is very low (an 8-by-8 grid) and very little detail can be made out. On the high end of that scale (a 4096-by-4096 grid) the representation begins to edge into uselessness because the individual networks communicating are simply too sparse. Even with the prevalence of network scanning activity, /24s that talk to a given network are much rarer than /24s that do not.

Our best results have come with a grid size of either 512-by-512 or 1024-by-1024, representing either 18-bit or 20-bit prefixes of IP addresses. At these sizes, bins contain enough hosts that it is reasonable to compare behavior of different portions of the Internet. In addition, because it is possible to resolve detail without the use of a magnifying lens, animating these images is an option. Such animations reveal further interesting details in Internet behavior over time.

Sensor Health

Succinctly representing a sensor's view of traffic greatly aids configuration of sensors and understanding of sensors' current states. For this task, we are less interested in precisely representing all readings and more interested in presenting the relative magnitude of sensor readings. In addition, we would like to report significant drops in data collection. Figure 2 presents a visualization that we are currently implementing to address these two concerns. This visualization provides over 1,000 data points, which includes information about the most common ports utilized, level of activity, and percentage of uptime. The visualization has three components: an overall health indicator, activity level overview, and traffic type overview.

The health indicator is a colored box with a numeric percentage that represents the total uptime of the sensor. It is prominently displayed in the center of the visualization. We calculate the percentage of uptime as the number of hourly files generated by the sensor compared to the total number of files that should be generated under normal circumstances. We assume that a sensor will observe at least one flow per hour per type. In the sensor configuration presented, there are four types: *inweb* (all inbound traffic to/from web ports), *in* (all inbound, non-web traffic), *outweb* (all outbound traffic to/from web ports), and *out* (all outbound, non-web traffic). If all hourly files are present, we label the sensor as being 100 percent active and color the box green. If the sensor produced no files, we color the box red. Otherwise, we color the box yellow and display the total percentage of files produced.

Below the overall health indicator is the sensor's activity level overview, which consists of four spark lines representing the traffic level observed by the sensor over a one-day period for the traffic types *in*, *inweb*, *out*, and *outweb*. The spark lines are based on hourly readings of each type of traffic. Here we are concerned with a relative measure of traffic and thus remove the scale.

The traffic breakdown consists of four stacked line charts surrounding the health indicator and activity level overview. The stacked line charts represent the proportional utilization of well-known ports as inbound source port, inbound destination port, outbound source port, and outbound destination port. The intent of this display is to capture the relative bandwidth utilization of the most common protocols and ports used by the network. The result is a graph that shows shifts in the distribution of bandwidth over time.

Together, these three components provide an overview of what the sensor sees. Note that this visualization does not provide the analyst with enough information to take action against phenomena, but it does provide information that could aid in investigation and provide a quick understanding of the current state of a sensor. For example, sensor failure is easily detected. Also, the analyst sees large-scale shifts in activity in the traffic breakdowns. Finally, the activity level overview shows drops in specific traffic types.

Existence Plots

The standard (x, y) time series is a common visualization that denotes a single variable's change over time. It is easy to use and universally understood. In a network situational awareness setting, analysts use time series to examine the activity (bytes, packets, and connections) of a single host or collection of strongly related hosts, such as a specific subnet. However, if multiple hosts' activities are aggregated, information is lost. Consider two scenarios for visualizing 100 hosts: plotting each host's activity independently as a time series and aggregating all hosts' activities into one time series. If the analyst plots each host's activity independently, lines will co-mingle, causing the involuntary hiding of information. Furthermore, spikes in a subset of hosts' activities will greatly overshadow hosts with lesser levels of activity. If an analyst aggregates the activities of all 100 hosts over a two-day period and plots the data as a time series, the analyst is unable to distinguish each host's individual contribution to the traffic volume. A spike in a single host's activity is indiscernible from a marginal increase in several hosts' activities.

In order to display the activity of multiple hosts without hiding individual host information, we augment the standard time series with a visualization called an existence plot. An existence plot is a time oriented visualization that can represent a magnitude component for a finite number of variables. In an existence plot, the x-axis represents time and each y-value represents a unique variable (in this case, a unique host). An (x, y) coordinate represents the number of bytes transferred by host y at time x, and is one of four values: none, low, medium or high. None means the host transferred no bytes at time x, and no color is associated with this state. Low, medium, and high are values relative to user defined saddle points S_0 and S_1 and are represented by the colors blue, green, and red, respectively.

Figure 3 is an example of the existence plot augmented time series. Here we represent 100 hosts' bandwidth utilization over a two-day period. The y-values of the existence plot represent each of the hosts. The time series is an aggregation of all 100 hosts' outgoing bandwidth. During the 17th hour of the first day, we observe that a spike occurs in the aggregate bandwidth. However, we do not see a corresponding increase in the number of active hosts, giving us the indication that a limited number of hosts contributed to the spike (in actuality, a single host). Also, we see a notable increase in active hosts in the 21st hour of the second day, with no increase in overall bytes transferred. This is a classic example of a linear scan propagating through the network. Note that by using both visualizations we are able to detect patterns that would have been obscured by using only the volume time-series or only the existence plot.

Existence plots are similar to heat maps, which have been applied to network traffic analysis by Fernandez-Campos et al. [2]. However, unlike heat maps, existence plots use only four discrete states to represent magnitude instead of continuous values represented by varying color intensity. Since varying time resolution can greatly affect the smoothness of magnitude changes (causing jarring color shifts in heat maps), we find the existence plot to be more informative as an augmentation of the time series.

Expected Benefits

With these visualizations, we expect added understanding of the current state of network hosts and quicker response to changes in network traffic. Specifically, we expect an increase in reaction time to changes in network traffic, primarily due to the analyst being able to develop intuition about what the data should look like when visualized.

2007 Accomplishments

This year we have accomplished four goals. First, we have conducted an exploratory analysis of animating Hilbert curves. In this investigation we were able to distinguish shifts in communication density according to the changes in time, a pattern primarily attributed to the seasonality of work days. Second, we have developed a proof-of-concept version of the sensor health visualization, which demonstrates the capabilities of the visualization method. We are currently putting the visualization into development. Third, we have cultivated the existence plot to better represent phenomena in our investigations. For instance, we have found it to be useful in representing beaconing behavior, or periodic phone home attempts, of compromised hosts. Finally, we have expanded the existence plot to represent a single host's port utilization, which will help in distinguishing internal hosts' behavioral patterns.

2008 Plans

In the coming year, we will expand on the current work with comparative analysis of our visualizations to existing methods and developing interactive visual methods. We intend to expand our Hilbert curve analysis by comparing it to other Internet-scale visualization methods, such as the tree map method demonstrated by Mansmann et al. [3]. Our aim is to provide an exhaustive analysis of the Hilbert curve's strengths and weaknesses. We will also explore the utility of existence plots when distinguishing between various types of behavior, such as the distinction between web servers and SMTP servers. Ultimately, we intend to produce interactive approaches that incorporate low-resolution plots, such as the existence plot with high-resolution plots, such as the Hilbert curve.

References

[1] Irwin, B. & Pilkington, N. "High Level Internet Level Traffic Visualization Using Hilbert Curve Mapping." *Proceedings of Workshop on Visualization for Computer Security (VizSEC).* Sacramento, CA, Oct. 2007. http://www.vizsec.org /workshop2007/presentations.html

[2] Fernandez-Campos, F., Nobel, A., Smith, F., & Jeffay, K. "Understanding Patterns of TCP Connection Usage with Statistical Clustering," 35–44. *Proceedings of 13th IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS).* Atlanta, GA, Sept. 2005. IEEE Computer Society Press, 2005.

[3] Mansmann F., Keim, D., North, S., Rexroad, B., & Sheleheda, D. "Visual Analysis of Network Traffic for Resource Planning, Interactive Monitoring, and Interpretation of Security Threats." *IEEE Transactions on Visualization and Computer Graphics 13,* 6 (Nov.-Dec. 2007): 1105–1112.

[4] Munroe, R. "Map of the Internet." http://xkcd.com/195/

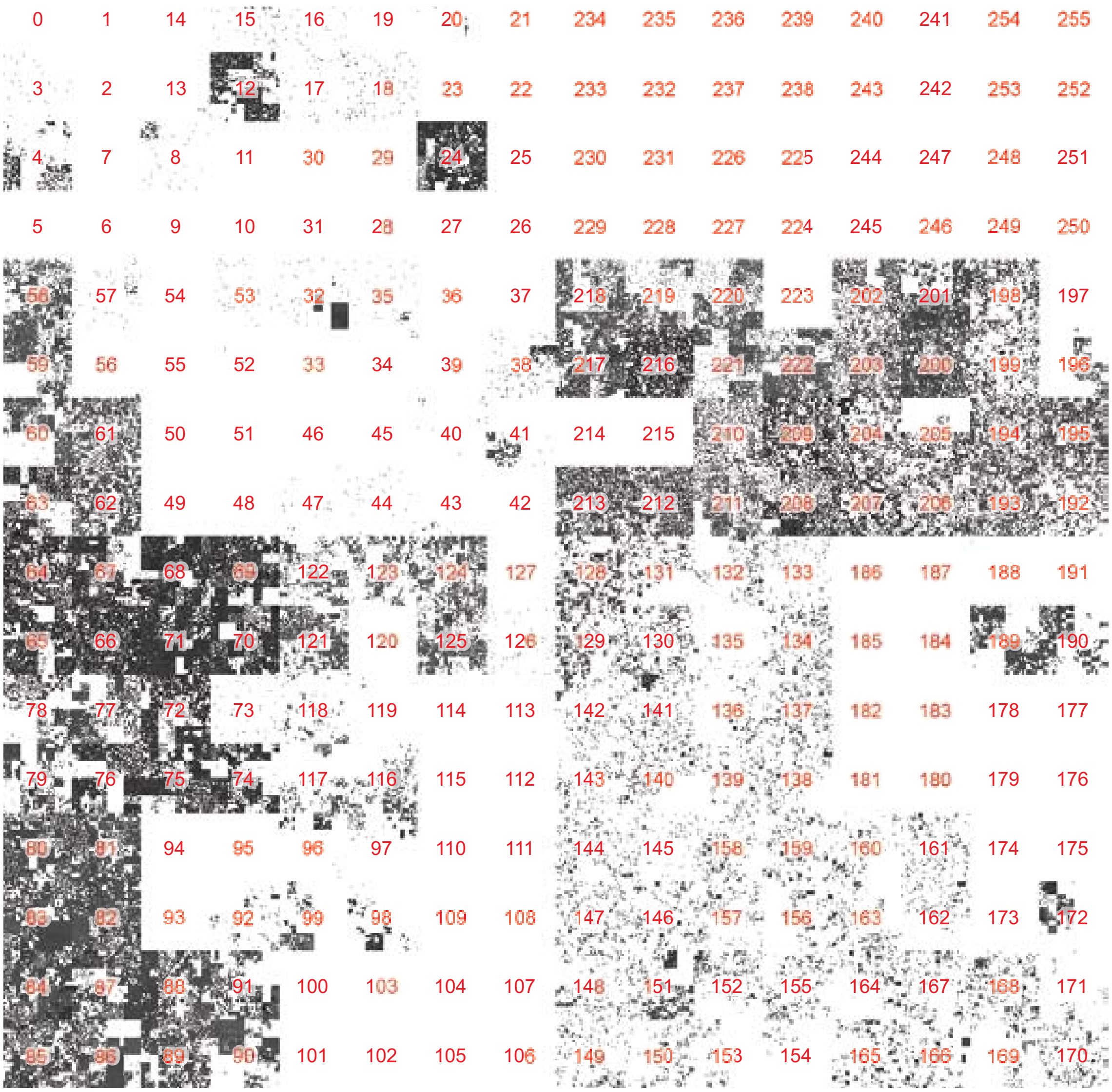


Figure 1: Hilbert Curve Representation of IP Space with /8 CIDR Blocks Labeled

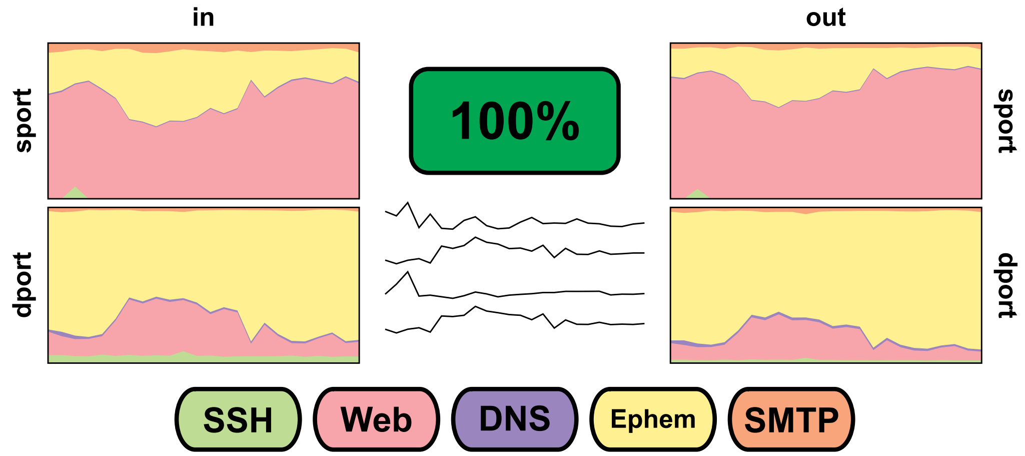


Figure 2: Sensor Health Visualization Displaying One Day of Activity

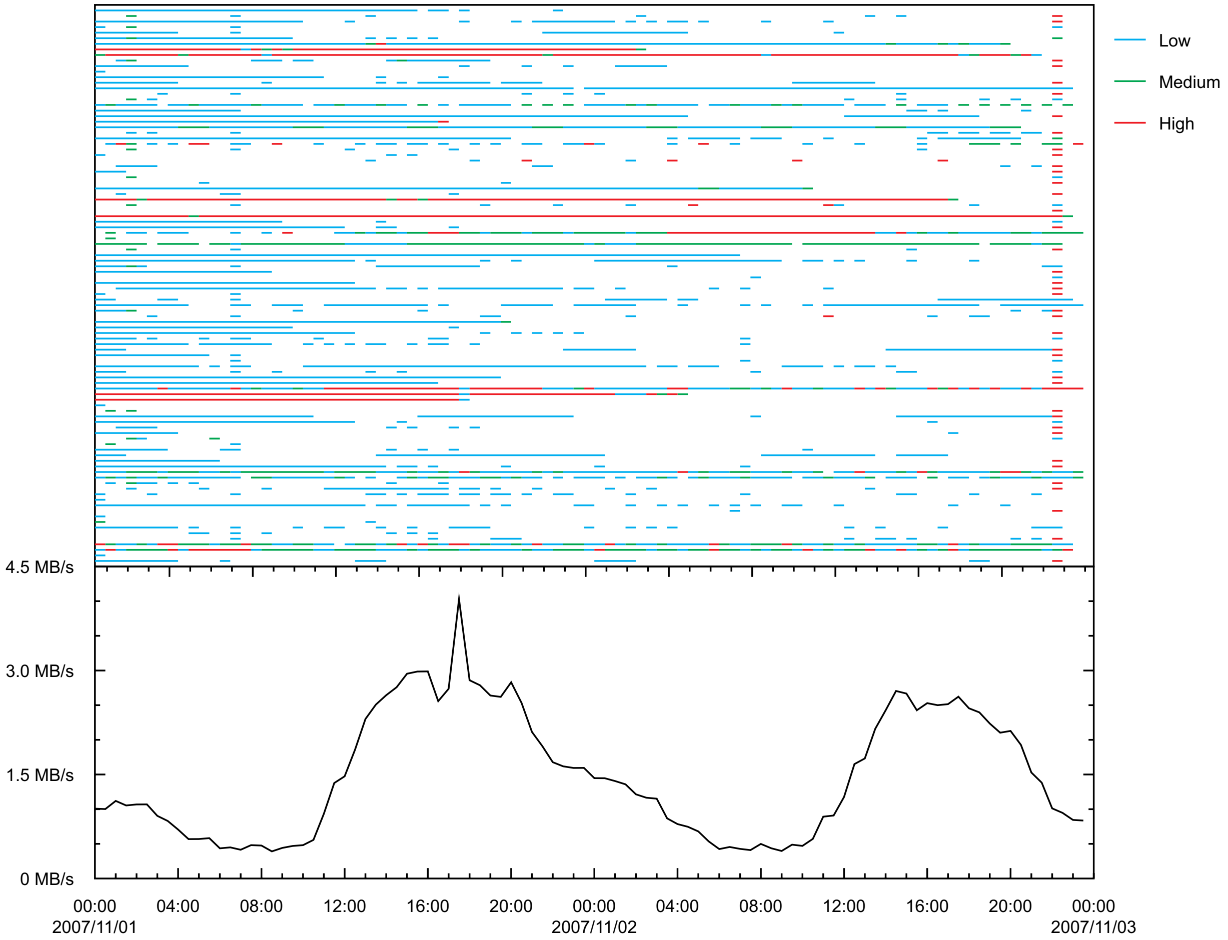


Figure 3: Existence Plot and Time Series Representing 100 Hosts' Bandwidth over Two Days

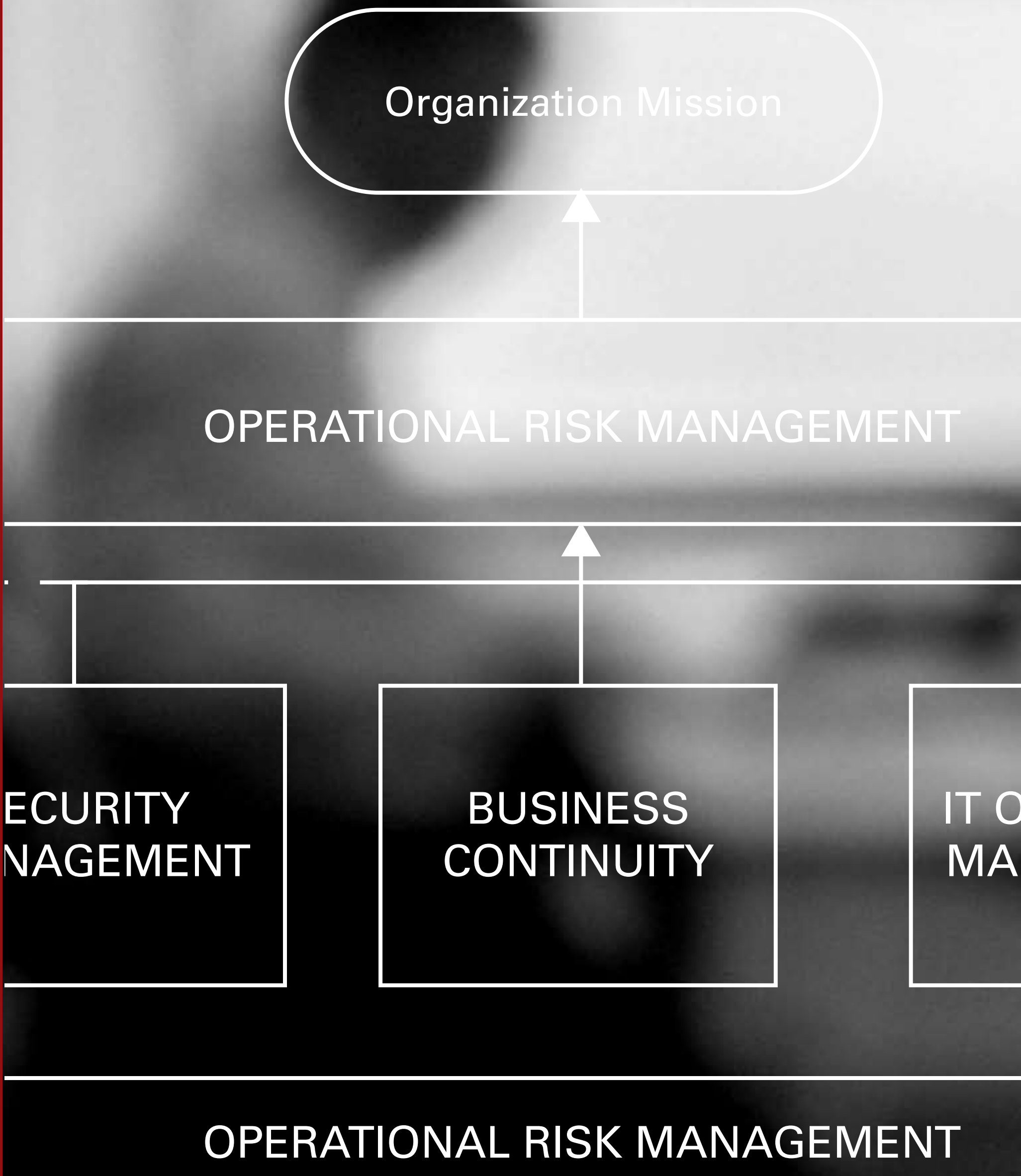
Resiliency Engineering Framework

Richard Caralli
412-268-9006

James F. Stevens
412-268-6935

David W. White
917-209-9284

Lisa R. Young
412-268-7700



Organization Mission

OPERATIONAL RISK MANAGEMENT

ECURITY
NAGEMENT

BUSINESS
CONTINUITY

IT O
MA

OPERATIONAL RISK MANAGEMENT

Resiliency Engineering Framework (REF)

Problem Addressed

Organizations in every sector—industry, government, and academia—are facing increasingly complex business and operational environments. Technology advances are helping organizations to automate business processes and make them more effective at achieving their mission. But the cost to organizations is that the technology is often more complex, takes specialized support and resources, and creates a rich environment for breeding vulnerabilities and risks. In addition to technology, organizations are also realizing that mission success relies on partnerships—engaging external partners to provide essential skills and functions, with the aim to increase productivity and reduce costs. As a result, the organization must expose itself to new risk environments, often in geographical regions of the world where emerging risks are not readily known. By employing a chain of partners to execute a business process, the organization cedes control and potential reliability of mission assurance in exchange for cost savings. This poses a problem for management in that governance and oversight must cross organizational and geographical lines like never before. And it must be acknowledged that the emerging worldwide socio-political environment is forcing organizations to consider threats and risks that have previously not been on their radar screens. Recent, well-publicized events have changed the view of what is feasible and expanded the range of outcomes that an organization must attempt to prevent and from which they must be prepared to recover. All of these new demands conspire to force organizations to rethink how they perform operational risk management and how they address the resiliency of critical business processes.

Traditional disciplines like security and business continuity must be expanded to provide protection and continuity strategies for critical assets that are commensurate with these new operating complexities. Unfortunately, current business and organizational approaches have not matured sufficiently to position organizations to effectively meet these challenges. For example, funding and staffing models tend to favor separation of security and business continuity activities into organizationally manageable silos rather than to view security and business continuity as activities that share responsibilities for holistic and convergent management of the entire risk equation—condition + consequence—with shared organizational goals. This path of least resistance often impairs the organization in the long run in making measurable and sustainable improvements in operational resiliency. This also reinforces the perception of senior management that security and business continuity are necessary evils that are funded because of uncertainty and fear or out of a need to comply, rather than because they enable the organization to meet strategic objectives.

In addition, organizations lack reliable means to assess their competency for managing operational resiliency. Typically, competency is measured by the way that an organization has performed during an event, or it is described in vague, unmeasurable terms. For example, when organizations are asked to describe how well they are managing resiliency, they typically revert to characterizing success in terms of events and consequences that they haven't been affected by. In other words, unless they are experiencing tangible pain, they conclude that their strategy must be working.

When organizations attempt to measure their competency for managing operational resiliency, they are prone to using tools and methods that are not purposeful for addressing competency and process maturity. Point-in-time reviews using codes of practice as a benchmark only provide comfort to the organization that they are doing the right things right now—they do not address how the organization will perform under times of stress or whether they will be able to sustain and repeat their successes in the long run. Because there will always be new and emerging threats, knowing how well the organization can perform today isn't as important as being able to predict how it will perform in the future when it encounters new events or a risk environment that is previously unknown.

Research Approach

CERT recognizes that organizations face challenges in managing operational resiliency in complex environments. The solution to addressing these challenges must have several dimensions. First and foremost, it must consider that security, business continuity, and IT operations management activities—typical operational risk management activities—are converging toward a continuum of practices that are focused on managing operational resiliency. Second, the solution must address the issues of measurement and metrics, providing a reliable and objective means for assessing competency and providing a basis for improving processes. And finally, the solution must help organizations to improve deficient processes—to close gaps that ultimately translate into weaknesses that diminish operational resiliency and impact the organization's ability to achieve strategic objectives.

Convergence

CERT's early research in this area concentrated on the shifting role of security from a technical specialty to a business and enterprise issue and competency. In this research, the challenges of security are characterized as a business problem that demands the attention of management and as a potential enabler to the organization in meeting its strategic objectives. Relative to this notion, security is described primarily as a risk management activity, rather than an IT management activity, having significant enterprise implications. But traditional security activities were deemed in our work to be too limiting with respect to operational risk management; in other words, the range of the types of operational risk are not expressly addressed by security activities (such as natural disasters) nor are the various components of operational risk (threat, actor, motive, and impact). To do this requires the contribution of other operational risk management activities such as business continuity and IT operations management.

Many organizations are now beginning to realize that security, business continuity, and IT operations management are complementary and collaborative functions that have the same goal: to improve and sustain operational resiliency. They share this goal because each function is focused on managing operational risk. This convergent view is often substantiated by popular codes of practice in each domain. For example, security practices now explicitly reference and include business continuity and IT operations management practices as an acknowledgement that security practices alone do not address both the conditions and consequences of risk. In CERT's research, we characterize this convergence of domains and the resulting continuum of practices across these domains as *resiliency engineering.* Figure 1 provides a graphic description of convergence.

Resiliency Engineering

Resiliency engineering is defined as the processes and related practices that an organization uses to design, develop, implement, and manage the protection and sustainability of business-critical services, related business processes, and associated assets such as people, information, technology, and facilities. It collectively addresses the prevention of operational risk (typically through security and IT operations management-related practices) as well as the management of organizational impact if risk is realized—both of which are necessary to manage operational resiliency comprehensively. To say that something has been "engineered" is to imply that a systematic process of design and construction originating from defined requirements has been undertaken [1]. Requirements are the foundation of all engineering-based processes, and the result of an engineered process is a product or service that substantially meets or exceeds all of the requirements that are established. Requirements also form the basis for managing operational resiliency. The protection and sustainability needs of an organizational service or asset are based on resiliency requirements that reflect how the service and related assets are used to support the organization's strategic objectives. When the organization fails to meet these requirements (either because of poor practices or as a result of disruptive events, threats, and risks), the operational resiliency of the service and assets is diminished, and one or more of the organization's strategic objectives fails to be met. Thus, operational resiliency depends on establishing requirements in order to build resiliency into assets and services and to keep these assets and services productive in the accomplishment of strategic objectives.

Through extensive review of existing codes of practice in the areas of security, business continuity, and IT operations management, as well as from experience with helping organizations to adopt a convergent view, CERT has codified a process definition for resiliency engineering processes in the CERT Resiliency Engineering Framework. The process definition embodies a requirements-driven foundation and describes the range of processes that characterize the organizational competencies necessary to actively direct, control, and manage operational resiliency.

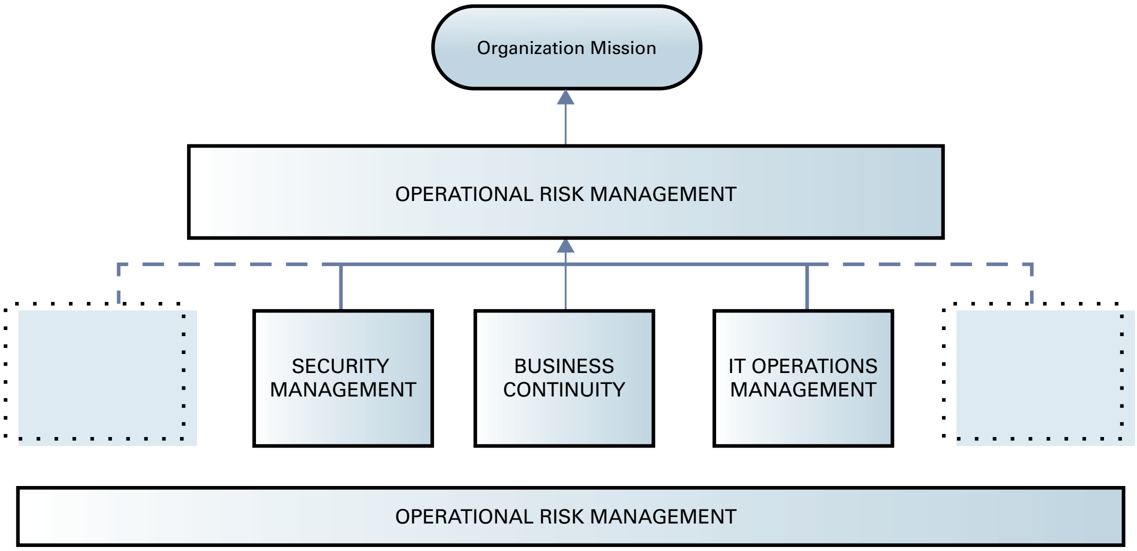


Figure 1: Foundation for Operational Resiliency Management

Providing a Process Improvement View

Defining the concept of resiliency engineering is not sufficient to help an organization transform from a security and business continuity perspective to one that is focused on resiliency and strategic objectives. While it provides a common understanding of the tasks that an organization must perform to manage operational resiliency, a simple definition of the resiliency engineering process will not provide sustainable process management and improvement. This is the domain of a process improvement approach.

As a process improvement model, the CERT Resiliency Engineering Framework seeks to allow organizations to use the process definition as a benchmark for identifying the current level of organizational competency, setting an appropriate and attainable desired target for performance, measuring the gap between current performance and targeted performance, and developing action plans to close the gap. By using the framework process definition as a foundation, the organization can obtain an objective characterization of performance not only against a base set of functional practices but also against practices that indicate successively increasing levels of process maturity. Thus, the organization is able to use the framework to determine its competency with an eye toward predicting its capabilities to perform consistently over time, to repeat its successes, and to perform reliably under times of stress.

More detailed discussion of the topics of convergence, resiliency engineering, and the application of a process improvement approach can be found in the technical notes *Managing for Enterprise Security* [2] and *Sustaining Operational Resiliency: A Process Improvement Approach to Security Management* [3], as well as other documents and presentations in the “Security and Resiliency Engineering” section of the “Organizational Security” portal on the CERT website (www.cert.org).

CERT Resiliency Engineering Framework

The CERT Resiliency Engineering Framework is the first step in the development of a process improvement approach to operational resiliency management. It has several key components. At the highest level, it is composed of over 25 competency areas that define the major elements of resiliency engineering. A competency area is an area of practice that the organization must master to an appropriate degree to manage operational resiliency. An organization can seek to improve its performance across all competency areas or select one or more areas in which to concentrate benchmarking and improvement efforts.

Framework Architecture

The architecture of the CERT Resiliency Engineering Framework is arranged in four categories:

- Enterprise Management
- Engineering
- Operations
- Process Management

These categories represent the broad range of activities that are important to managing operational resiliency. However, because resiliency engineering is a process that traverses the organization and is dependent on cooperation and coordination, these categories serve only as a way to group competencies by their common elements and focus. In reality, there is extensive interaction between competencies, and thus the categories provide a foundation from which interaction can be performed.

Enterprise Management

The enterprise is an important concept in the resiliency engineering process. At the enterprise level, the organization establishes and carries out many activities that the resiliency engineering process relies on. In addition, it provides the enterprise focus, oversight, and governance that is required for effective organizational and operational risk management. Typical competencies in this category include financial resource management, compliance, communications, organizational training and awareness, and risk management.

Engineering

The management of operational resiliency is a requirements-driven engineering function. Thus, the competencies in the Engineering category represent those that are focused on establishing and implementing resiliency for organizational assets, business processes, and services. These competencies establish the basic building blocks for resiliency and create the foundation for the protection and sustainability of assets and, by reference, business processes and services. Engineering competencies include asset definition and management, requirements definition, requirements management, service continuity, and controls management.

Operations

The Operations competencies represent the core activities for managing the operational resiliency of assets and services. These competencies are focused on sustaining an adequate level of operational resiliency as prescribed by the organization’s strategic drivers, critical success factors, and risk appetite. These competencies represent core security, business continuity, and IT operations and service delivery management activities and focus on the resiliency of information, technology, and facilities assets. Operations competencies include incident management and control, knowledge and information management, environmental control, technology management, vulnerability analysis and resolution, and sourcing.

Process Management

Process Management competencies represent those that are focused on measuring, managing, and improving the resiliency engineering process. These competencies establish the initial extension of process improvement concepts to the resiliency engineering process and, by default, to the disciplines of security and business continuity. Competencies in this category are intended to catalyze the organization's view of resiliency as a manageable and improvable process over which it has a significant level of control. Competencies in this area are expected to expand significantly as more process improvement concepts are introduced to the framework, but currently include measurement and analysis and monitoring.

A more detailed description of the CERT Resiliency Engineering Framework that includes a detailed outline of the full framework can be found in the technical report *Introducing the CERT Resiliency Engineering Framework: Improving the Security and Sustainability Processes* [4].

Benefits

A framework-based process improvement approach to resiliency engineering is meant to help an organization be more efficient and effective in managing operational resiliency. Specifically, the CERT Resiliency Engineering Framework aims to

- Create a common process definition. A common process definition for resiliency engineering can reduce the ambiguity and vagueness that results from traditionally ill-defined processes like security. It can also lay the foundation for future improvement because it provides a common understanding that can be discussed, debated, and revised.
- Create a common language. A common and sharable process definition brings a common language that can further reduce ambiguity, improve understanding and assimilation, and remove inhibitive barriers to growing a community around resiliency engineering. This is important not only for the organization itself but also in communications with suppliers, vendors, customers, regulators, and any external person or organization that needs to avoid issues that result from miscommunication.
- Provide a consistent benchmark for measurement. A common process definition and language are essential for establishing a capability benchmark. A benchmark can be a powerful tool for providing information on current performance, potential gaps, and strengths and weaknesses relative to other organizations. Benchmarking can also strengthen an entire industry by providing a way to communicate with regulators and lawmakers. And, organizations can use the benchmark to assess the capabilities of their vendors, customers, and other stakeholders who can affect their resiliency.
- Help organizations to eliminate redundancy and cost. The costs of managing operational resiliency continue to grow as organizations encounter new and increasingly unfamiliar risk environments. This results in mounting pressure to obtain funding (and to find new sources of funding) but also to be more cost effective and responsible in the ways that these funds (and other resources) are used. A process view forces the organization to look at how efficiently the process outcomes are being achieved and provides a foundation upon which the organization can make rational and, in some cases, quantitatively based decisions regarding the optimal redeployment of resources.
- Create viable process metrics. The ability to extract cost from managing operational resiliency and bring value to the organization is dependent upon being able to measure the effectiveness and efficiency of the resiliency engineering process. Organizations have become complacent in accepting the absence of data as a measurement of effectiveness of risk management activities. Therein lies the advantages of a process view—a process that can be defined can also be measured and controlled. Organizations are not left to wonder whether their investment has value or whether the end result of the process is achieved because a process view allows them to objectively measure it.
- Guide practice selection and implementation. A process perspective turns the organization's focus to the outcome of the process. Through a process improvement framework, a process view provides a descriptive structure in which the right prescriptive best practices for the organization can be implemented and integrated. With a process view, an organization is less inclined to implement practices without connecting them to processes that can be measured and improved. In addition, because the CERT Resiliency Engineering Framework is industry and domain agnostic, an organization can use any code of practice to achieve process goals. Thus, adoption of the framework does not require an organization to abandon practices that it currently uses and that have been successful.
- Provide a roadmap for maturing processes. One of the challenges for security or business continuity is the ability to sustain competency and success. The organization's current practices may appear to be effective in managing the protection and sustainability of critical assets and business processes, but the accomplishment may be temporal. As conditions in the operational environment change, the organization may not be able to sustain its competency or repeat its success because it has not established the institutionalizing structures and practices that it needs to adopt mature processes. As a benchmark, the CERT Resiliency

Engineering Framework will measure not only how well an organization is performing now (as is typical of existing assessment instruments), but whether its performance is sustainable, consistent, and repeatable over time.

2007 Accomplishments

In 2007, significant progress was made on the development of the CERT Resiliency Engineering Framework. The technical report *Introducing the CERT Resiliency Engineering Framework: Improving the Security and Sustainability Processes* [4] was published. This report describes the key concepts in resiliency engineering and describes a process improvement approach. It also includes a detailed outline of the framework. The outline serves as the current instantiation of the framework until a draft version is released in early 2008 for comment and review.

In 2007, benchmarking efforts were commenced with members of the Financial Services Technology Consortium (FSTC). FSTC provides a forum for financial institutions to come together to discuss technology-related problems and to develop working solutions, and has been a collaborative partner on the development of the framework since 2004. Using a subset of the framework's competency areas, FSTC member organizations have been benchmarking their performance against the model to characterize industry performance, validate the framework, and to begin process improvement efforts in individual organizations. (Information on the collaboration with FSTC can be found on their website at www.fstc.org.) Along with this benchmarking activity, CERT has developed a strawman appraisal methodology using concepts from the SCAMPI method (Standard CMM Appraisal Methodology for Process Improvement) that is used for CMMI appraisals.

Two pilots of the *Introduction to the Resiliency Engineering Framework* course were delivered to various groups in 2007 with the aim to provide broad education in resiliency engineering and process improvement in 2008.

2008 Plans

In 2008, a draft version of the full framework will be released for comment and review. CERT also plans to make significant progress in validating the framework and making it available for use for a much wider audience beyond our current set of collaborators. A fully functional appraisal methodology will be created to allow for independent competency appraisals using the framework as a foundation. And CERT plans to offer two courses in the area of resiliency engineering in 2008: one course focuses on the executive's role in sponsoring and initiating process improvement for resiliency engineering and the other is an introduction to the framework based on the pilot courses delivered in 2007.

In addition to publication of the draft version of the framework, a major focus of our work in 2008 will be the continuation of piloting and benchmarking activities. The benchmarking and piloting will include activities for

- diagnosis of current competencies through assessment
- definition of process improvement targets
- identification of process improvement gaps
- implementation of practices and subpractices to close gaps
- identification and measurement of process improvement results

These activities will not only help an organization to begin the transition to a process improvement approach but also provide much needed validation and refinement of the framework.
In addition, this activity will support the development and publication for guidance on framework adoption and process improvement later in 2008.

References

[1] MSN Encarta. *engineering*. http://encarta.msn.com/dictionary_/engineering.html (2007).

[2] Caralli, R. A. *Managing for Enterprise Security* (CMU/SEI-2004-TN-046). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2004. http://www.sei.cmu.edu/publications/documents/04.reports/04tn046.html

[3] Caralli, R. A. *Sustaining Operational Resiliency: A Process Improvement Approach to Security Management* (CMU/SEI-2006-TN-009). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2006. http://www.sei.cmu.edu/publications/documents/06.reports/06tn009.html

[4] Caralli, R. A. *Introducing the CERT Resiliency Engineering Framework: Improving the Security and Sustainability Processes* (CMU/SEI-2007-TR-009). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2007. http://www.sei.cmu.edu/publications/documents/07.reports/07tr009.html

[5] Chrissis, M. B., Konrad, M., & Shrum, S. *CMMI: Guidelines for Process Integration and Product Improvement.* Boston, MA: Addison-Wesley, 2003 (ISBN 0-321-15496-7).

Spam Detection at the ISP Level

Tim Shimeall
412-268-7611

Rhiannon Weaver
412-268-8511

Jeff Janies
412-268-8225

M. Patrick Collins
412-268-91241

Spam Detection at the ISP Level

Problem Addressed

The current state of the practice in defending against unwanted email (spam) involves the use of endpoint-centric solutions, such as Bayesian filtering. As spam volume rises and the spammer community shifts behavior to avoid detection, we expect the effectiveness of these solutions to decrease.

In the past, spammers have been limited by bandwidth requirements for spam generation. The new approach to spam generation uses botnets to send spam, allowing spammers to devote even more resources and resulting in a flood of traffic. Botnet-based spam generation provides considerable advantages to spammers, including lower-cost bandwidth, increased difficulty for blacklisting due to the large size of modern botnets, and increased anonymity of the spammer. Due to the size and diversity in location of botnets, a spammer does not have to worry about exhausting all the bandwidth from a single connection or ISP, thus increasing the volume of spam issued.

We define a spammer as a party earning profit from the generation of spam, and a spam source as an IP address observed emitting unwanted email. We describe a method for detecting sources of spam rather than attempting to detect individual spams or to trace spammers. This method forms part of an infrastructural spam solution. Other researchers' methods classified success as the email rejected. The increased use of botnets for spam implies that a new measure of success is distinguishing spam sources from those sending other email, allowing a source-based filter that could spare the processing load associated with previous work.

Research Approach

Our method is a payload-agnostic, flow-based mechanism for detecting hosts that send suspected spam. A payload-agnostic approach examines traffic without consideration of its content, such as flow-based analysis. In contrast to previous work, this method identifies source addresses that send spam exclusively, at least for short periods of time, rather than identifying specific email messages as spam. This method relies exclusively on flow data, which is compact and efficient to process. At a syntactic and protocol level, spam is acceptable email. Our method distinguishes normal email use from uses that are in a spammer's interest. Our results are empirically validated using traffic logs from active hosts on approximately five percent of the Internet public address space. We evaluate the success of our mechanism at distinguishing spam from other email and then evaluate the design decisions that result from this detection method: because our technique identifies distinctive *behavioral* characteristics of spam traffic, we can increase the cost to a spammer, if these approaches are adopted within an ISP-level network.

We differentiate email traffic using information in *flow records*, where a *flow* is a sequence of packets that have the same addressing information (source and destination addresses, source and destination ports, and protocol) that occur within a short time of each other. A single flow record may approximate one direction of a TCP session.

We will focus primarily on TCP flow records containing the SYN flag, the FIN flag, and at least some payload (a minimum of 60 bytes/packet), in order to constrain ourselves to legitimate SMTP traffic, as opposed to scans, attempted denial of service, or failed email connections. This constraint is applicable to a protocol such as SMTP, which is a non-interactive file transfer protocol; however, interactive protocols (e.g., Telnet or SSH), will demonstrate different behaviors.

Spammers tend to send out floods of email in hopes of making money by sales of advertised products or by a variety of frauds. Both sales and frauds generate returns based on the volume of messages that are read and accepted by the recipient, where "accepted" refers to either generating a reply email or some other action (such as buying a fraudulent stock) that is to the spammer's benefit. The rate of acceptance is proportional to the volume of solicitation or advertising and the penetration of new destination populations. From a traffic analysis point of view, this leads to three hypotheses:

1. (**the rate hypothesis**) that a source of spam will send messages at a faster rate than other email sources
2. (**the SMTP exclusivity hypothesis**) that a source of spam will generate traffic with a larger proportion of SMTP traffic to other traffic than other email sources
3. (**the locality hypothesis**) that a source of spam will not generate messages to a given address as regularly as other email sources

These hypotheses form the basis of the spam detection model and its associated operational method.

Expected Benefits

Compared with random selection of sources sending email to the addresses covered by the collection system, the validation results are significantly better, with $p<0.01$. These results are strong confirmation as to the validity of the method. Future work will include providing a trial implementation of a blocking method based on this analysis, to reduce server workload and decrease network bandwidth waste.

2007 Accomplishments

Starting with flows describing completed TCP traffic, our detection model is a sieve on source IP addresses, progressively eliminating sources that do not exhibit spam-consistent behavior. This progressive sieving allows for efficient implementation (i.e., pipelined or parallel algorithms) and extensibility or substitution as spammer behavior is better understood or changes.

First, sources within the recipient's email locality are eliminated. This locality is determined by the recipient *a priori* as the set of business- or mission-relevant sources that are not to be considered as spam even if they frequently contact the network. Eliminating these sources at the start results in two benefits. The possibility of misclassification of beneficial sources as spam sources is reduced. The number of sources that must be considered in the following, much more computationally complex, steps is also reduced.

Second, addresses sending low numbers of complete SMTP flows (less than 15) within a given time window (five minutes) are eliminated. Both the time window and the minimum number of SMTP flows required are key parameters at this step. It is at this point that the majority of email sources appear to be classified. The values here were derived from an empirical sensitivity analysis.

Last, addresses that do not demonstrate SMTP exclusivity within the five-minute window are eliminated. The key parameters here are the time window and the maximum ratio of non-SMTP flows to SMTP flows required (1%). This step appears to be largely confirmatory to the preceding steps.

To test the validity of these hypotheses, we collected TCP flows across several /8s on an arbitrary day (August 15, 2006), collecting all incoming flows across the border routers for these addresses. We then segregated the sources as spam and other sources (for testing purposes) using the AHBL[1] public spam blacklist.

We randomly chose 512 spam and 50 other addresses for detailed analysis and the subsequent testing of hypotheses. Monte Carlo estimates of each expected value in the previous section were calculated for 144 disjoint, 10-minute sampling windows using these addresses. As a first step, we assumed that the sample means were weakly stationary and independent across the day, and used the aggregated sample means calculated across 10-minute windows to estimate an overall time-independent average value for both the spam and non-spam IP addresses.

Figure 1 shows box plots of the resulting values for the rate estimator. Figure 1(a) shows the distribution of the mean for each source IP addresses. Figure 1(b) shows the distribution of the maximum values seen from each source IP addresses. The plot clearly shows a significant separation in the distribution of values. A two-sample *t*-test of the rate hypothesis rejects the null in favor of the alternative with high confidence ($p < 0.01$).

To test the exclusivity hypothesis, we compared the data in Figure 1 with the total number of incoming flows (as shown in Figure 2, with analogous diagrams) received from each source, classified as spam or non-spam sources. Again, the distributions show clear distinctions, and a two-sample *t*-test shows the null hypothesis for exclusivity is rejected with high confidence ($p<0.01$).

To test the locality hypothesis, we computed the number of incoming email sources (as opposed to flows in the previous tests) received by hosts on the network, again classified as spam sources and non-spam sources. The resulting counts are shown in Figure 3. Again, a two-sample *t*-test rejects the null hypothesis for the locality hypothesis with high confidence ($p<0.01$).

Based on these validated hypotheses, the detection method was implemented using a combination of C shell scripting and Python. The SiLK tool suite[2] was used to process network flows. The detection method was then validated against several data sets.

For the first data set used for validation, we analyzed a pool of known spam email (sent to a specific email address between November 1, 2005 and August 31, 2006), isolating IP addresses involved in its transmission. Of those addresses, 332 communicated with the addresses covered by the collection system during September 1-14, 2006. The detection method classified 58 addresses (approximately 17% of the 332) as generating spam. Of the 274 addresses not definitively identified as spammers, about 10% showed only non-SMTP activity, about 60% showed SMTP activity at a rate less than the threshold value used by the detection method, and the remainder showed a mix of activity that was not clearly related to spam.

The second data set used for validation was a pool of 50,000 IP addresses extracted from reported spam email to a specific address. Of the 39,000 of these addresses that communicated with an address covered by the collected flows during September 1-14, 2006, 10,500 were designated as spam by the detection method.

[1] http://www.ahbl.org/

[2] http://tools.netsa.cert.org/silk/

2008 Plans

During 2008, we expect to continue both to validate the method and to improve it based on information gained during further validation. We also are in preparation for deployment of this method in both analytical and operational environments. Experience in these pilot deployments will also yield improvements in the method.

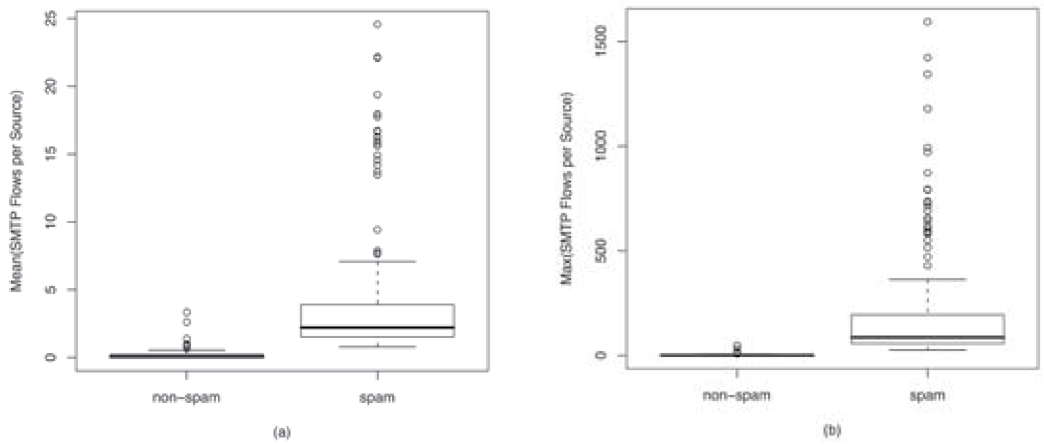


Figure 1: Distribution of (a) Mean and (b) Maximum Number of Incoming SMTP Flows from Each Source Address (10-Minute Windows)

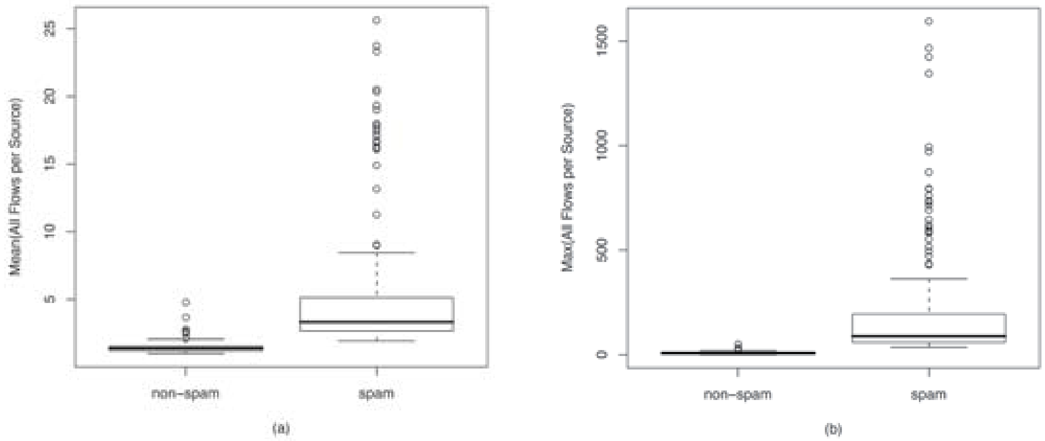


Figure 2: Distribution of (a) Mean and (b) Maximum Number of Incoming Flows (All Types) from Each Source Address (10-Minute Windows)

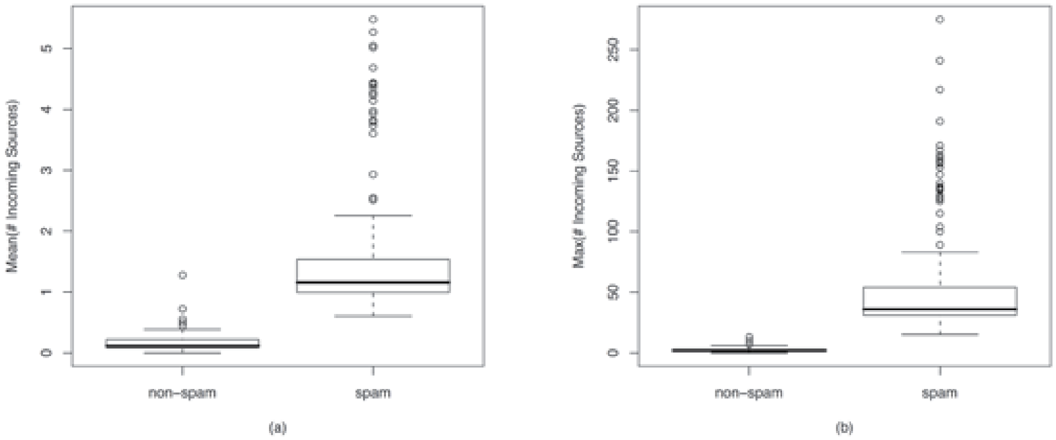


Figure 3: Distribution of (a) Mean and (b) Maximum Number of Incoming Sources of Email to each Destination Address (10-Minute Windows)

Nancy Mead
412-268-5756

SQUARE
Requirements Engineering for Improved System Security

	Step	Input
1	Agree on definitions	Candidate definitions fro IEEE and other standard
2	Identify security goals	Definitions, candidate goals, business drivers, policies and procedures examples
3	Develop artifacts to support security requirements definition	Potential artifacts (e.g., scenarios, misuse cases templates, forms)
4	Perform risk assessment	Misuse cases, scenarios security goals
5	Select elicitation techniques	Goals, definitions, candi date techniques, experti of stakeholders, organiz tional style, culture, leve of security needed, cost benefit analysis, etc.
6	Elicit security requirements	Artifacts, risk assessment results, selected techniques

SQUARE: Requirements Engineering for Improved System Security

Problem Addressed

It is well recognized in industry that requirements engineering is critical to the success of any major development project. Several authoritative studies have shown that requirements engineering defects cost 10 to 200 times as much to correct once fielded than if they are detected during requirements development [1,2]. Other studies have shown that reworking requirements, design, and code defects on most software development projects costs 40 to 50 percent of total project effort [3], and the percentage of defects originating during requirements engineering is estimated at more than 50 percent [4]. The total percentage of project budget due to requirements defects is 25 to 40 percent [5].

An earlier study found that the return on investment when security analysis and secure engineering practices are introduced early in the development cycle ranges from 12 to 21 percent, with the highest rate of return occurring when the analysis is performed during application design [6]. The National Institute of Standards and Technology (NIST) reports that software faulty in security and reliability costs the economy $59.5 billion annually in breakdowns and repairs [7]. The costs of poor security requirements make apparent that even a small improvement in this area will provide a high value. By the time an application is fielded and in its operational environment, it is very difficult and expensive to significantly improve its security.

Requirements problems are among the top causes of why

- projects are significantly over budget, past schedule, have significantly reduced scope, or are cancelled
- development teams deliver poor-quality applications
- products are not significantly used once delivered

These days we have the further problem that the environment in which we do requirements engineering has changed, resulting in an added element of complexity. Software development occurs in a dynamic environment that changes while projects are still in development, with the result that requirements are in flux from the beginning. This can be due to conflicts between stakeholder groups, rapidly evolving markets, the impact of tradeoff decisions, and so on.

When security requirements are considered at all during the system life cycle, they tend to be general lists of security features such as password protection, firewalls, virus detection tools, and the like. These are, in fact, not security requirements at all but rather implementation mechanisms that are intended to satisfy unstated requirements, such as authenticated access. As a result, security requirements that are specific to a system and that provide for protection of essential services and assets are often neglected. In addition, the attacker perspective is not considered, with the result that security requirements, when they exist, are likely to be incomplete. We believe that a systematic approach to security requirements engineering will help to avoid the problem of generic lists of features and to take into account the attacker perspective.

In reviewing requirements documents, we typically find that security requirements, when they exist, are in a section by themselves and have been copied from a generic set of security requirements. The requirements elicitation and analysis that is needed to define a better set of security requirements seldom takes place.

Much requirements engineering research and practice has addressed the capabilities that the system will provide. So while significant attention is given to the functionality of the system from the user's perspective, little attention is given to what the system should *not* do. In one discussion on requirements prioritization for a specific large system, ease of use was assigned a higher priority than security requirements. Security requirements were in the lower half of the prioritized requirements. This occurred in part because the only security requirements that were considered had to do with access control.

Research Approach

CERT has developed a methodology to help organizations build security into the early stages of the production life cycle. The Security Quality Requirements Engineering (SQUARE) methodology consists of nine steps that generate a final deliverable of categorized and prioritized security requirements. Although the SQUARE methodology could likely be generalized to any large-scale design project, it was designed for use with information technology systems.

The SQUARE process is most effective when conducted with a team of requirements engineers with security expertise and the stakeholders of the project. It begins with the requirements engineering team and project stakeholders agreeing on technical definitions that serve as a baseline for all future communication. Next, business and security goals are outlined. Then artifacts and documentation are created, which are necessary for a full understanding of the relevant system. A structured risk assessment determines the likelihood and impact of possible threats to the system.

Following this work, the requirements engineering team determines the best method for eliciting initial security requirements from stakeholders. This determination depends on several factors, including the stakeholders involved, the expertise of the requirements engineering team, and the size and complexity of the project. Once a method has been established, the participants rely on artifacts and risk assessment results to elicit an initial set of security requirements. Two subsequent stages are devoted to categorizing and prioritizing these requirements for management's use in making tradeoff decisions. Finally, an inspection stage is included to ensure the consistency and accuracy of the security requirements that have been generated.

SQUARE's nine discrete steps are outlined in Table 1. Each step identifies the necessary inputs, major participants, suggested techniques, and final output. Generally, the output of each step serves as the sequential input to the ensuing steps, though some steps may be performed in parallel. For instance, it might be more efficient for the requirements engineering team to perform Step 2 (Identify Security Goals) and Step 3 (Develop Artifacts) simultaneously, since to some extent they are independent activities. The output of both steps, however, is required for Step 4 (Perform Risk Assessment). In principle, Steps 1–4 are actually activities that precede security requirements engineering but are necessary to ensure that it is successful.

	Step	Input	Techniques	Participant	Output
1	Agree on definitions	Candidate definitions from IEEE and other standards	Structured interviews, focus group	Stakeholders, requirements team	Agreed-to definitions
2	Identify security goals	Definitions, candidate goals, business drivers, policies and procedures, examples	Facilitated work session, surveys, interviews	Stakeholders, requirements engineer	Goals
3	Develop artifacts to support security requirements definition	Potential artifacts (e.g., scenarios, misuse cases, templates, forms)	Work session	Requirements engineer	Needed artifacts: scenarios, misuse cases, models, templates, forms
4	Perform risk assessment	Misuse cases, scenarios, security goals	Risk assessment method, analysis of anticipated risk against organizational risk tolerance, including threat analysis	Requirements engineer, risk expert, stakeholders	Risk assessment results
5	Select elicitation techniques	Goals, definitions, candidate techniques, expertise of stakeholders, organizational style, culture, level of security needed, cost benefit analysis, etc.	Work session	Requirements engineer	Selected elicitation techniques
6	Elicit security requirements	Artifacts, risk assessment results, selected techniques	Joint Application Development (JAD), interviews, surveys, model-based analysis, checklists, lists of reusable requirements types, document reviews	Stakeholders facilitated by requirements engineer	Initial cut at security requirements
7	Categorize requirements as to level (system, software, etc.) and whether they are requirements or other kinds of constraints	Initial requirements, architecture	Work session using a standard set of categories	Requirements engineer, other specialists as needed	Categorized requirements
8	Prioritize requirements	Categorized requirements and risk assessment results	Prioritization methods such as Triage, Win-Win, etc.	Stakeholders facilitated by requirements engineer	Prioritized requirements
9	Requirements inspection	Prioritized requirements, candidate formal inspection technique	Inspection method such as Fagan, peer reviews, etc.	Inspection team	Initial selected requirements, documentation of decision-making process and rationale

Table 1: Security Requirements Elicitation and Analysis Process

The SQUARE process has been baselined, and the baseline was defined in a technical report [8]. SQUARE was also described in the requirements engineering section of the Build Security In website [9], and in two books [10, 11]. Several case studies with real-world clients have shown that the methodology holds good promise for incorporation into industry practice, and we are working informally with additional industry clients. The current model is summarized in Table 1. CERT is currently continuing research and application of the process and is working in parallel to create a CASE tool to support each stage of the methodology.

Expected Benefits

When SQUARE is applied, the user should expect to have identified, documented, and inspected relevant security requirements for the system or software that is being developed. SQUARE may be more suited to a system under development or one undergoing major modification than one that has already been fielded, although it has been used both ways.

2007 Accomplishments

In conjunction with Carnegie Mellon University's CyLab, the prototype tool was completed and released. Also in conjunction with CyLab, workshop, tutorial, and educational materials on SQUARE were produced. The tutorial was presented at the CyLab subscribers' meeting. A Distinguished Lecture on SQUARE was delivered at Nortel. The initial version of SQUARE-Lite, a four-step process extracted from SQUARE, was developed and is being applied in a client organization. A technical note on SQUARE [12] discussed ways of comparing SQUARE with other security requirements engineering methods. Papers on the cost/benefit aspects of SQUARE and on security requirements elicitation were also published [13, 14].

2008 Plans

We are working with two client organizations to incorporate SQUARE into their development processes. We will seek to work with a commercial vendor to further develop the tool as a commercial product. Workshop, tutorial, and academic educational materials on SQUARE will be made available on the CERT website. SQUARE will be further documented in a chapter of a book based on the Build Security In website. The book is titled *Secure Software Engineering: A Guide for Project Managers* and will be published by Addison-Wesley in the first half of 2008.

References

[1] Boehm, B. W. & Papaccio, P. N. "Understanding and Controlling Software Costs." *IEEE Transactions on Software Engineering SE-4*, 10 (October 1988): 1462–77.

[2] McConnell, S. "From the Editor - An Ounce of Prevention." *IEEE Software 18,* 3 (May 2001): 5–7.

[3] Jones, C., ed. *Tutorial: Programming Productivity: Issues for the Eighties,* 2nd ed. Los Angeles, CA: IEEE Computer Society Press, 1986.

[4] Wiegers, K. E. "Inspecting Requirements" (column). StickyMinds.com, July 30, 2001. http://www.stickyminds.com

[5] Leffingwell, D. & Widrig, D. *Managing Software Requirements—A Use Case Approach, 2nd ed.* Boston, MA: Addison-Wesley, 2003.

[6] Soo Hoo, K., Sudbury, A. W., & Jaquith, A. R. "Tangible ROI through Secure Software Engineering." *Secure Business Quarterly 1*, 2 (2001). http://www.musecurity.com/assets/files/Tangible%20ROI%20Secure%20SW%20Engineering.pdf

[7] National Institute of Standards and Technology. "Software Errors Cost U.S. Economy $59.5 Billion Annually" (NIST 2002-10). http://www.nist.gov/public_affairs/releases/n02-10.htm (2002).

[8] Mead, N. R., Hough, E., & Stehney, T. *Security Quality Requirements Engineering (SQUARE) Methodology* (CMU/SEI-2005-TR-009). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2005. http://www.sei.cmu.edu/publications/documents/05.reports/05tr009.html

[9] Department of Homeland Security. *Build Security In.* https://buildsecurityin.us-cert.gov/ (2008).

[10] Mead, N. R., Davis, N., Dougherty, C., & Mead, R. Ch. 8, "Recommended Practices," 275–308. *Secure Coding in C and C++*, Robert Seacord. Upper Saddle River, NJ: Addison Wesley, 2005.

[11] Mead, N. R. Ch. 3, "Identifying Security Requirements Using the SQUARE Method," 44–69. *Integrating Security and Software Engineering: Advances and Future Visions.* Edited by H. Mouratidis and P. Giorgini. Hershey, PA: Idea Group, 2006 (ISBN 1-59904-147-2).

[12] Mead, N. R. *How To Compare the Security Quality Requirements Engineering (SQUARE) Method with Other Methods* (CMU/SEI-2007-TN-021). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2007. http://www.sei.cmu.edu/publications/documents/07.reports/07tn021.html

[13] Caulkins, J., Hough, E. D., Mead, N. R., & Osman, H. "Optimizing Investments in Security Countermeasures: A Practical Tool for Fixed Budgets." *IEEE Security & Privacy 5*, 5 (September/October 2007).

[14] Mead, N. R. "Experiences in Eliciting Security Requirements." *CrossTalk 19*, 12 (December 2006): 14–19.

Richard Linger
301-926-4858

Gwendolyn H. Walton
412-268-7700

Mark Pleszkoch
412-268-7700

Kirk Sayre
412-256-7700

Stacy Prowell
412-268-9205

STAR*Lab

Computed Behavior

```
⊟ condition:  ? true

   ⊟ Registers
      EAX  := 0
      ECX  := ((EAX   *d EBX) +d ECX)
      ECX  := case((0  ==EAX)  -> EDX | (0
   ⊟ Flags
      AF := false
      CF := false
      OF := false
      PF := true
      SF := false
      ZF := true
   ⊟ Memory
      label  :=  "exit"
     External
```

STAR*Lab: A Software Development Laboratory for Security Technology Automation and Research

Developing Engineering Automation for Challenge Problems in System Security

CERT has established a software development laboratory in response to the growing needs of its customers. The mission of STAR*Lab (Security Technology Automation and Research Laboratory) is development of theory-based prototype automation that provides solutions to challenge problems in security engineering and software assurance.

Challenge problems are long-standing barriers to progress identified by the Department of Defense (DoD) and other organizations whose solutions can have substantial impact on engineering capabilities. The focus of STAR*Lab is not on producing studies and reports that may leave implementation speculative and undone, but rather on applying theory to develop working tools. The purpose of the laboratory is to help its sponsors achieve three objectives:

1. Faster development. Solutions must replace time- and resource-intensive operations with engineering automation that permits faster system development.
2. Improved quality. Solutions must augment human processes with foundations-based automation to improve system security and dependability.
3. Fewer resources. Solutions must increase the span of intellectual control through automation for more effective use of resources in developing systems.

The laboratory operates according to three principles:

1. Foundations-first principle. Theoretical foundations are necessary to ensure completeness and correctness in automated solutions and confidence in the results they produce. All projects start with sound foundations to avoid ad hoc solutions with limited applicability.
2. Proof-by-automation principle. Automation is essential to replace resource-intensive human operations with solutions that augment intellectual control. All projects will demonstrate solutions through automated engineering tools.
3. Practical application principle. Automation must solve challenge problems with practical engineering operations for routine use by practitioners. All projects will scale up engineering solutions for widespread application.

STAR*Lab projects are managed within a gated review structure designed to maintain visibility, reduce risk, and ensure effective use of sponsor resources. Projects must satisfy the requirements of each gate to receive funding to progress to the next gate:

- Gate 1: Challenge problem definition. Each project must address a barrier to progress through a project plan that defines team composition, tasks, and schedules.
- Gate 2: Theoretical feasibility. Each project must identify theoretical foundations to avoid heuristic or partial approaches of limited value for achieving a comprehensive solution.
- Gate 3: Proof-of-concept automation. Each project must develop prototype automation that demonstrates application of the theoretical foundations.
- Gate 4: Scale-up for application. Each project must evolve the prototype automation to scale up engineering capabilities for routine application.

STAR*Lab is currently engaged in the Function Extraction (FX) for Software Assurance project. This multiyear effort has satisfied the requirements of Gate 3 and is progressing to Gate 4. In addition, the laboratory is ready to capitalize on function extraction technology in four potential FX-based project areas:

- Computational Security Attributes
- Software Correctness Verification
- System Component Composition
- Flow-Service-Quality Engineering

These projects are described in the next sections.

STAR*Lab Function Extraction for Software Assurance: Engineering Automation for Computing Software Behavior

Richard Linger

Problem Addressed

STAR*Lab recognizes both the importance of software assurance to national defense and economic security and the difficulty of achieving it. Software assurance depends on knowing and verifying the complete behavior of software. Unfortunately nothing less will do, because behavior that is not known can contain errors, vulnerabilities, and malicious content. It is a sobering fact that current software engineering provides no practical means for developers to determine the full behavior of software, and no testing effort, no matter how elaborate, can exercise more than a small fraction of possible behavior. Complex software systems are difficult to understand because of their immense numbers of execution paths, any of which can contain errors and security exposures. Faced with innumerable execution possibilities, developers and analysts often achieve no more than a general understanding of system behavior. This technology gap is at the heart of many issues in software and security engineering. Simply put, systems experience errors and vulnerabilities in large measure because their developers have no practical means to determine what they do in all possible uses.

Research Approach

While software assurance has been limited by engineering capabilities in the past, it may be less so in the future. Function-theoretic foundations of software illuminate a challenging but feasible strategy for developing automated tools to calculate the behavior of software and present it to users in understandable form. STAR*Lab is conducting research and development in the emerging technology of function extraction (FX). The objective of FX is to move from an uncertain understanding of program behavior derived in a human time scale of days to a precise understanding automatically computed in a machine time scale of seconds. This technology applies function-theoretic mathematical foundations to automate calculation of the functional behavior of software to the maximum extent possible. These foundations define the transformation of code structures into procedure-free functional form and are the basis for the function extraction process [1,2]. While theoretical results impose some constraints on behavior calculation (for example, for certain forms of loops), STAR*Lab development of engineering solutions suggests that nearly all software behavior will be amenable to calculation. And any level of behavior calculation can help improve human capabilities for understanding and analysis.

To explore the impact of FX technology, STAR*Lab developed a proof-of-concept function extractor prototype that calculates the behavior of programs expressed in a small subset of the Java programming language. In a controlled experiment, the group using the FX prototype reduced the time required to derive the functional behavior of example programs by several orders of magnitude, and was about four times better at providing correct answers to comprehension and verification questions in a fourth of the time, compared to the control group [3].

Function extraction technology can be applied to any programming language and has the potential to impact many aspects of the software engineering life cycle. To better understand this impact, STAR*Lab conducted an SEI-sponsored study with a major corporation to determine how FX could improve engineering operations in activities ranging from software specification and design to implementation and testing [4]. This study produced guidance for FX evolution from experienced software developers, including the following recommendations:

- Development of FX automation for assembly language should be a priority.
- FX automation should be developed for correctness verification of software.
- FX automation should be developed for high-level languages, starting with Java.
- Research on FX automation for specification and architecture should be initiated.

Expected Benefits

The function extraction system currently under development targets programs written in or compiled into Intel assembly language. The system is expected to help security analysts to determine intruder strategies by providing precise information on the structure and function of malicious code [5]. In terms of broader application, opportunities exist to make progress on the problems of malicious code detection, computational security analysis, correctness verification, legacy system understanding, creation of assured software repositories, and automated component composition. The basis for all of this is the realization that programs are mathematical artifacts subject to mathematical analysis. Human fallibility may still exist in interpreting the analytical results, but there can be little doubt that routine availability of calculated behavior would help reduce errors, vulnerabilities, and malicious code in software and make software development more manageable and predictable.

2007 Accomplishments

In its current state of development, the FX system demonstrates (a) transformation of spaghetti-logic assembly language programs into understandable structured form and (b) automated computation of behavior for sequence, alternation, and iteration structures. For example, Figure 1 demonstrates behavior computation for the miniature program shown on the left (depicting the result of the FX system transforming the original spaghetti-logic version into structured form; jump instructions are retained in the program text as comments for traceability but have no effect). The program has an initialized nested loop structure. The computed behavior on the right shows that the net behavioral effect of the program is to always (condition is true) carry out the following concurrent assignments:

- set register EAX to 0
- set register ECX to the product of the initial values of EAX and EBX plus the initial value of ECX, and
- if initial EAX is 0, leave EDX unchanged; otherwise, if initial EAX is not 0, set EDX to 0

In terms of malware analysis, the current system can demonstrate examples of behavior computation for (a) viruses with repeatedly obfuscated control logic, (b) viruses hidden in large programs, and (c) repeatedly obfuscated virus unpackers, including use of computed behavior to unpack the virus payload.

2008 Plans

FX system development is planned to continue in 2008. Sponsors are welcome to participate in completing the system and moving the technology forward. STAR*Lab is also ready to apply FX to additional languages and phases of the software engineering life cycle.

References

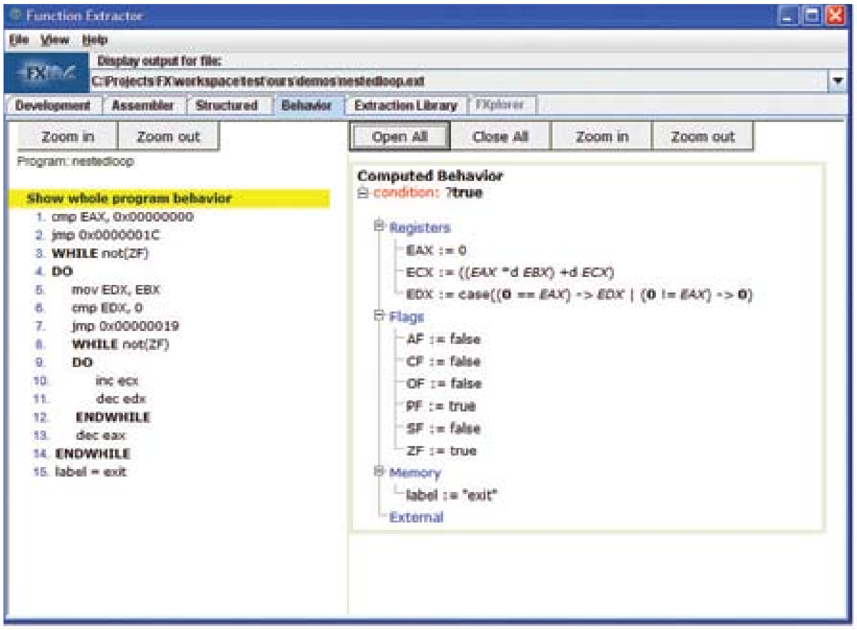
[1] Prowell, S., Trammell, C., Linger, R., & Poore, J. *Cleanroom Software Engineering: Technology and Practice.* Reading, MA: Addison Wesley, 1999.

[2] Mills, H. & Linger, R. "Cleanroom Software Engineering." *Encyclopedia of Software Engineering, 2nd ed.* Edited by J. Marciniak. New York, NY: John Wiley & Sons, 2002.

[3] Collins, R., Walton, G., Hevner, A., & Linger, R. *The CERT Function Extraction Experiment: Quantifying FX Impact on Software Comprehension and Verification* (CMU/SEI-2005-TN-047). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2005. http://www.sei.cmu.edu/publications/documents/05.reports/05tn047.html

[4] Hevner, A., Linger, R., Collins, R., Pleszkoch, M., Prowell, S., & Walton, G. *The Impact of Function Extraction Technology on Next-Generation Software Engineering* (CMU/SEI-2005-TR-015). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2005. http://www.sei.cmu.edu/publications/documents/05.reports/05tr015.html

[5] Pleszkoch, M. & Linger, R. "Improving Network System Security with Function Extraction Technology for Automated Calculation of Program Behavior." *Proceedings of the 37th Hawaii International Conference on System Sciences* (HICSS-37). Waikoloa, HI, Jan. 5-8, 2004. Los Alamitos, CA: IEEE Computer Society Press, 2004.



Program: nestedloop

```
cmp EAX, 0x00000000
jmp 0x0000001C
WHILE not(ZF)
DO
    mov EDX, EBX
    cmp EDX, 0
    jmp 0x00000019
    WHILE not(ZF)
    DO
        inc ecx
        dec edx
     ENDWHILE
     dec eax
 ENDWHILE
 label = exit
```

```
Computed Behavior
⊟ condition: ?true
  ⊟ Registers
    ├ EAX := 0
    ├ ECX := ((EAX *d EBX) +d ECX)
    └ EDX := case((0 == EAX) -> EDX | (0 != EAX) -> 0)
  ⊟ Flags
    ├ AF := false
    ├ CF := false
    ├ OF := false
    ├ PF := true
    ├ SF := false
    └ ZF := true
  ⊟ Memory
    └ label := "exit"
  └ External
```

Figure 1: FX System Computation of the Behavior of a Miniature Looping Program

STAR*Lab Computational Security Attributes: Engineering Automation for Software Security Analysis

Gwendolyn H. Walton

Problem Addressed

Security strategies must be sufficiently dynamic to keep pace with organizational and technical change. However, in the current state of practice, security properties of software systems are often assessed through labor-intensive human evaluation. The results can be of limited value in the dynamics of system operation where threat environments and security attributes can change quickly. The Computational Security Attributes project takes a fundamentally different approach, focusing on the question "What can be computed with respect to security attributes?" to develop theory-based foundations for defining and computing attribute values with mathematical precision [1].

The ultimate goal of the project is to provide foundations to help transform security engineering into a theory-based computational discipline. Achieving this goal will require development of mathematical foundations and corresponding automation to permit both rigorous evaluation and improvement of security attributes of software during development and real-time evaluation of security performance during operation.

Research Approach

The problem of determining the security properties of programs comes down in large measure to the question of how they behave when invoked with stimuli intended to cause harmful outcomes. Thus, the first step in security analysis is to understand program behavior at a level of completeness and correctness that is generally impractical with current technology. The emergence of STAR*Lab's new function extraction (FX) technology, unavailable to previous researchers, provides the basis for this critical first step by computing the functional behavior of programs as a starting point for security analysis. The foundations of FX treat programs as rules for mathematical functions or relations that can be computed from program logic. These foundations can be generalized to accommodate what are often termed *non-functional* properties, in this case security properties, but which in reality exhibit functional characteristics amenable to computational approaches [2].

Automated evaluation of software security attributes consists of three major steps:

1. Specify security attributes in terms of required functional behavior for the operational environment of the software.
2. Apply FX technology to the software to compute a behavior database that specifies its as-built functional behavior.
3. Perform computational analysis to verify that the behavior is correct with respect to required security attribute behavior.

The properties analyzed in the project include authentication, authorization, non-repudiation, confidentiality, privacy, and integrity.

Expected Benefits

There are several advantages to this approach:

- A rigorous method is used to specify security attributes in terms of the actual behavior of code during execution.
- The security properties of code can be checked through analysis of computed behavior.
- The specified security behaviors provide requirements for a security architecture.
- Vulnerabilities can be better understood, making it easier to address evolution of code and its usage environment.
- The use of constraints provides a mechanism for explicitly defining all assumptions.

Computational security attribute technology can address specification of security attributes of software systems before they are built, specification and evaluation of security attributes of acquired software, verification of as-built security attributes of software, and real-time evaluation of security attributes during system operation.

2007 Accomplishments

The evolving FX system was employed to demonstrate detection of security attribute violations involving the presence of malware embedded in software.

2008 Plans

Interested organizations are invited to sponsor development of FX-based engineering tools for computational evaluation of security attributes.

References

[1] Linger, R., Pleszkoch, M., Walton, G., & Hevner, A. *Flow-Service-Quality (FSQ) Engineering: Foundations for Network System Analysis and Development* (CMU/SEI-2002-TN-019). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002. http://www.sei.cmu.edu/publications/documents/02.reports/02tn019.html

[2] Walton, G., Longstaff, T., & Linger, R. *Technology Foundations for Computational Evaluation of Software Security Attributes* (CMU/SEI-2006-TR-021). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2006. http://www.sei.cmu.edu/publications/documents/06.reports/06tr021.html

STAR*Lab Software Correctness Verification: Engineering Automation for Software Assurance

Mark Pleszkoch

Problem Addressed

Software containing errors and vulnerabilities cannot be trustworthy or secure. Yet despite best efforts, software is often developed and delivered with incorrect and even unknown behavior. In the current state of practice, no practical means exists for automation support of large-scale correctness verification of software with respect to intended behavior. As a result, much time and energy is devoted to inspection and testing activities that can provide only limited evidence of correctness.

Research Approach

The objective of this potential project is to develop a prototype function verification system that will help users to check the correctness of programs based on their actual computed behavior. The system will employ the mathematics-based foundations of function extraction to achieve completeness and correctness of results, but the user will not be exposed to, or required to know, these foundations. The system will provide a proof of concept for function verification technology and a foundation for elaboration into industrial-strength verification systems. In addition, the system will provide a standard, machine-processable form for representing intended behavior. Users will be able to code programs to satisfy intended behavior and execute the system to check correctness.

Function extraction and function verification are closely related. Functional correctness verification requires computing the as-built functional behaviors of program structures, just as in the function extraction process, and then comparing those behaviors to intended behaviors for equivalence or not. As noted, the function-theoretic model of software treats programs as rules for mathematical functions or relations—that is, mappings from domains to ranges. While programs can contain an intractable number of execution paths, they are at the same time composed of a finite number of control structures, each of which implements a mathematical function or relation in the transformation of its inputs into outputs. A theorem defines the mapping of these control structures into procedure-free functional form [1,2]. These mappings are the starting point for the function extraction process and its application to correctness verification.

Expected Benefits

Large-scale software assurance requires a commensurate scale of engineering automation. This project can provide substantial benefits to sponsors who must deal with software failures and vulnerabilities in enterprise operations. It is difficult to achieve trustworthiness and security goals for systems without knowing whether they are correct with respect to intended behavior. Routine availability of functional verification can substantially reduce errors, vulnerabilities, and malicious code in software. FX-based verification technology can replace much of the labor-intensive and error-prone work of program inspection and testing, with corresponding reductions in resource requirements and improvements in product quality [3].

2007 Accomplishments

The function extraction system currently under development provides a foundation for implementing correctness verification capabilities.

2008 Plans

STAR*Lab is ready to extend FX technology for automation of correctness verification for interested sponsors.

References

[1] Prowell, S., Trammell, C., Linger, R., & Poore, J. *Cleanroom Software Engineering: Technology and Practice.* Reading, MA: Addison Wesley, 1999.

[2] Mills, H. & Linger, R. "Cleanroom Software Engineering." *Encyclopedia of Software Engineering, 2nd ed.* Edited by J. Marciniak. New York, NY: John Wiley & Sons, 2002.

[3] Hevner, A., Linger, R., Collins, R., Pleszkoch, M., Prowell, S., & Walton, G. *The Impact of Function Extraction Technology on Next-Generation Software Engineering* (CMU/SEI-2005-TR-015). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2005. http://www.sei.cmu.edu/publications/documents/05.reports/05tr015.html

STAR*Lab System Component Composition: Engineering Automation for Understanding System Behavior

Kirk Sayre

Problem Addressed

Modern systems are characterized by large-scale heterogeneous networks with many components that must be correctly integrated to achieve mission objectives. It is often the case that the components are complex systems in their own right and must be dynamically integrated to provide end-to-end capabilities. System integration today is a complex, labor-intensive process that can require substantial effort for large systems. Automation support for behavior analysis of component compositions could help reduce the time and effort required to achieve operational capabilities [1].

Research Approach

This potential project will define the extent to which component compositions can be automatically calculated. Automation support for determining composite behavior of components architected into systems could enable fast and reliable understanding and development. Composition computation must generate mathematically correct abstractions of behavior at any level and help scale up the reliable unit of construction for systems. Because behavior calculation is essentially a compositional task, function extraction is the key underlying technology for component composition. FX produces behavior databases of individual programs; the databases themselves can be composed to reveal the composite behavior of the programs when combined into systems.

Expected Benefits

Automated derivation of the net effect of program compositions can reveal combined functionality, illuminate mismatches, facilitate analysis of design alternatives, and support evaluation of commercial off-the-shelf products. This approach can also guide rapid and reliable refactoring of components and systems in responding to new system requirements.

2007 Accomplishments

Research and development carried out in the FX project has direct applicability to automated composition of components.

2008 Plans

A key step toward creation of an automated composition capability is extension of FX technology to create a proof-of-concept prototype. Sponsors are welcome to join in this effort.

Reference

[1] Feiler, P., Goodenough, J., Linger, R., Longstaff, T., Kazman, R., Klein, M., Northrop, L., Wallnau, K., Gabriel, R., Schmidt, D., & Sullivan, K. *Ultra-Large-Scale Systems: The Software Challenge of the Future.* Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, June, 2006. http://www.sei.cmu.edu/uls

STAR*Lab Flow-Service-Quality (FSQ) Engineering: Foundations for Developing Network-Centric Systems

Stacy Prowell

Problem Addressed

Modern society is dependent on large-scale, network-centric systems whose complexity can often exceed engineering capabilities for intellectual control. The result can be frustrations and delays in development and failures and compromises in operation. Intellectual control does not mean the absence of uncertainties or failures—they are inevitable—but rather the capability to address them in a rigorous engineering framework. System complexity and survivability are closely related. Complexity diminishes survivability by masking errors and vulnerabilities and hiding unforeseen paths for intrusion. The survivability of complex systems that support national infrastructures is of particular concern. The problem lies not with developers but with the lack of engineering methods to cope with system complexities. More effective engineering technology is required across the life cycle for fast and precise development and evolution of network-centric systems.

A promising path lies in the investigation of mathematical foundations that can accommodate the realities of large-scale networked systems: highly distributed heterogeneous components, shifting boundaries and users, uncertain commercial off-the-shelf component function and quality, extensive asynchronous operations, unpredictable failures and compromises, and lack of visibility and control. These foundations must also address enterprise needs for rapid development and evolution, and system interoperability to achieve mission goals. The objective of Flow-Service-Quality (FSQ) engineering is to develop theory-based engineering methods for network-centric system analysis, specification, design, verification, implementation, and operation. The focus of FSQ is on developing high-assurance systems, with special emphasis on complexity reduction and survivability improvement.

Research Approach

Initial research has identified three integrated engineering concepts that address the realities of network-centric systems:

1. **Flow Structures:** User task flows and their refinements into system service uses can provide engineering foundations for analysis, specification, design, verification, and implementation of system functionality and quality attributes.
2. **Computational Quality Attributes:** Quality attributes can be associated with both flows and the system services they invoke and computed as dynamic functional properties, rather than treated as static, *a priori* assessments of limited value in system operations.
3. **Flow Management Architectures:** Flow structures and computational quality attributes support architecture frameworks that manage flows, network services, and quality attributes in execution.

Flow Structures. Flow structures are compositions of system services distributed across networks that combine to carry out user tasks that accomplish enterprise missions. They employ mathematical semantics that permit human understanding and analysis, despite the underlying asynchronism of network behavior. Flow structure engineering requires designing for unpredictable events that can impact mission survivability. In addition, flow structures provide a vehicle for specification and management of quality attributes such as security and reliability. Thus, the first-class concepts of flow, service, and quality are the primary artifacts of FSQ engineering [1,2,3].

Network-centric systems are usefully viewed as webs of asynchronously communicating components that provide services whose functions can be combined in various patterns to satisfy enterprise mission requirements. System services include all the functional capabilities of a system, from protocols, operating systems, and middleware, to databases and applications. The sequencing of operations in user task flows can be refined into compositions of network hardware, software, and human components that provide the services. These compositions are end-to-end traces that define slices of network architectures whose net effect is to carry out operations that satisfy user requirements.

The mathematical semantics of flow structures are defined to support development and verification for the uncertain environments of large-scale networked systems as a standard engineering practice. Flow structures are essentially procedures that define compositions of network service uses at levels of abstraction ranging from an enterprise mission down to its network implementation. Flows can specify integration and traversal of many systems and components. They can be expressed in simple control structures and refined, abstracted, and verified with precision. Flows invoke services, which can be refined into flows, and so forth, in a recursive process that employs identical methods at all levels of design. The functional specification of a network system is envisioned as a set of flow structures, where the union of the flows defines a necessary network architecture for further optimization, and the functional specification of each service in the network is based on the union of all its uses in flows where it appears.

Computational Quality Attributes. FSQ engineering treats quality attributes as ever-changing functions that must be dynamically computed. Attributes must be measurable in defined metrics as computable functions. While such functions rely on what can be computed and may differ thereby from traditional methods, they permit new approaches to attribute analysis and evaluation. Attribute requirements can be associated with system component uses embedded within flow structures and dynamically compared with computed attribute capabilities in operation.

Flow Management Architectures. Flow structures and computational quality attributes support system architectures that carry out dynamic flow and attribute management in execution. Flow management architectures (FMA) can provide design and implementation frameworks for this purpose. An open family of such frameworks can be defined for architecture development both in the small and in the large.

Expected Benefits

FSQ foundations prescribe engineering practices and tools for network-centric system analysis and development. In particular, the deterministic nature of flow structures facilitates human understanding. Computational quality attributes permit automated reactions to dynamically changing quality values in system execution. In addition, flow management architectures provide systematic frameworks for managing flows and quality attributes in operation.

2007 Accomplishments

Work continued on relating FSQ engineering to web services and service-oriented architectures.

2008 Plans

STAR*Lab is interested in continued development and application of FSQ engineering for large-scale networked systems. Interested organizations are invited to participate in creation of a proof-of-concept prototype and associated engineering practices.

References

[1] Hevner, A., Linger, R., Sobel, A., & Walton, G. "The Flow-Service-Quality Framework: Unified Engineering for Large-Scale, Adaptive Systems." *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS35).* Waikoloa, HI, Jan. 7-10, 2002. Los Alamitos, CA: IEEE Computer Society Press, 2002.

[2] Linger, R., Pleszkoch, M., Walton, G., & Hevner, A. *Flow-Service-Quality (FSQ) Engineering: Foundations for Network System Analysis and Development* (CMU/SEI-2002-TN-019). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002. http://www.sei.cmu.edu/publications/documents/02.reports/02tn019.html

[3] Hevner, A., Linger, R., Pleszkoch, M., & Walton, G. "Flow-Service-Quality (FSQ) Engineering for the Specification of Complex Systems." *Practical Foundations of Business System Specifications.* Edited by H. Kilov & K. Baclawski. Dordrecht, NL: Klewer Academic Publishers, 2003.

Survivability Analysis Framework

Robert Ellison
412-268-7705

Carol Woody
412-268-9137

Survivability Analysis Framework (SAF)

Problem Addressed

Large systems and particularly systems of systems raise the importance of complexity management. The complexity is an aggregate of technology, scale, scope, and operational and organizational issues. While small system security may have been implemented by a set of point solutions that mitigated specific threats, the mitigation of threats of the magnitude and diversity of those associated with large distributed systems of systems (SoS) requires foundational support.

Separation of concerns is a powerful tactic for managing complexity during design and development. A software architecture may try to maintain separation among security, performance, reliability, and other system quality attributes. However, it is the visibility of these qualities within the operational context as the technology is used to address an organizational need that is of most interest. We frequently have maintained separation among system operations, systems development, and business operations, but that separation was often reflected by the expression "toss it over the wall." This approach worked well as long as all requirements could be effectively established in advance and evaluated prior to implementation. Business integration requirements and the appearance of technologies such as web services to support that integration for distributed systems challenge these traditional separations. Even organizations with well-established processes are finding the complexity overwhelming. A vast range of legacy technology and processes are being hooked together through bridges of software and people without a thorough consideration of how these connections function under stress and failure. Development is primarily looking at the individual pieces of new functionality, operations is focusing on the infrastructure, and the gray area of business process connectivity is largely ignored, thereby exposing organizations to increased risk of operational failure.

We define survivability as the capability of a system to fulfill its mission, in a timely manner, in the presence of attacks, failures, or accidents. Survivability concentrates initially on the availability aspects of security but also incorporates confidentiality, integrity, and reliability considerations. Availability must be focused on the specific functions and services needed to satisfy a specific organizational mission, which is increasingly dependent on multiple systems. Survivability concentrates on the organizational activity supported by software systems rather than on the individual systems.

This research initially focused on developing assurance analysis methods that are applicable to systems of systems to address the challenge of increased demands for interoperability, integration, and survivability. Having shown the value of the mission focus for analyzing organizational and technology dependencies, this research effort has expanded to address the need for analytical capability of services such as components of a service-oriented architecture (SOA) and the integration of these shared services with organizational mission. In addition, the consideration of quality assurance and exploration of ways in which an integrated view of mission and technology can support the development of a quality assurance case are under development.

Research Approach

Essential work processes increasingly span multiple systems that are geographically distributed and independently managed. The individual systems are useful in their own right, addressing a selected subset of organizational needs. The business demands for adaptability and integration result in a mix of systems and work processes that are constantly changing. Development is evolutionary as functions and purposes are added, removed, and modified with experience. Completion of each individual system activity is no longer sufficient to meet organizational needs, and the measures for success must focus on the complete organizational mission, which extends beyond component systems.

Consider Figure 1, where each circle represents a geographically distributed system and the blue and black lines are business processes that use those systems. The right side of the figure expands one of those systems. For a military example, a circle might be a specific Service system, whereas the work process might be joint activity that requires coordination across the Services. The specific Service system receives both joint and Service-specific requests. A joint Service activity would likely generate a sequence of actions similar to the actions generated for a Service-specific request.

We need to take two perspectives in analyzing that diagram: the end-to-end work process and the individual systems. The is-used-by relationship is critical for the system participants. A work process, especially in an SoS environment, could create usage patterns that were not anticipated in the design of a specific system and hence could adversely affect the operation of that system. An individual system may need to take a defensive posture with respect to external requests to protect local resources. In addition, failure of one piece will have an impact on the organizational mission that cannot be evaluated within the context of the individual component.

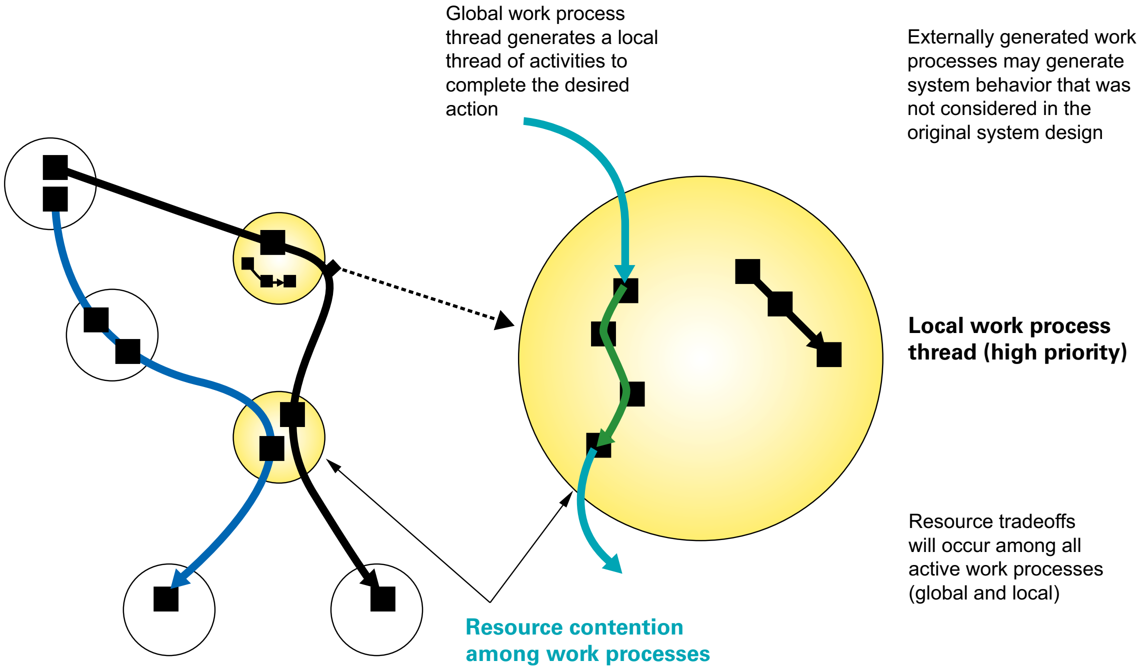

Global work process thread generates a local thread of activities to complete the desired action
Externally generated work processes may generate system behavior that was not considered in the original system design
Local work process thread (high priority)
Resource tradeoffs will occur among all active work processes (global and local)
Resource contention among work processes

Figure 1: System of Systems Resource Contention

The success of the end-to-end work process depends on the successful composition of the individual process steps and an acceptable completion. The key relationship for the work process is depends-on. We would like to assure the end-to-end behavior of a work process, but the interoperability capabilities and failure conditions for each component could drastically affect an acceptable outcome if that step is critical to mission success and internal quality choices do not match mission quality needs. The work process thread will need to be analyzed end to end and step by step to identify gaps that could lead to survivability loss. To do this requires the following detailed process thread information: a description of work process success; expected work process quality attributes such as performance and reliability; and scenarios of both expected and unacceptable behavior, which includes the kinds of things that may go wrong and what will happen should they occur. In addition, each work process to be analyzed must be decomposed into required steps with the following types of information about each step: roles in the process, preconditions, functions, postconditions, constraints, and dependencies. Each step may be composed of multiple components (human, software, system, and/or hardware) acting independently or in a coordinated manner.

Systems and systems of systems can create failure states that are difficult to solve. Historically, system failure analysis has sought to identify a single root cause, but for software-intensive systems that involve human interactions a failure may be the result of multiple software, hardware, or human errors. Each error when considered individually would be perceived as minor. Other failures may arise because of emergent behavior. Each system behaves as specified, but the collective behavior is unacceptable. For example, feedback among systems might generate unexpected resource contention. At this stage, our research considers the stresses that might be induced by a work process thread. We initially focus on the interactions among the systems that participate in that thread and the stresses that might be induced by those interactions on the supporting systems. The stress types include

- Interaction (data): missing, inconsistent, incorrect, unexpected, incomplete, unintelligible, out of date, duplicate
- Resource: insufficient, unavailable, excessive, latency, inappropriate, interrupted
- People: information overload, analysis paralysis, fog of war, distraction (rubbernecking), selective focus (only looking for information for positive reinforcement), diffusion of responsibility, spurious correlations

The scenarios of potential problems, especially those with anticipated high impact, will be used to potentially limit the areas of each stress type to a subset of high interest issues for the mission thread stakeholders. For each type of stress, the analysis framework will be applied to identify what is currently in place, what should be in place, and expected step and/or component behavior should survivability be affected. The analysis framework will be applied at a specific point in time to a selected example mission thread. In order to analyze the change in risk over time, an assessment is needed for the existing work process to establish a baseline of current risk.

Survivability concentrates on what can go wrong. The issues considered by the SAF analysis are shown in Figure 2.

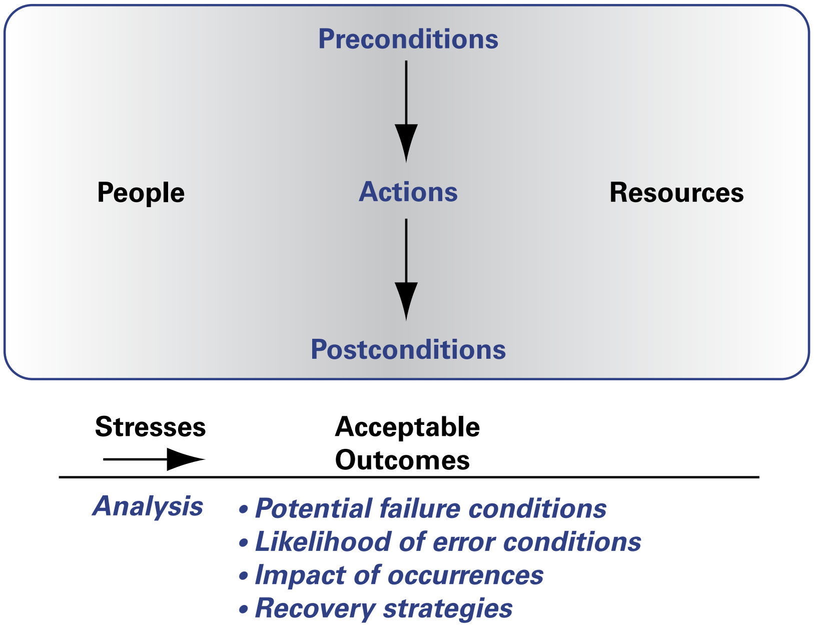


Figure 2: SAF Analysis

Expected Benefits

The expansion of the scope and scale of systems induce new stresses. An objective of the initial phase of this project is to identify indicators of stress that may lead to system failures. Indicators that are appropriate to the early life-cycle phases of software development help to change current practice, whereas software failure analysis typically concentrates only on the errors that are derived from testing. The goal is to generate a sufficient number of examples so that patterns emerge. A pattern, for example, may represent ways to reduce complexity by consolidating risk mitigations.

The Survivability Analysis Framework (SAF), with its emphasis on business process threads, also enables better traceability between technology risks and business work processes. It may also enable better traceability of the design decisions to the requirements of multiple organizational levels.

2007 Accomplishments

The SAF was applied to two additional pilot applications beyond the initial work in 2006. One pilot was within the DoD and the second in a large, non-DoD federal agency. The DoD project considered the challenges of information assurance (IA) across a mission thread, looking at ways to appropriately characterize the impact of IA decisions on the organizational mission. The non-DoD pilot evaluated the impact of technology choices made in development on existing organizational processes for alpha and beta test sites. In addition, SAF concepts were presented to researchers and practitioners at the following conferences: System and Software Technical Conference, Computer Security Institute Conference, International Conference on Commercial Off-the-Shelf (COTS)-Based Software Systems, and the Homeland Security: Research * Innovation * Transition Conference.

An example assurance case for security was developed with support from researchers knowledgeable with safety and reliability assurance.

2008 Plans

A description of SAF and the example security assurance case will be published in a technical note in the second quarter of 2008. The objective for further research is to evaluate ways in which the development of SAF information can contribute to an understanding of assurance for software, systems, and information and influence tradeoff decisions that impact mission quality early in the design and development processes. Pilot engagements will be selected that allow the consideration of organizational and technology options early in the system development life cycle to identify ways to influence the quality tradeoff choices for appropriate consideration of survivability risk and realistic usage.

Additional Research Activities

SEI Customer Relations
412-268-5800

Analyzing Border Gateway Protocol Data for Edge Network Routing Policy Violations

Evan Wright
Markus De Shon

Networks generally play one of two roles relative to other networks they communicate with: transit networks, where they transfer packets *between* others, and edge networks, where they transfer packets between themselves and others. With the exception of ISPs, many enterprises would intend their networks to be edge networks, as they have no advantage to gain by allowing external traffic to cross them.

For edge networks, it is important for the central network configuration authority to control all Internet Access Points (IAPs) on the network. Many edge networks can handle this problem by using a single IAP, but this option is less viable as a network's size and scope increases. Violations can exist where some part of the network purchases its own Internet connection, as traffic that would normally run through the managed access points can then enter and/or leave through the unmanaged gateway, bypassing central security controls. This is complicated by routing protocols themselves; routing implements reliability through redundancy and reallocating resources. Security can therefore be compromised when a well-designed network, performing as specified by the protocols, routes traffic to ensure connectivity at the expense of security.

We examine all network blocks that are allocated to the known Autonomous System (AS) number(s) for the network. Particularly complex networks may have a hierarchical arrangement, where particular AS numbers act as the gateway ASes, so that all traffic terminating at an AS owned by the organization must follow an AS path that includes one of the designated gateway ASes.

In this work, we designed and implemented software that consumes publicly available Routing Information Base (RIB) tables from sources such as Routeviews[1] or the RIPE Routing Information Service[2] (and could also consume privately available routing tables), looking for three possible violations of routing policy:

1. "Network escapee": the unmanaged gateway routes all traffic from the violating subnetwork, bypassing all central security controls. In this case, a netblock that is allocated to the edge network is advertised as part of an AS that does not belong to the organization (usually some other ISP AS).
2. "Everyone's ISP": the unmanaged gateway is advertising a route for some subset of the Internet, such that the edge network becomes a transit network for traffic destined for that advertised address space. In this case, a network block that does *not* belong to the organization is present in an AS path where at least one AS belongs to the organization, but the endpoint AS does not.
3. "Multi-homing": the unmanaged gateway advertises address space for part of the address space on the edge network, which is also advertised through authorized gateways. In this case, traffic for that subnet could enter or leave through either gateway, in some cases bypassing central security controls. In this case, a network block that belongs to the organization is advertised through AS paths that terminate in the organization through an expected gateway AS and *also* terminate in a path that does not pass through one of the gateway ASes.

The results of this analysis provide useful insight to network administrators regarding unmanaged gateways on their networks and the risks posed by those gateways. In the next year, we expect further development of the tools that enable this analysis and useful presentation of the results they produce. Such development should occur in collaboration with specific networks interested in such analysis, including guidance as to how to incorporate internal topology information in assessment of the risks of unmanaged gateways.

[1] http://www.routeviews.org/
[2] http://www.ripe.net/ris/docs/beacon.html

Anomaly Detection Approaches for Network Time Series

Rhiannon Weaver

In anomaly detection systems designed around network traffic data collected as a time series of counts, we would like to employ statistical modeling techniques to detect outlying behavior. However, network traffic has characteristics that make traditional time series analysis difficult:

1. **seasonality**: Traffic patterns have repeatable elements across many different time scales (days, weeks, months, etc.), that include complicated patterns.
2. **non-normality**: Traffic is often subject to many large spikes, as well as increasing variability with increasing size.
3. **heterogeneity**: Typical behavior can comprise both smooth action and abrupt jumps or jitters, with variability depending on the time of day or week.

We take a hierarchical approach to modeling these series, in which we look for patterns of anomalous activity on two different scales: a smooth trend scale that includes most of the model's long-term seasonal patterns for detecting slowly realized anomalies, and a stationary drift or noise scale for detecting short-term anomalies that appear as rapid spikes in recent activity.

In the first step of this research initiative, we have explored methods that focus on removing the large-scale, smooth trend in the time series data and on processing the remaining residual scale so that standard statistical thresholds can be used to detect spikes of activity relative to the recent past. Two different approaches have been explored:

1. Using "moving splines" to remove trends from a single time series. Once long-term trends and seasonal effects have been removed, we scale the one-lag difference in the noise according to the standard deviation calculated from the recent history. We then use a simple threshold to detect whether a spike is large enough to be considered anomalous. Figure 1 shows an example of this "spike detector" run on one week's worth of packet counts at one-minute intervals.

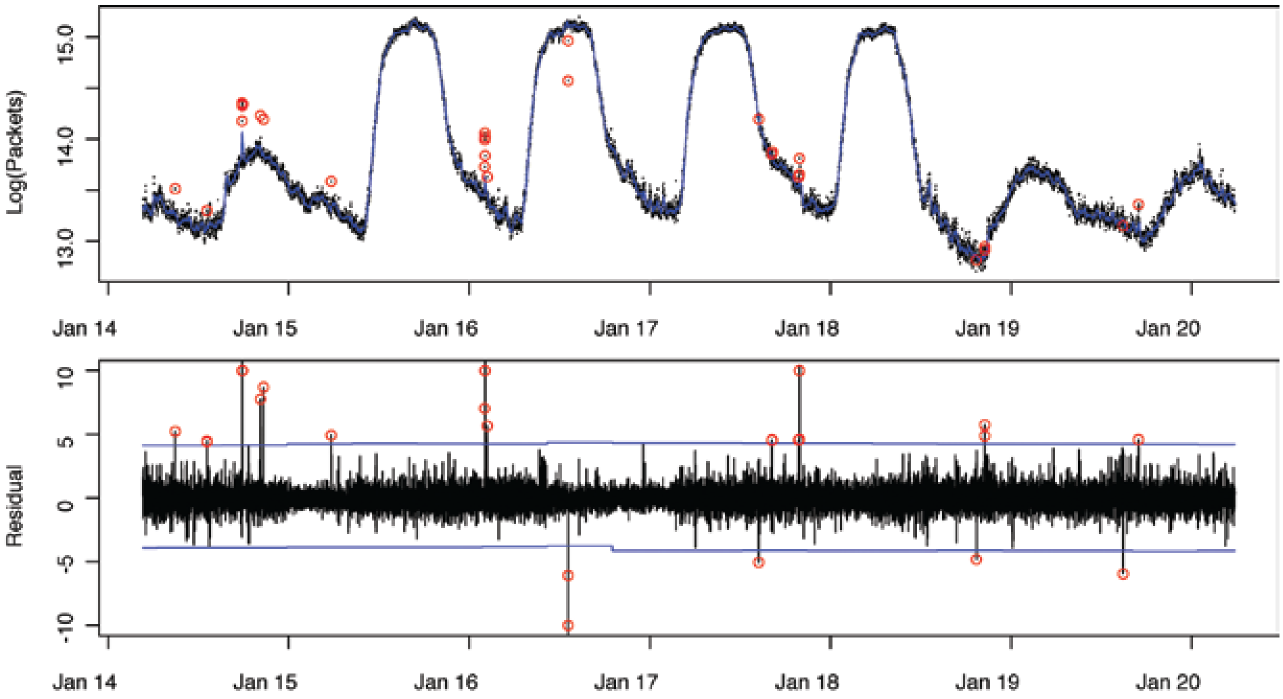


Figure 1: Results of a "Spike Detector" Using Moving Splines to Remove Trend

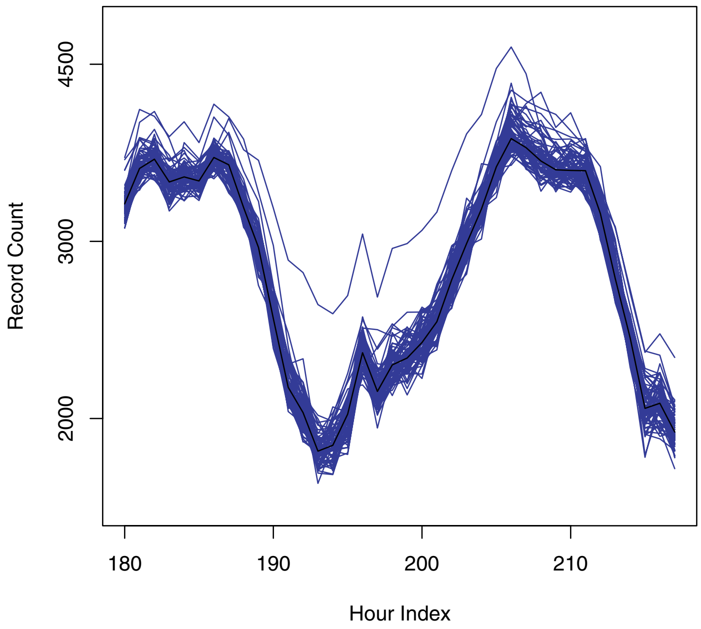


Figure 2: An 18-Hour Port Surge Detected Using Clustering

2. Using clustering based on correlation to package many similar series together, and modeling trends with the cluster median. This approach shows promise for applications such as detecting surges in ephemeral port activity due to peer-to-peer activity. Figure 2 shows an example of an 18-hour port surge that was detected in one ephemeral port using a cluster of 72 of its nearby neighbors. Further examination of flow data suggested that the port was being used for a peer-to-peer application.

The next step in the analysis is to develop methods to study and predict patterns in the smooth trends. In the coming year, we plan to examine traditional time series methods, such as state space models, applied to low-dimensional parameterizations of these smooth curves in time.

We believe that a principled statistical approach to these problems can help improve anomaly detection techniques in both theory and application. Statistical theory has much to offer beyond the traditional "mean +/- 3 standard deviations" approach, including robust analysis, diagnostics for complicated modeling scenarios, and methods for controlling false positives on a large scale. The methods we have explored can be used to detect surges in activity such as distributed denial-of-service (DDoS) attacks, initiation of peer-to-peer traffic, and flash crowds, as well as drops in activity, for example due to sensor failures or subnet loss.

Assessing Enterprise Security

Bradford J. Willke

When senior managers ask "Are we secure?" they are not just looking for how long it has been since the last security event. They are seeking a calculated answer involving risks avoided or mitigated, processes measured and analyzed, and benchmarks reflected against peers. For certain, they seek a more simplified answer than the litany of best practices deployed by the security team, but a more comprehensive answer than "Yes." The problem is not just how to answer the question, but when to answer it. It is not practical to freeze the enterprise to take account of how effective security countermeasures are in protecting business services and assets. Still, managers need accurate assessments and evaluations that focus on sound risk management principles and at the same time survive their changing environment.

Structured information security needs analyses are one answer to the problem space. Our research into needs analysis techniques shows that the basis for assessing a changing environment must be an understanding of how well versed the enterprise is in recognizing mission-assurance properties. Business imperatives are accounted for within all aspects of the security management process. The rate of change in security mechanisms, policy, and human resources is much greater than the frequency or changes related to inputs of good decision making: risk evaluation criteria, enterprise security requirements, and critical success factors. Therefore, the viability of security is not measured by the collection of practices or the technology alone or even within a system or network context, but as a function of persistent management behaviors over time. The focus on these decision-making factors is a far better gauge of how stable security is and a far more predictive indicator of future performance. As an efficient answer to the senior management question, this evaluative process samples the practice of security management to provide a better answer, but to a better question. That question is, "How able are we to manage security?"

Comparison of Scan Detection Methods Using Expert Evaluation

Timothy J. Shimeall
Rhiannon Weaver

Many organizations have an interest in detecting outsiders scanning their networks as a means of assessing network security and predicting potential activity by intruders. Most organizations have developed their own methods for this assessment, which makes comparison of results difficult. Many of the methods have never had a systematic evaluation of their effectiveness.

The CERT Network Situational Awareness group has had a long-running interest in efficient and effective detection of network scanning using network flow information. Previous work[1] developed a scan detection algorithm (MISSILE), studied it in conjunction with other algorithms (Threshold Random Walk (TRW) and SYN flow counting), and implemented a scan detection tool, rwscan, combining MISSILE and TRW as part of the SiLK analysis tool suite.[2] This work examined the effectiveness of rwscan in contrast to a rule-based detection scheme. Of particular interest was estimation of two relative attributes of the methods in comparison to one another:

- False Positive Rate: count per hour of sources classified as scanning when these sources were not actually scanning
- False Negative Rate: count per hour of sources not classified as scanning when these sources were actually scanning

The rule-based detection method was developed based on analysis of the nmap scanning tool and on descriptions of ICMP-based scanning methods. The approach was to count several types of scan-related flows:

- TCP flows with only the SYN flag set during the flow
- TCP flows with only the SYN and ACK flags set during the flow
- TCP flows with only the RST flag set during the flow
- TCP flows with only the RST and ACK flags set during the flow
- ICMP ECHO flows
- ICMP ECHO-Response flows
- ICMP Timestamp Request flows
- ICMP Information Request flows
- ICMP Address Mask Request flows

If the sum of these counts for a given source was ten or greater, the rule-based method classified it as a scanner. The two methods (rwscan and rule-based) were run over two weeks of flow data collected from a large computer network. Figure 1 shows the counts per hour of flows identified as scans during those two weeks. The "Neither" curve identifies flows that were designated as scans by neither method.

One major difficulty in estimating false positive and false negative rates is the lack of ground truth for detected flows. To overcome this, experts were used to examine the flow data and provide judgment on when the flows represented a scan or not. Four experts were used in this study, each with documented expertise in network security and over a year of experience in network flow analysis for security.

The experts were each given a set of flows to examine. The set of flows was selected to evaluate the effectiveness of the methods by stratifying the set of flows per hour per source on the basis of three factors: which method classified the flows as scans (rwscan only, or rule-based only), hourly average byte volume in the flows (1-60 bytes, 61-999 bytes, 1000+ bytes), and persistence of classification of the source as a scanner (<14 hours, 14+ hours). This yielded 12 strata, one of which was empty (there were no large-volume persistent sources classified by rwscan). Equal size samples of sources were taken from each strata, plus an unstratified sample of sources classified by both methods.

One hour of data (when the source was classified as a scanner) was randomly chosen for each source. A subset of the sample was randomly chosen to be evaluated by all experts, and further subsets were randomly chosen to be evaluated by each pair of experts. Finally, the remaining sources were divided evenly to be examined by experts individually. The experts were not aware of how their set of flows was constructed while they were performing their evaluation, nor did they work in consultation with one another. However, follow-on analysis of their results indicated agreement among the experts 85% of the time.

The results of the analysis showed that when rwscan classified a source as a scanner and rule-based did not, 93% of the time the experts classified the source as a scanner (±3%). When rule-based classified a source as a scanner and rwscan did not, 24% of the time the experts classified the source as a scanner (±6%). The rwscan method missed sources when they used RST or RST-ACK packets for scanning. The rule-based method missed sources when their scans were replied to by the target. These results offer opportunity for further improvements in rwscan, while also providing improved confidence in this method of scan detection.

[1] http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tr005.pdf

[2] http://tools.netsa.cert.org/silk/

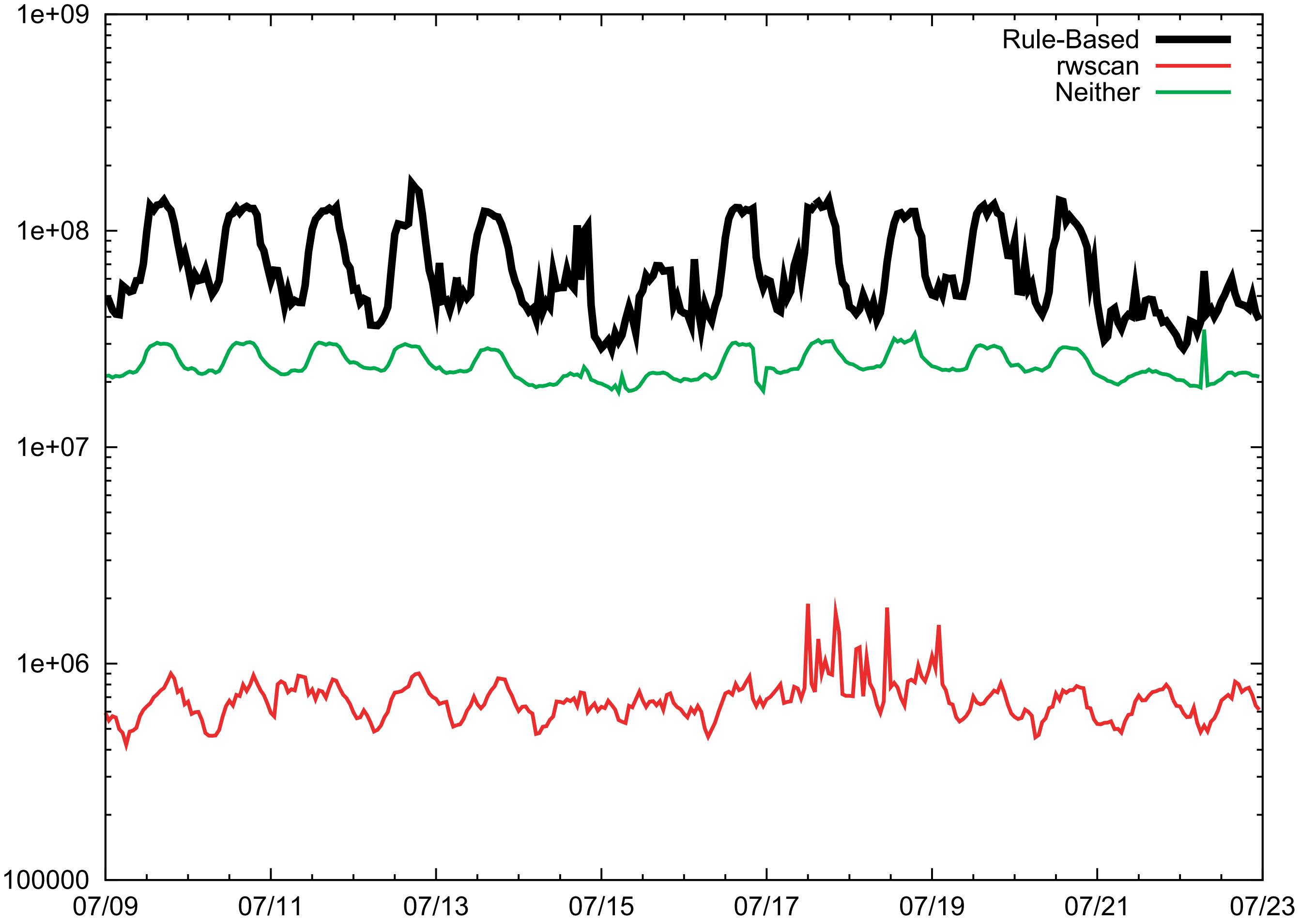


Figure 1: Classification of Flows as Scans by Method, Logarithmic Scale

Control System Security and Critical Infrastructure Survivability

Howard Lipson

The complex control systems and devices that automate operation of society's critical infrastructures, including the electric power grid, transportation, oil and natural gas, chemical plants, and manufacturing and industrial processes, are increasingly Internet-connected. While this connectivity brings benefits, it is accompanied by growing risks of targeted cyber attacks that could result in economic and societal consequences of substantial proportions. These attacks are literally enabled by common software vulnerabilities and the inherent susceptibility of networked systems to malicious access.

For example, the Internet and its enabling technologies are playing an increasing role in electric power systems, improving the efficiency and sophistication of business and technical operations. However, Internet connectivity also introduces significant new risks to the power grid's automated control systems and hence to the power grid itself. Moreover, the scientific and engineering foundations for secure and survivable system design must be substantially extended to address the scope and complexity of the sophisticated forms of control envisioned for next-generation energy systems. Equally important is the need for technology guidelines to ensure that engineering best practices are consistently employed during system design, development, and evolution.

CERT plans to significantly expand its operational and research focus on control system security engineering and critical infrastructure survivability in 2008. A primary goal is to use lessons learned from incident and vulnerability analysis to contribute to the creation and enhancement of software engineering methodologies and best practices for secure and survivable control system development and operation. Another major objective is to raise vendor and end-user awareness of engineering best practices for secure and survivable control system development and operation. Along these lines, CERT will be initiating a new content area on *Secure Control System Software* as part of the Department of Homeland Security Build Security In (BSI) website.[1]

In 2007, CERT provided a Capitol Hill briefing for Congressional staff at the invitation of the Center for Science, Technology and Security Policy (CSTSP) at the American Association for the Advancement of Science (AAAS). The briefing, "Cyber Security: Protecting Our Networks and Critical Infrastructure – Technical and Policy Challenges," was hosted by the U.S. House of Representatives Committee on Homeland Security.[2] CERT also participated in a National Institute of Standards and Technology (NIST) Workshop on Applying NIST Special Publication (SP) 800-53 (an IT cyber security standard) to Industrial Control Systems.[3] A member of the CERT staff also serves as an adjunct research faculty member at the Carnegie Mellon Electricity Industry Center.

[1] https://buildsecurityin.us-cert.gov/daisy/bsi/901.html

[2] http://www.aaas.org/news/releases/2007/0608cybersec.shtml

[3] http://csrc.nist.gov/groups/SMA/fisma/ics/documents/presentations/Knoxville/NIST-Knoxville-workshop-Aug2007.html

Developing National Risk Management Capabilities

Bradford J. Willke

Convergence of the cyber and physical worlds has brought great advances in the productivity, service, and control of business, government, and consumer activities. In the wake of these converged infrastructures, many nations have realized that these advances are both an advantage to public and private interests and a potential new avenue for threats and attack. At a notional level, governments understand there is a relationship between the threats, vulnerabilities (some as a direct result of the convergence), and capabilities of attackers to wage asymmetric attacks and the negative impacts that would result to national economies and public safety and confidence. Not knowing precisely what the national-level resolution needs to be, nations at least ask the appropriate questions of "How much of a problem do we have?" and "How much security do we need given the problem?" One related and important question is often asked in parallel: "How do we know the nation is making progress?"

The answer to the problem space and all of these questions is, in part, national risk management and the measurement of policy, technology, and physical security countermeasures against the risks identified. CERT's work in enterprise information security risk assessment and management, beginning formally with OCTAVE in 1999, is providing a basis for identification and improvement of national risk management capability and definition. This research involves exploration of adaptable measures such as critical information infrastructure best practices and standards, threat identification and predictability, vulnerability evaluation, and national risk evaluation criteria. The research also involves dependency analysis among critical infrastructure sectors through asset, threat, and vulnerability identification and analysis. Finally, there is a benefit to identifying scalable solutions to national risk, not only in systems and software assurance, technical infrastructure management, and security operations, but also across non-traditional domains, such as governance, policy, legal, and cultural imperatives.

Development of the CERT Forensic Toolkit

Matthew Geiger
Cal Waits

Capitalizing on our strong relationship with federal law enforcement and security agencies, CERT's Forensics team has worked with these partners to identify gap areas in existing forensic analysis tools and techniques, as well as to identify new challenges arising from emerging technologies. This process is informed by discussion with practitioners, observation of current methodologies and tools, and by our experience in support of operational missions.

In 2007, as a result of these collaborations, we built a collection of unique tools and deployed them to meet the needs of investigators and forensic analysts. Our principal method to distribute the forensic toolkit, together with training and briefings on foundational research, is through the Virtual Training Environment (VTE), a CERT website created to quickly and effectively share knowledge, training, and software.[1] This mechanism has enabled us to distribute the toolkit to dozens of investigative agency partners, both in the U.S. and abroad.

Current components of the toolkit include the following:

Live View LE

This is the law enforcement version of the Live View tool,[2] which enables forensic investigators to take a physical device or an "image" file of a disk or partition and transform it into a VMware virtual machine. Among the benefits is the ability to interact with bespoke or proprietary software on the target system in a controlled environment. Both Windows and Linux systems are supported. Features added in 2007 include support for disk images stored in the EnCase forensic format. The LE version of the tool offers the ability to remove account passwords and dump cryptographic credential stores to facilitate offline analysis.

CryptHunter

Designed to address the challenge presented by the growing prevalence of disk and volume encryption, CryptHunter is a pre-acquisition screening utility that will detect mounted, encrypted volumes as well as whole disk encryption on running systems. This enables investigators to modify their data acquisition procedures to preserve decrypted data while it remains accessible. New detection algorithms, several new signatures, and support for Microsoft Vista's BitLocker encryption scheme were added in 2007.

[1] https://www.vte.cert.org/vteweb/
[2] http://liveview.sourceforge.net/

Aperio

Our research helps analysts establish whether counter-forensic tools were used on target file systems and, if so, identify which of a range of commercial software packages were employed. The research also identifies a number of often significant operational flaws in counter-forensic software, which can enable examiners to recover probative data overlooked by the software. Aperio, which is bundled with an analyst's reference library based on this research, automates the process of detecting file system fingerprints and matching them to a particular counter-forensic software package.

In 2008, aside from extending and maintaining the existing tool set, research will focus on new tools, including the following:

Mnemo

Mnemo is a framework to permit investigators to access and copy main memory contents directly via the PCI bus, enabling data seizure in situations where interaction with the OS is blocked or undesirable. The use of a functional prototype to recover cryptographic material has been demonstrated in operational deployment with a federal law enforcement agency partner.

CCFinder

CCFinder is a suite of utilities designed to facilitate the discovery, organization, and querying of financial data and related personally identifiable information in large-scale investigations. Development is a direct result of the operational requirement of a federal law enforcement partner and has additionally streamlined the process of victim notification.

We are developing forensic analysis and recovery techniques for storage devices based on flash-memory, from those used in mobile phones and personal digital assistants (PDAs) to the new generation of solid-state storage for ultra-portable laptops. The research is focused on new techniques for low-level data acquisition and for recovery and reconstruction of data blocks reallocated by the "wear-leveling" algorithm employed in the flash medium. This, in turn, will provide a fuller picture of the "history" of the superimposed file system.

References

Boileau, A. "Hit By A Bus: Physical Access Attacks with Firewire." RUXCON 2006. http://www.ruxcon.org.au/2006-presentations.shtml#14

Geiger, M. "Evaluating Commercial Counter-Forensic Software." DFWS 2005. http://dfrws.org/2005/proceedings/geiger_couterforensics.pdf

van der Knijff, R. "10 Good Reasons Why You Should Shift Focus to Small Scale Digital Device Forensics." DFRWS 2007. http://dfrws.org/2007/proceedings/vanderknijff_pres.pdf

Leaking Private IP Information: AS112 DNS Traffic Analysis

Sidney Faber

A well-secured network will keep all information about its internal topology within the network. However, many networks leak private information out to the Internet through reverse DNS lookups and update attempts. The large volume of these requests has led to the creation of a specific any-cast autonomous system, AS112, specifically engineered to reduce the load on the root servers, consuming these queries close to their originator [1].

On January 9 and 10, 2007, a number of root and top-level DNS name server operators collected full packet traffic captures as part of the "Day in the Life of the Internet" (DITL) exercise [2]. These packet captures included data from two AS112 DNS servers, presenting an opportunity to analyze data exfiltration related to private network topology.

Analysis of the AS112 traffic revealed a wealth of private network topology information. The captured data falls into the following categories of DNS traffic:

- PTR Queries: Clients requesting the DNS name associated with a private IP address.
- SOA Queries: Clients locating the authoritative name server that manages reverse lookup entries (PTR records) for a block of private addresses.
- UDP Update Requests: Clients attempting to update the reverse lookup entry (PTR record) for a private address. An SOA query often precedes the UDP update request.
- TCP TKEY Attempts: Clients attempting to negotiate a signature that can be used to perform a cryptographically signed update. A failed UDP update usually precedes the TCP TKEY attempt.

When aggregated, AS112 data can expose many features of the susceptible network:

- Private address ranges in use and the associated public gateway addresses appear in PTR and SOA queries.
- Names and private addresses for individual hosts behind public gateway addresses appear in UDP Update Requests.
- Windows workstation and server names and domains appear in TCP TKEY attempts.

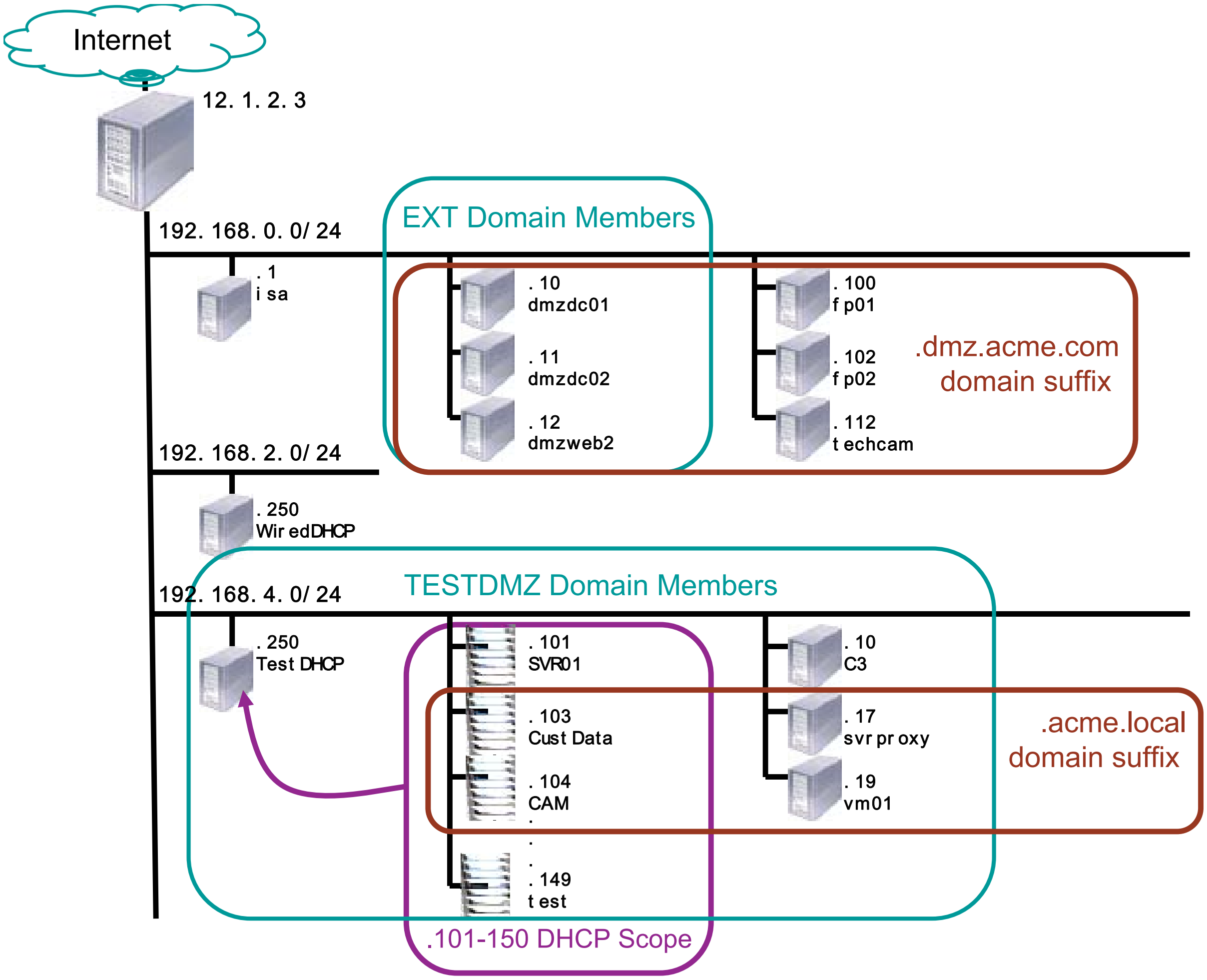


Figure 1: Anonymized Example of a Network Diagram Obtained by AS112 Data Analysis

Name and location of dynamic addressing (DHCP) servers become evident when observing UDP and TCP TKEY attempts generated by individual hosts.

Figure 1 presents an anonymized network diagram obtained from a consolidated overview of actual AS112 data for one specific gateway address. Although most networks do not leak enough data to create a detailed topology such as this, over two million hosts presented at least some identifying information in the DITL packet capture data.

Most network administrators would prefer to keep AS112 data contained within their internal network, which can be easily accomplished with simple changes to the network's DNS configuration. By configuring DNS servers on the internal network as authoritative for all private reverse lookup address zones, all PTR requests and updates for private address space are kept within the local network. An IETF Best Current Practice, "Locally-served DNS Zones," currently in draft format [3], proposes the steps necessary to contain this traffic.

Since the "Day in the Life of the Internet" exercise is a recurring exercise, this analysis suggests that organizations may be able to use information from the exercise to periodically re-assess their network configurations and to get quantitative results on possible leakage of private topology information. One natural outgrowth of the study presented in this paper is the development of automated analysis scripts and tools that would facilitate that assessment. Collaboration with network administrators interested in that assessment is essential in proper development of the analysis scripts and tools.

References

[1] AS112 Project. http://public.as112.net (2008).

[2] Cooperative Association for Internet Data Analysis (CAIDA). "A Day in the Life of the Internet." http://www.caida.org/projects/ditl/ (2008).

[3] Andrews, M. "Locally-served DNS Zones." http://tools.ietf.org/html/draft-ietf-dnsop-default-local-zones-02 (June 8, 2007).

Listening to Whispers in the Hurricane: Eliminating Attack Noise from Attack Detection Systems

M. Patrick Collins

Anomaly detection systems generally require a period of training time in order to develop a model of network activity. Since these systems generally operate by identifying deviations from the model caused by attacks, it is critical that attacks not dominate the training period. However, several common protocols, such as SSH, are attacked consistently and regularly.

The focus of this work is on identifying and eliminating noise from common attacks (in particular, scans and failed takeover attempts) in order to produce a model that can then identify *subtle* attacks. To do so, we use a methodology we term *attack reduction*. Attack reduction assumes that the majority of attacks are failures and that the parties that conduct most attacks have little chance of successfully compromising network security. An example of this type of traffic are the scans propagated by worms and other automated software. The success rate for these scans is very low relative to the number of hosts scanned, and many of the addresses contacted by a randomly scanning worm do not even have a host located there.

Attack reduction eliminates this noise in a two-stage process. The first stage, which we call *log-level filtering*, identifies outliers by examining individual records and discarding records that represent failed connections and other interactions that were likely rejected by the targeted host's TCP stack. Log-level filtering does not eliminate all evidence of attacks, but it does reduce the amount of data that has to be examined by more sophisticated processes.

The results of the log-level filtering are then examined by a *state-level filter*, which uses statistical tests to determine whether or not outliers are present in the remaining data. State-level filtering assumes that in the absence of an attack, system state can be modeled using a standard statistical distribution, such as the normal distribution.

By combining these two filtering techniques, we are able to produce results that are orders of magnitude more accurate than when training on raw data alone. The resulting anomaly detection system can then detect subtle attacks and, more importantly, gauge the impact of a particular attack on the network it observes.

Managing IP Address Data Using IPA

Anthony Cebzanov

Network analysts often require a repository of IP addresses and data associated with them. Initially, this repository may be as simple as a list of suspicious addresses or network blocks, but as networks increase in size and complexity, it becomes valuable to have a central repository for storage and retrieval of more complex data about IP addresses on a network. To address these requirements, we have developed the IP Association repository, or IPA.

Within IPA, data is organized into catalogs, which are logical partitions of IP associations. Each catalog is assigned a catalog type, which determines the structure of the data stored within that catalog. The simplest type is the set catalog, which stores collections of distinct IP ranges. Set catalogs are useful when no additional data about IP ranges is necessary other than the name of the group to which it belongs. The remaining IPA catalog types extend sets with additional information about each IP range in the set. IPA defines the structure of the data for each catalog type, leaving the semantics up to each application.

To facilitate analysis of historical as well as current data, records imported into IPA are grouped into datasets, which are collections of IP associations within each catalog that are valid only for a specified time interval. This allows retrospective queries to use host inventories that were valid at the time the traffic was captured, rather than at the time the query is executed.

In addition to the IPA repository, we provide the libipa client library for applications that wish to import data from and export data to an IPA repository. Using this library, IPA support was added to the System for Internet-Level Knowledge (SiLK) tool suite, which many researchers and network administrators use for network flow analysis.

In 2007, we deployed IPA at a large client site with a catalog of the client's internal IP address allocations. Another deployment is scheduled for early 2008, with a focus on providing server inventory data. IPA is released under an open source license and is available at http://tools.netsa.cert.org/ipa/.

Standardizing Incident Reporting with IODEF

Roman Danyliw

Effective response to computer security incidents by computer security incident response teams (CSIRTs) or security operations centers (SOCs) is predicated on timely communication with their constituency, parties involved in the incident, and other coordinating CSIRTs or SOCs. Furthermore, the broader dissemination of this incident information to watch-and-warning organizations improves situational awareness of possible threat.

Significant policy challenges exist in exchanging this class of information. These challenges are further compounded by the diversity of instrumentation to detect incidents and the varied level of expertise of staff conducting analysis on them. Nevertheless, incident data is being successfully exchanged by cooperating parties, and further adoption of these successes has been inhibited by a lack of a standard data format for computer security incident data.

The Extended Incident Handling (INCH) Working Group at the Internet Engineering Task Force (IETF) was chartered to define a transport format and protocol to encode information commonly exchanged between CSIRTs and their constituency and the data shared between CSIRTs. The work was scoped to only define information relevant across administrative domain and ignored considerations of internal workflow.

The data model for computer security incidents, the Incident Object Description Exchange Format (IODEF),[1] was co-authored by CERT and was specified with an XML schema. It supports the commonly exchanged data in an incident report and provides a well-defined means to extend the data model for domain-specific information not included in the base specification. The data model includes free-form text and enumerated values, and it supports internationalization. Since organizations may define incidents in different ways, consideration was given to providing flexibility in the data elements and no overarching taxonomy is enforced.

The IODEF provides operational and research communities a standardized means to specify incident data in a format that is machine readable and well specified.

[1] Danyliw, R., Meijer, J., & Demchenko, Y. "The Incident Object Description Exchange Format." RFC 5070, November 2007.

Using Protocol Graphs to Identify Hit-List Attackers

M. Patrick Collins

A protocol graph is a representation of entities communicating by means of a single protocol over a limited period. We expect that these graphs will have structured behavior because these protocols themselves have structure. For example, protocols that rely on password or key authentication, such as SSH, FTP, and Oracle's TNS protocol, will naturally be divided into multiple communities of authentication. These groups will exhibit little mutual overlap, as the authentication process will limit the number of parties with access to one of these servers. Conversely, we expect that non-authenticated protocols such as HTTP, DNS, and SMTP will have a much larger community of users, and will naturally form large clusters.

The impetus for this work comes from the broader discipline of social network analysis (SNA). Social networks have been used to describe communities and critical individuals in organizations as diverse as academic citations and criminal conspiracies. We use protocol graphs as a form of social network: the various servers and clients comprising protocol graphs have well-defined reasons for communicating with each other, and we consequently expect that by applying SNA metrics to these graphs, we can identify specific behaviors in the networks and use these methods for anomaly detection.

To test this approach, our initial work has focused on identifying hit-list attackers. These are attackers who, instead of blindly scanning the network, begin with advance knowledge of their targets and then exclusively attack them. In comparison to blind scanning, hit-list attacks have very low failure rates. That is, whereas normal scanning detection techniques can rely on failed connections, alerts such as ICMP "host not found" messages, or communications with nonexistent hosts, hit-list attacks will communicate successfully with their targets each and every time they choose to. Because of their high success rate, scan and attack detection methods that rely on an attacker failing to communicate will miss these attacks.

To test our method, we examine the behavior of two metrics over four protocols. Using 30s and 60s, we examine the traffic from HTTP, Oracle, SMTP, and FTP traffic over the course of two weeks. On these protocols, we examine the behavior of the largest component size and total graph size. We hypothesize that the largest component of the Oracle and FTP graphs will make up a smaller fraction of the total protocol graph than for protocols such as DNS, which do not rely on user authentication.

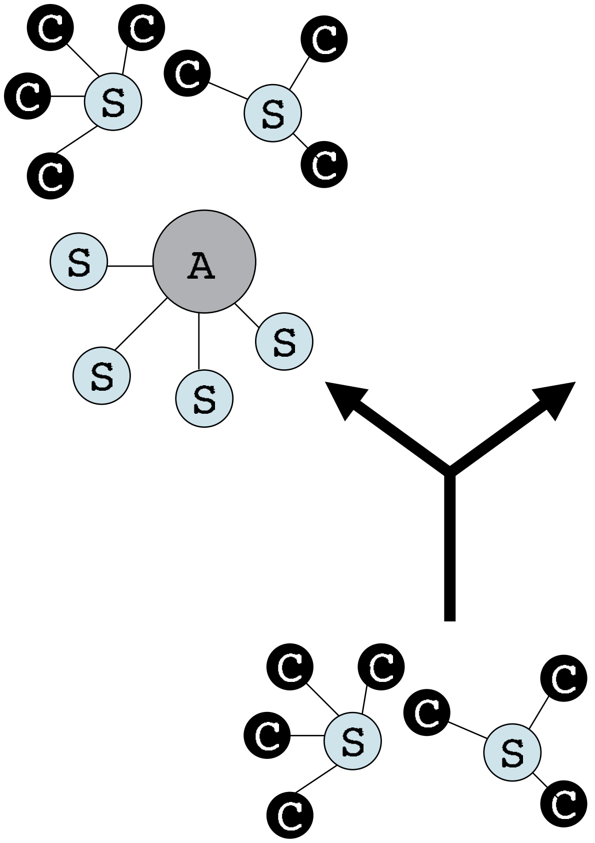


Figure 1: Mutation of a Protocol Graph by Attackers

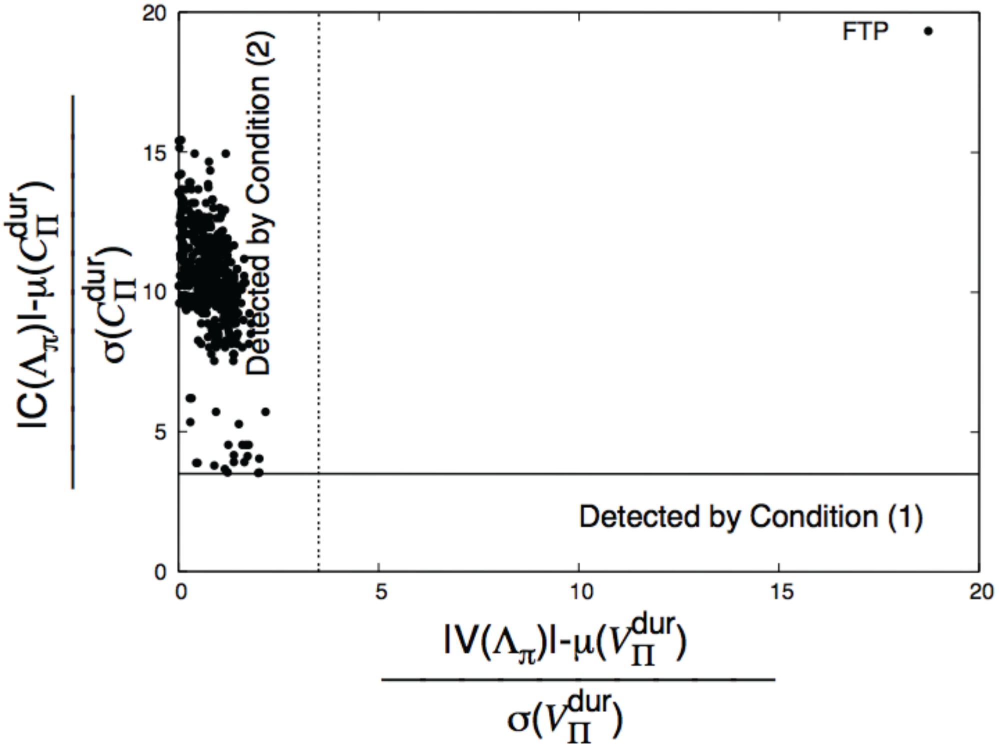


Figure 2: Detection of Graph by Changes in Largest Component Size or Total Graph Size

Depending on the type of attack conducted, we expect that a protocol graph can change in two ways, as indicated by Figure 1. In the left-hand case, the attacker has limited knowledge of the activity on the network and adds additional nodes to the graph by communicating with nonexistent or inactive targets. In the right-hand case, the attacker integrates components that are otherwise discrete by attempting to communicate with multiple disparate communities.

By measuring the largest component and total graph sizes over 30s periods over the course of two weeks, we are able to satisfactorily model these values using normal distribution and then run simulated attacks against existing protocol graphs to measure attacker impact.

An example of the results of these attacks is given in Figure 2. This figure plots the effectiveness of detection by total graph size (Condition (1)) and largest component size (Condition (2)) over FTP data. For these plots, the barrier lines at 3.5 standard deviations indicate when the attack is detected. As the figure indicates, changes in largest component size are a very powerful indicator of hit-list attacks.

In the coming year, we plan to expand our graph-based description mechanisms to include additional attributes and more complex descriptors of graph state. We are finding promising results in other common SNA metrics and intend to implement these systems on live data.

References

Collins, M. P. & Reiter, M. K. "Hit-List Worm Detection and Bot Identification in Large Networks Using Protocol Graphs." *Recent Advances in Intrusion Detection (RAID)*, 10th International Symposium, RAID 2007. Queensland, Australia, Sept. 2007. Springer, Lecture Notes in Computer Science, Vol. 4637, 2007 (ISBN 978-3-540-74319-4).

Identifying Future Research Needs

Archie Andrews, Manager, New Research Initiatives

In the almost 20 years that CERT has existed, it has become obvious that security is a constantly changing challenge. Underlying technologies continuously morph and adapt, unpredicted uses of technology continue to emerge, and threats grow and evolve based on new intrusion targets and opportunities. In this constantly changing environment, the challenge for CERT is to bring the best technical solutions to bear for improving the state of information security and software engineering practice. CERT responds to this challenge by anticipating and addressing problems likely to be faced in the mid-term future by system administrators, system architects, and software engineers, often in collaboration with the information security and software engineering research communities.

To ensure that new ideas and research initiatives continue to drive CERT, we look to the natural evolution of our projects as they work through the research life cycle. Our staff has developed a deep understanding of bodies of knowledge in many security subject areas. We recognize gaps in this knowledge that impact our sponsors and sustain a research and development program aimed at filling the gaps to reduce their impact. This method of extending our research work leverages past successes and knowledge to build toward the future.

We are also looking beyond our current core capability and concentrations to engage external resources to help identify new areas of potential need. We are enlisting the aid of visionaries, thought leaders, and influential practitioners from government, industry, and academia to present their views on trends in technology development, emergent uses of information systems and networks, and the changing nature of vulnerabilities and threats, in order to anticipate security problems and initiate research relevant to future needs. Engaging with individuals who work at the horizons of technology, corporations that are pushing the envelopes of technology application, and government entities that are bringing advanced technology to bear on difficult problems will help CERT gain an appreciation of how the information security picture is likely to change. Our intention is to put in place a continuous, self-supporting process to keep our thinking and research results fresh and relevant to tomorrow's issues.

List of Selected Publications

Talks/Panels/Workshops

Technical Leadership

Biographies

List of Selected Publications

Book Chapters

Mead, N. R. “Identifying Security Requirements Using the Security Quality Requirements Engineering (SQUARE) Method,” 943-963. *Information Security and Ethics: Concepts, Methodologies, Tools, and Applications*. Edited by Hamid Nemati. IGI Global, 2007.

Reports

Cappelli, D. M., Desai, A. G., Moore, A. P., & Trzeciak, R. F. *Management and Education of the Risk of Insider Threat (MERIT): Mitigating the Risk of Sabotage to Employers’ Information, Systems, or Networks* (CMU/SEI-2006-TN-041). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2007. http://www.sei.cmu.edu/publications/documents/06.reports/06tn041.html

Caralli, R., Stevens, J., Wallen, C., White, D., Wilson, W., & Young, L. *Introducing the CERT Resiliency Engineering Framework: Improving the Security and Sustainability Processes* (CMU/SEI-2007-TR-009). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2007. http://www.sei.cmu.edu/publications/documents/07.reports/07tr009.html

Caralli, R., Stevens, J., Wilson, W., & Young, L. *Introducing OCTAVE Allegro: Improving the Information Security Risk Assessment Process* (CMU/SEI-2007-TR-012). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2007. http://www.sei.cmu.edu/publications/documents/07.reports/07tr012.html

Mead, N. R. *How To Compare the Security Quality Requirements Engineering (SQUARE) Method with Other Methods* (CMU/SEI-2007-TN-021). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2007. http://www.sei.cmu.edu/publications/documents/07.reports/07tn021.html

Woody, C. *Process Improvement Should Link to Security: SEPG 2007 Security Track Recap* (CMU/SEI 2007-TN-025). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2007. http://www.sei.cmu.edu/publications/documents/07.reports/07tn025.html

Papers

Collins, M. P. “Hit-List Worm Detection and Bot Identification in Large Networks Using Protocol Graphs.” *Recent Advances in Intrusion Detection*, 10th International Symposium , Gold Coast, Australia, Sept. 2007. Berlin, Germany: Springer, Lecture Notes in Computer Science, Vol. 4637 (ISBN 978-3-540-74319-4), 2007.

Collins, M. P. “Playing Devil’s Advocate: Inferring Sensitive Information from Anonymized Network Traces.” NDSS Symposium, San Diego, CA, Feb. 2007.

Collins, M. P. “Playing Devil’s Advocate: Inferring Sensitive Information from Anonymized Network Traces.” USENIX Security Symposium, Boston, MA, Aug. 2007.

Collins, M. P., Shimeall, T., Faber, S., Janies, J., Weaver, R., & DeShon, M. “Predicting Future Botnet Addresses with Uncleanliness.” Internet Measurement Conference 2007, San Diego, CA, Oct. 2007.

Danyliw, R., Meijer, J., & Demchenko, Y. “The Incident Object Description Exchange Format.” RFC 5070, Dec., 2007. http://tools.ietf.org/html/rfc5070

Goodenough, J., Lipson, H., & Weinstock, C. “Arguing Security – Creating Security Assurance Cases.” Department of Homeland Security *Build Security In* website, Jan. 2007. https://buildsecurityin.us-cert.gov/daisy/bsi/articles/knowledge/assurance/643.html

Moore, A. P., Cappelli, D. M., Joseph, H., Shaw, E. D., & Trzeciak, R. F. “An Experience Using System Dynamics Modeling to Facilitate an Insider Threat Workshop.” *Proceedings of the 25th International Conference of the System Dynamics Society*, July 2007. http://www.systemdynamics.org/conferences/2007/proceed/papers/MOORE349.pdf

Peterson, G. & Lipson, H. “Security Concepts, Challenges, and Design Considerations for Web Services Integration.” Department of Homeland Security *Build Security In* website, Dec. 2006. https://buildsecurityin.us-cert.gov/daisy/bsi/articles/bestpractices/assembly/639.html

Shoemaker, D., Drommi, A., Ingalsbe, J., & Mead, N. R. “A Comparison of the Software Assurance Common Body of Knowledge to Common Curricular Standards.” *Proceedings of the 20th Conference on Software Engineering Education & Training*, July 2007, Dublin, Ireland. Los Alamitos, CA: IEEE Computer Society Press, 2007.

Trammell, B. “From NetFlow to IPFIX: The Evolution of IP Flow Information Export.” NANOG 41, Albuquerque, NM, Oct. 2007.

Weaver, R. & Collins, M. P. “Fishing for Phishes—Applying Capture-Recapture Methods to Estimate Phishing Populations.” *APWG 2007 eCrime Researchers Summit*, Pittsburgh, PA, Oct. 2007. http://www.cert.org/netsa/publications/ecrimes07-collins-weaver-fish-for-phish.pdf

Journal Articles

Caulkins, J., Hough, E. D., Mead, N. R., & Osman, H. “Optimizing Investments in Security Countermeasures: A Practical Tool for Fixed Budgets.” *IEEE Security & Privacy 5*, 5 (Sept./Oct. 2007): 57-60.

Mead, N. R. “Experiences in Eliciting Security Requirements.” *CrossTalk 19*, 12 (Dec. 2006): 14-19.

Mead, N. R., Shoemaker, D., & Drommi, A. “Maintaining IT’s Corporate Impact Through a Governance Framework.” *Cutter IT Journal 20*, 7 (July 2007): 30-35.

Mead, N. R. & Shoemaker, D. “Producing Alignment-Savvy CIOs.” *Cutter IT Journal 20*, 2 (Feb. 2007): 29-33.

Woody, C. & Alberts, C. “Considering Operational Security Risk During System Development.” *IEEE Security & Privacy 5*, 1 (Jan./Feb. 2007): 30-35.

Talks/Panels/Workshops

Burns, L., Linger, R., & Pleszkoch, M. "Reducing Risk and Cost in DoD Systems: Computing Software Behavior with Function Extraction Technology," National Defense Industry Association (NDIA) 8th Annual Science and Engineering Technology Conference, Charleston, SC, April 2007.

Cappelli, D. M. "A Risk Mitigation Model: Lessons Learned From Actual Insider Sabotage," CyLab Japan, Kobe, Japan, Oct. 2007.

Cappelli, D. M. "A Risk Mitigation Model: Lessons Learned From Actual Insider Sabotage," CSO Executive Forum, Chicago, IL, April 2007.

Cappelli, D. M. "A Risk Mitigation Model: Lessons Learned from Actual Insider IT Sabotage," Information Assurance Capacity Building Program, Pittsburgh, PA, July 2007.

Cappelli, D. M. "A Risk Mitigation Model: Lessons Learned From Actual Insider Sabotage," Information Security Institute, Yokohama, Japan, Oct. 2007.

Cappelli, D. M. "Detecting the Fraudulent Footprint," MIS Training Institute Summit on Insider Threats, Chicago, IL, Aug. 2007.

Cappelli, D. M. "Insider IT Sabotage: What Can You Do to Mitigate Your Risk?" American Bankers Association, Pittsburgh, PA, April 2007.

Cappelli, D. M. "Insider Threat: Insights from our Research," Director of National Intelligence Private Sector Engagement on Insider Threats, Washington, D.C., May 2007.

Cappelli, D. M. "Insider Threat Research," Center for Identity Management and Information Protection, Pittsburgh, PA, Feb. 2007.

Cappelli, D. M. "Insider Threat Research," FS ISAC, Pittsburgh, PA, March 2007.

Cappelli, D. M. "Insider Threat Research," Royal Bank of Canada, Pittsburgh, PA, Feb. 2007.

Cappelli, D. M. "Insider Threats in the Software Development Lifecycle: Lessons Learned from Actual Incidents of Fraud, Theft of Sensitive Information, and IT Sabotage," Software Engineering Process Group Conference, March 2007.

Cappelli, D. M., "Management, Human Resources, and Information Technology: How to Work Together to Prevent Insider Threats," FFRDC CIO meeting, Pittsburgh, PA, June 2007.

Cappelli, D. M. "Management, Human Resources, and Information Technology: How to Work Together to Prevent Insider Threats," FS ISAC Conference, St. Pete Beach, FL, May 2007.

Cappelli, D. M. "Management, Human Resources, and Information Technology: How to Work Together to Prevent Insider Attacks," XIII Simposium de Sistemas y Tecnologías Computacionales del Tecnológico de Monterrey Campus Toluca, Toluca, Mexico, Sept. 2007.

Cappelli, D. M. "Policy Implications of Our Insider Threat Research," National Infrastructure Advisory Council, Washington, D.C., May 2007.

Cappelli, D. M. "Risk of Insider IT Sabotage: An Empirically Based Approach to Understanding the 'Big Picture,'" Enterprise Information Management Conference, San Francisco, CA, Jan. 2007.

Cappelli, D. M. "The Evolving Threat from Insiders," Center for Identity Management and Information Protection workshop, Queenstown, MD, July 2007.

Cappelli, D. M. & Crane, E. (Department of Homeland Security). "Insider Threat & Information Security," Department of Homeland Security 2007 Security Conference and Workshop, Baltimore, MD, Aug. 2007.

Cappelli, D. M. & Moore, A. P. "A Risk Mitigation Model: Lessons Learned from Actual Insider IT Sabotage," The I3P (Institute for Information Infrastructure Protection) Consortium Meeting, Pittsburgh, PA, June 2007.

Cappelli, D. M. & Moore, A. P. "Insider Threats in the SDLC," Computer Security Institute, Washington, D.C., Nov. 2007.

Cappelli, D. M. & Trzeciak, R. F. "Comparing Insider IT Sabotage and Espionage: A Model-Based Analysis," Military Operations Research Society Workshop, Baltimore, MD, March 2007.

Caralli, R. "CERT Resiliency Engineering Framework," Financial Services Technology Consortium Annual Conference, Dallas, TX, May 2007.

Caralli, R. "Operational Risk Management Strategies," Bank Security Summit, Los Angeles, CA, June 2007.

Collins, M. P. First Workshop on Hot Topics in Understanding Botnets (HotBots '07), April 2007.

Danyliw, R. "Mitigating Network Events Through Structured Information Sharing," RSA 2007, San Francisco, CA, Feb. 2007.

Ellison, R. & Woody, C. "An Operational Approach for Assessing Security, Reliability, and Survivability Risk," System and Software Technical Conference (SSTC) 2007, Tampa, FL, June 2007.

Ellison, R. & Woody, C. "Transforming Information Assurance for Netcentric Operations: Providing Assurance Information for National Security," Survivability Analysis Framework plenary presentation, Military Operations Research Society (MORS), Laurel, MD, March, 2007.

Ellison, R. & Woody, C. "Workshop: Assessing the Quality of a Business Process Implemented Across Systems of Systems," Sixth International Conference on Commercial Off-the-Shelf (COTS)-Based Software Systems (ICCBSS) 2007, Banff, Alberta, Canada, Feb. 2007.

Faber, S. "Analysis of AS112 Traffic," 2007 OARCI (Operations Analysis and Research Center), Los Angeles, CA, Nov. 2007.

Faber, S. "Geopolitical and Diplomatic Aspects of eCrime Networks," APWG 2007 eCrime Researchers Summit, Pittsburgh, PA, Oct. 2007.

Geiger, M. "Emerging Capabilities and Practical Applications in Memory Forensics," CERT Malicious Code Analysis Workshop, Arlington, VA, Oct. 2007.

Geiger, M. "Internal Investigations: A Forensic View," I-4 Forum, Baltimore, MD, Oct. 2007.

Geiger, M. Memory analysis hands-on workshop for participants in the FIRST Technical Colloquium in Doha, Qatar, April 2007.

Geiger, M. "Memory Forensics: The Challenge, Emerging Tools and Techniques," FIRST Technical Colloquium in Doha, Qatar, April 2007.

Geiger, M. Presented on the CERT Forensics program at Identifying and Fighting Cyber Crime/Theft: A Project for Russia, a workshop organized by the Department of State for Russian Cyber Crime Enforcement officials, Feb. 2007.

Lipson, H. "Cyber Security: Protecting Our Networks and Critical Infrastructure – Technical and Policy Challenges," invited Capitol Hill Briefing hosted by the U.S. House of Representatives Committee on Homeland Security and sponsored by the American Association for the Advancement of Science (Center for Science, Technology, and Security Policy), Rayburn House Building, May 2007.

Mead, N. R., Panel Chair. "On the Feasibility of Specialization Within Software Engineering Degree Programs." IEEE Conference on Software Engineering Education & Training (CSEET'07), Dublin, Ireland, July 2007.

Mead, N. R. "Process Improvement Should Link to Security," SEPG Industry Panel 2007, Austin, TX, March 2007.

Moore, A. P. "Dynamic Stories of Real Insider IT Sabotage: Observations from Pattern Analysis," Workshop on Insider Threat and Cyber Security, held by I3P and sponsored by ARO and FSTC, Arlington, VA, June 2007.

Moore, A. P. "Preventing Insider IT Sabotage: Lessons Learned from Actual Attacks," U.S. Census Bureau's Annual Information Security Awareness and Expo, MD, June 2007.

Moore, A. P. & Trzeciak, R. T. "Insider Threat Workshop: Management, Human Resources, and Information Technology Working Together to Prevent or Detect Insider Threats," CIO Institute, Washington, D.C., Nov. 2007.

Prowell, S. "Function Extraction: Automated Software Behavior Computation for Security Analysis," NIST Static Analysis Summit II, Fairfax, VA, Nov. 2007.

Shimeall, T., DeShon, M., Faber, S., Collins, M. P., Weaver, R., & Janies, J. "Uncleanliness: Quantifying Network Reputation." APWG 2007 eCrime Researchers Summit, Pittsburgh, PA, Oct. 2007.

Trammell, B. "Requirements for a Standardized Flow Storage Solution," SAINT 2007, The 2007 International Symposium on Applications and the Internet, Hiroshima, Japan, Jan. 2007.

Trzeciak, R. F. "Insider Threat Workshop: Management, Human Resources, and Information Technology Working Together to Prevent or Detect Insider Threats," Information Systems Audit and Control Association, Tampa, FL, Nov. 2007.

Trzeciak, R. F. "Management, Human Resources, and Information Technology: How to Work Together to Prevent Insider Threats," CIO Forum and Executive Summit, Pittsburgh, PA, Sept. 2007.

Trzeciak, R. F. "Management, Human Resources, and Information Technology: How to Work Together to Prevent Insider Threats," Information Security Forum, Montreal, Canada, May 2007.

Trzeciak, R. F. "Management, Human Resources, and Information Technology: How to Work Together to Prevent Insider Threats," Pittsburgh Kiwanis Meeting, Pittsburgh, PA, Sept. 2007.

Trzeciak, R. F. & Moore, A. P. "Insider Threat Workshop: Management, Human Resources, and Information Technology Working Together to Prevent or Detect Insider Threats," International Collaboration for Advancing Security Technology, Pittsburgh, PA, Nov. 2007.

Weaver, R. "Fishing for Phishes—Applying Capture-Recapture Methods to Estimate Phishing Populations." APWG 2007 eCrime Researchers Summit, Pittsburgh, PA, Oct. 2007.

Willke, B. "Best Practices for CIIP Focusing on National Risk Management," ENISA/CERT/CC Workshop on Mitigation of Massive Cyberattacks, Porto, Portugal, Sept. 2007.

Willke, B. "CSIRT Contributions to National Efforts in Critical Information Infrastructure Protection," International Telecommunications Union Regional Workshop on Frameworks for Cybersecurity and Critical Information Infrastructure Protection, Buenos Aires, Argentina, Oct. 2007.

Willke, B. "Engineering National Cybersecurity and Critical Infrastructure Protection," International Telecommunications Union Regional Workshop on Frameworks for Cybersecurity and Critical Information Infrastructure Protection, Buenos Aires, Argentina, Oct. 2007.

Woody, C. "Megasystem Survivability Analysis Framework," Computer Security Institute 2007, Crystal City, VA, Nov. 2007.

Woody, C. "Process Improvement Should Link to Security," SEPG Industry Panel 2007, Austin, TX, March 2007.

Young, L. "Focus on Resiliency: A Process Improvement Approach to Security," North American CACS, Dallas, TX, April 2007.

Young, L. "Focus on Resiliency: A Process Improvement Approach to Security," NSA Enterprise Security Management Conference 2007, Washington, D.C., Sept. 2007.

Young, L. "Information Security Risk Assessment," Veteran's Administration Project Manager Conference, Pittsburgh, PA, May 2007.

Young, L. "Process Improvement Should Link to Security," SEPG Industry Panel 2007, Austin, TX, March 2007.

Young, L. "Resiliency Engineering Framework," CyLab Corporate Partners Conference, Pittsburgh, PA, May 2007.

Technical Leadership

Dawn M. Cappelli

Reviewer, 25th International Conference of the System Dynamics Society, July 2007

Matthew Geiger

Local chair and organizing committee member, DFRWS 2007 forensics conference

Program committee member and reviewer, DFRWS 2007 forensics conference

Richard Linger

Program Committee Member, IEEE Hawaii International Conference on System Sciences (HICSS), 2002-present

Howard Lipson

Capitol Hill briefing, "Cyber Security: Protecting Our Networks and Critical Infrastructure – Technical and Policy Challenges," May 21, 2007, hosted by the U.S. House of Representatives Committee on Homeland Security and sponsored by the American Association for the Advance of Science

Invited instructor; taught CyLab's four-day Security Engineering course to faculty members from minority-serving institutions as part of Carnegie Mellon's NSF-sponsored Information Assurance Capacity Building Program, July 2007

Member, Advisory Board for Duquesne University's Graduate Program in Computational Mathematics, 1999-present (Chair 2002-2004)

Member (founding), Carnegie Mellon Electricity Industry Center

Program Co-Chair, Third ACM Workshop on Digital Identity Management (DIM 2007), ACM CCS, Nov. 2007

Program Committee Member, Second ACM Workshop on Digital Identity Management (DIM 2006), ACM Conference on Computer and Communications Security (ACM CCS), Nov. 2006

Reviewer, 25th International Conference of the System Dynamics Society, July 2007

Reviewer, International Risk Governance Council (Geneva, Switzerland) of their document *Managing and Reducing Social Vulnerabilities from Coupled Critical Infrastructures*

Session Chair, "User-centric Identity Management Frameworks" for the Second ACM Workshop on Digital Identity Management (DIM 2006), ACM CCS, Nov. 2006

Nancy R. Mead

Editorial Board Member, 2002-present, *Requirements Engineering Journal*

Timothy Shimeall

Project Management Technical Advisor, NICIAR Technical Conference, Boston, MA, Sept. 2007

Rhiannon Weaver

Reviewer, *Cognitive Science*, 2007
Reviewer, *Psychometrika*, 2005-2006

Carol Woody

Lifecycle Information Assurance Workshop, Track Chairperson, Military Operations Research Society (MORS), Laurel, MD, March 2007

Program Committee, Hawaii International Conference on System Sciences (HICSS) - 41, Jan. 2007

Security Track Chairperson, SEPG 2007, Austin, TX, March 2007

Workshop Program Committee, Sixth International Workshop on Requirements for High Assurance Systems (RHAS-6), Delhi, India, Oct. 2007

Biographies

Archie Andrews

Archie Andrews is a senior member of the CERT technical staff. He currently leads the effort to develop a catalog of new research initiatives to help guide CERT in addressing challenging issues in cybersecurity for the nation. Prior to his current assignment, Andrews led the effort to form Qatar's national computer security incident response team, Q-CERT, located in Doha, Qatar, and the computer security incident response capability for the Middle Eastern region, the GCC-CERT.

Prior to joining CERT, Andrews was Director of the Information Protection Technology group of the Advanced Technology Institute, a private nonprofit research and development management firm located in Charleston, South Carolina. He managed and led a number of projects that addressed information security and privacy protection issues for the Department of Defense, the Veteran's Administration, and commercial firms primarily in the healthcare and insurance industries. Andrews served in the U.S. Army Signal Corps, retiring at the rank of colonel. During his military career, Andrews commanded units in Europe and Vietnam, served in a variety of staff and teaching roles, including instructor in information technology at the U.S. Army War College and U.S. representative for the Joint Chiefs of Staff on Command, Control, Intelligence and Electronic Warfare in NATO, Brussels, Belgium. His last army assignment was as Director, U.S. Army Computer Science School in Augusta, Georgia. He holds a bachelor's degree from the University of Georgia and a master's degree from American University, Washington, D.C.

Dawn Cappelli

Dawn Cappelli is a senior member of the CERT technical staff. She has over 25 years experience in software engineering, including programming, technical project management, information security, and research. She is technical lead of CERT's insider threat and espionage research, including the Insider Threat Study conducted jointly by the U.S. Secret Service and CERT.

Before joining CERT in 2001, Cappelli was the Director of Engineering for the Information Technology Development Center of the Carnegie Mellon Research Institute (CMRI). While with CMRI, she was a technical and program manager for a variety of information networking projects. These projects included the design and development of large-scale databases and Internet-based systems that adhered to data privacy and security requirements, the design and implementation of multi-organizational portals for preparation and response to weapons of mass destruction and collaboration among public health department epidemiologists, and the design and development of a networked media solution for remote collaboration between teachers of children with special needs and consultants.

Previously she worked in Computing Services at Carnegie Mellon University, where she led several teams in the areas of web application development, database development, and networked media. Cappelli began her career at Carnegie Mellon University in the Information Technology Department at the SEI, managing development of various web and database applications, as well as managing the redesign of the SEI's website by an interdisciplinary project team. Prior to her career at Carnegie Mellon, Cappelli worked for Westinghouse Electric Corporation. While at Westinghouse, she designed and developed systems for nuclear power plants, including real-time graphical user interface systems for power plant operators and computer-aided engineering systems for nuclear plant designers.

Cappelli regularly presents at national conferences and is also adjunct professor in Carnegie Mellon's Heinz School of Public Policy and Management. Cappelli has been with Carnegie Mellon since 1988.

Richard Caralli

Richard Caralli is a senior member of the technical staff in CERT's Survivable Enterprise Management group.

Caralli is currently the team leader for developing and delivering methods, tools, and techniques for enterprise security and resiliency management. His work includes the exploration and development of process-oriented approaches to security management. At present, Caralli is heading a team that is developing a framework for security process improvement called the CERT Resiliency Engineering Framework. In other work at the SEI, Caralli has been active in developing and delivering various information security risk assessment, analysis, and management technologies for customers in the government and the private sector.

Before joining the SEI, Caralli was responsible for developing the information security assessment and risk management capabilities of the CyberSecurity Center at Carnegie Mellon Research Institute. In addition, Caralli has over 25 years experience in information technology (particularly systems analysis and information systems audit and security) in Fortune 1000 companies covering banking and finance, steel production, light manufacturing, and energy industries. Caralli's most recent work in the private sector included 12 years at Consolidated Natural Gas (now Dominion Resources). Of note, Caralli was responsible for managing Consolidated's global Y2K project in the U.S. and overseeing Y2K activities in Australia and Argentina, concentrating on information and process control systems.

Caralli holds a BS degree in Accounting from St. Vincent College and an MBA with a concentration in Information Technology from the John F. Donahue Graduate School of Business at Duquesne University. He has previously been on the adjunct faculty at Community College of Allegheny County and is a frequent lecturer in Carnegie Mellon's Heinz School of Public Policy and Management and the CIO Institute's Executive Education programs.

Anthony Cebzanov

Anthony Cebzanov is a member of the CERT technical staff. As a software developer in the Network Situational Awareness group, he participates in the design and implementation of software systems used to measure and categorize security risks through network flow analysis.

Prior to joining the SEI, Cebzanov worked as an Information Systems Engineer for The Vanguard Group of Investment Companies, where he designed and developed authentication, authorization, and access control systems.

Cebzanov holds Master of Software Engineering and Bachelor of Science in Computer Science degrees from The Pennsylvania State University.

M. Patrick Collins

M. Patrick Collins has been a member of the analysis team in the Network Situational Awareness group since the group's founding. In this capacity, he worked with the late Dr. Suresh Konda to develop the first version of the SiLK packing system and analysis suite. He now focuses on questions in traffic analysis, particularly issues of anonymity and application identification. Collins has published and holds patents for work in several fields, including security and knowledge management.

Collins holds a BS in Physics and an MS in Electrical Engineering from Carnegie Mellon University and is currently a PhD candidate in Electrical Engineering. Prior to his work at CERT, he worked as a researcher for the Institute for Complex Engineered Systems and as a consultant for several Fortune 1000 companies.

Roman Danyliw

Roman Danyliw is the technical manager of the CERT Network Situational Awareness (NetSA) group. Danyliw directs applied research and operations in support of the Department of Homeland Security, Department of Defense, and the Internet community to form an actionable and holistic understanding of network security threats.

Previously, Danyliw was a member of the CERT Coordination Center (CERT/CC), where he performed analysis on widespread Internet activity. He was the primary architect of the CERT/CC situational awareness capability, AirCERT, which focused on creating analytical products from event streams and contextual information collected from a wide array of organizations. In this work, Danyliw was one of the core developers to the open source, network instruction detection system Snort and ACID. He also co-chaired the Internet Engineering Task Force (IETF) Incident Handling (INCH) working group and co-authored the Incident Object Description Exchange Format (IODEF).

Prior to joining the SEI, Danyliw ran various corporate and university networks. His professional research interests focus on intrusion detection, network measurement, data representation standards, and cross-organizational coordination on security threats.

Danyliw holds a BS from La Salle University and an MS from Carnegie Mellon University.

Markus De Shon

As the analysis team lead for the Network Situational Awareness group, Markus De Shon guides the research activities of the network security analysts and provides support to analysis operations for government agency customers.

Prior to joining the SEI, De Shon was Chief Scientist at SecureWorks, Inc. His work included designing intrusion prevention (IPS) technologies, developing IPS signatures based upon vulnerabilities and reverse engineering of malicious code, analyzing network activity, and acting as the final incident handling escalation point for an IPS service to nearly 1,000 small to medium-sized businesses.

De Shon holds a PhD in Nuclear Physics and an MS in Health Physics from the Georgia Institute of Technology.

Robert J. Ellison

As a member of the CERT Survivable Systems Engineering group, Robert J. Ellison has served in a number of technical and management roles. He was a project leader for the evaluation of software engineering development environments and associated software development tools. He was also a member of the Carnegie Mellon University team that wrote the proposal for the SEI; he joined the new FFRDC in 1985 as a founding member.

Before coming to Carnegie Mellon, Ellison taught mathematics at Brown University, Williams College, and Hamilton College. At Hamilton, he directed the creation of the Computer Science curriculum. Ellison belongs to the Association for Computing Machinery (ACM) and the IEEE Computer Society.

Ellison regularly participates in the evaluation of software architectures and contributes from the perspective of security and reliability measures. His research draws on that experience to integrate security issues into the overall architecture design process. His current work explores developing reasoning frameworks to help architects select and refine design tactics to mitigate the impact of a class of cyber attacks. He continues to work on refinements to the Systems Analysis Method.

Sidney Faber

As a member of the analysis team in the Network Situational Awareness group, Sid Faber supports customers by providing detailed reports of current and historical network activities. Much of his time is spent studying usage and routing patterns of very large networks and in understanding large scale DNS trends.

Prior to joining the SEI, Faber worked as a security architect with Federated Investors, one of the largest investment managers in the United States. His experience includes over 10 years in software application development and evaluation, and 5 years in the U.S. Navy Nuclear Power program. Faber holds GIAC certifications for Intrusion Detection (GCIA), Windows Security Administrator (GCWN), and Forensics Analyst (GCFA).

Matthew Geiger

Matthew Geiger is a forensic specialist and researcher at CERT. His recent work has focused on new utilities for memory acquisition and analysis and counter-forensic tool performance. Prior to joining CERT, Geiger resided for about 14 years in Asia. As a forensic analyst in the private sector, Geiger conducted investigations involving corporate fraud, network intrusion, proprietary data theft, corruption, and official misconduct for clients that included Fortune 500 companies.

Jeff Janies

Jeff Janies has been a member of the analysis team within the CERT Network Situational Awareness group for a year. Here he has worked on developing large-scale network visualizations and passive network inventories. Prior to joining the Network Situational Awareness group, Janies was a graduate student at the University of South Carolina in the Computer Science program. There his primary research areas were network intrusion detection systems and wireless ad hoc sensor network security.

Janies holds an MS in Computer Science from the University of South Carolina and a BS from Louisiana State University.

Richard Linger

Richard Linger is manager of the CERT Survivable Systems Engineering group and STAR*Lab. He has extensive experience in function-theoretic foundations for software engineering. Linger directs research and development for the function extraction project for software behavior computation and the Flow-Service-Quality (FSQ) engineering project for network-centric system development. He serves as a member of the faculty at the Carnegie Mellon University Heinz School of Public Policy and Management. At IBM, Linger partnered with Dr. Harlan Mills, IBM Fellow, to create Cleanroom Software Engineering technology for development of ultra-reliable software systems, including box-structure specification, function-theoretic design and correctness verification, and statistical usage-based testing for certification of software fitness for use. He has extensive experience in project management; system specification, architecture, design, verification, and certification; software re-engineering and reverse engineering; and technology transfer and education. He has published three software engineering textbooks, 12 book chapters, and over 60 papers and journal articles. He is a member of the IEEE and the ACM.

Howard F. Lipson

Howard F. Lipson is a senior member of the CERT technical staff. Lipson has been a computer security researcher at CERT for more than fifteen years. He is also an adjunct professor in Carnegie Mellon University's Department of Engineering and Public Policy, and an adjunct research faculty member at the Carnegie Mellon Electricity Industry Center. He has played a major role in extending security research at the SEI and Carnegie Mellon into the new realm of survivability, developing many of the foundational concepts and definitions and making key contributions to the creation of new survivability methodologies. Lipson has been a chair of three IEEE Information Survivability Workshops. His research interests include the foundational concepts of survivability, the analysis and design of survivable systems and architectures, survivable systems simulation, critical infrastructure protection (specifically the electric power grid), and the technical and public policy aspects of Internet traceability and anonymity. He has been co-principal investigator on a National Science Foundation award to investigate "Secure and Robust IT Architectures to Improve the Survivability of the Power Grid."

Lipson's early research at Carnegie Mellon included detailed workflow analyses of the incident response and vulnerability handling activities at the CERT/CC. He later designed and developed tools to automate and improve key aspects of the incident response and security advisory processes. His work was recognized as a primary factor in the CERT/CC's ability to sustain its effectiveness in the face of the rapid growth of the Internet. Prior to joining Carnegie Mellon, Lipson was a systems design consultant, helping to manage the complexity and improve the usability of leading-edge software systems.

Earlier, he was a computer scientist at AT&T Bell Labs, where he did exploratory development work on programming environments, executive information systems, and integrated network management tools. Lipson holds a PhD in computer science from Columbia University. He is a member of the IEEE and the ACM.

Nancy R. Mead

Nancy R. Mead is a senior member of the technical staff in the CERT Survivable Systems Engineering group. Mead is also a faculty member in the Master of Software Engineering and Master of Information Systems Management programs at Carnegie Mellon University. She is currently involved in the study of secure systems engineering and the development of professional infrastructure for software engineers. She also served as director of education for the SEI from 1991 to 1994. Her research interests are in the areas of information security, software requirements engineering, and software architectures.

Prior to joining the SEI, Mead was a senior technical staff member at IBM Federal Systems, where she spent most of her career in the development and management of large real-time systems. She also worked in IBM's software engineering technology area and managed IBM Federal Systems' software engineering education department. She has developed and taught numerous courses on software engineering topics, both at universities and in professional education courses.

Mead has more than 100 publications and invited presentations, and has a biographical citation in Who's Who in America. She is a Fellow of the Institute of Electrical and Electronic Engineers, Inc. (IEEE) and the IEEE Computer Society, and a member of the ACM. Mead serves on the Editorial Boards for IEEE Security and Privacy and the Requirements Engineering Journal, and is a member of numerous advisory boards and committees.

Mead received her PhD in mathematics from the Polytechnic Institute of New York, and a BA and an MS in mathematics from New York University.

Andrew P. Moore

Andrew Moore is a senior member of the CERT technical staff. Moore explores ways to improve the security, survivability, and resiliency of enterprise systems through insider threat and defense modeling, incident processing and analysis, and architecture engineering and analysis. Before joining the SEI in 2000, he worked for the Naval Research Laboratory (NRL) investigating high-assurance system development methods for the Navy. He has over twenty years' experience developing and applying mission-critical system analysis methods and tools, leading to the transfer of critical technology to both industry and the military. Moore received his BA in Mathematics from the College of Wooster and MA in Computer Science from Duke University.

While at the NRL, Moore served as member of the U.S. Defense Science and Technology review (Information Technology TARA) panel on Information Assurance; the International Technical Cooperation Program, Joint Systems and Analysis Group on Safety-Critical Systems, (TTCP JSA-AG-4); and the Assurance Working Group of DARPA's Information Assurance Program. He has served as principal investigator on numerous projects sponsored by NSA and DARPA. He has also served on numerous computer assurance and security conference program committees and working groups. Moore has published a book chapter and a wide variety of technical journal and conference papers. His research interests include computer and network attack modeling and analysis, IT management control analysis, survivable systems engineering, formal assurance techniques, and security risk analysis.

Richard Pethia

Richard Pethia is the Director of the CERT Program. The program conducts research and development activities to produce technology and systems management practices that help organizations recognize, resist, and recover from attacks on networked systems. The program's CERT Coordination Center (CERT/CC) has formed a partnership with the Department of Homeland Security to provide a national cyber security system, US-CERT. In 2003, Pethia was awarded the position of SEI Fellow for his vision and leadership in establishing the CERT/CC, for his development of the research and development program, and for his ongoing work and leadership in the areas of information assurance and computer and network security. Pethia is also a co-director of Carnegie Mellon University's CyLab. CyLab is a public/private partnership to develop new technologies for measurable, available, secure, trustworthy, and sustainable computing and communications systems. This university-wide, multidisciplinary initiative involves more than 200 faculty, students, and staff at Carnegie Mellon.

Mark Pleszkoch

Mark Pleszkoch is a senior member of the CERT technical staff. He is an expert in function-theoretic mathematical foundations of software, and focuses on automation of formal methods. As a member of the function extraction research and development team, he is responsible for creating theoretical foundations and engineering automation for FX systems.

Prior to joining CERT, Pleszkoch worked at IBM for 21 years in various capacities. As a member of IBM's Cleanroom Software Technology Center, he provided education and consultation to clients in software process, software engineering technologies, and software testing. He was the principal architect of the IBM Cleanroom Certification Assistant tool set for statistical testing automation. Pleszkoch received his PhD in Computer Science from the University of Maryland

and an MA and a BA in Mathematics from the University of Virginia. He has a number of publications in formal methods and software engineering. He served on the adjunct faculty in the Computer Science department of the University of Maryland, Baltimore County, from 1986 to 1995. As an undergraduate, Pleszkoch was a Putnam fellow of the Mathematics Association of America, and is a member of the IEEE and the Association for Symbolic Logic.

John Prevost

Since arriving at CERT, John Provost has worked primarily on database and user interface integration for the SiLK network data collection and analysis toolset. His past work is quite diverse, with a range that includes systems administration for the Carnegie Mellon School of Computer Science, developing data visualization software at MAYA Design, Inc., and designing database-backed websites at ArsDigita.

Stacy Prowell

Stacy Prowell is a senior member of the CERT technical staff, and chief scientist of STAR*Lab. He is an expert in the function-theoretic foundations of software, and is currently conducting research and development for function extraction technology. Prowell has managed both commercial and academic software development projects and consulted on design, development, and testing of applications ranging from consumer electronics to medical scanners, from small embedded real-time systems to very large distributed applications.

Prior to joining the SEI in 2005, Prowell was a research professor at the University of Tennessee. To support wider adoption of rigorous methods in industry, he started the Experimentation, Simulation, and Prototyping (ESP) project at the University of Tennessee, which develops software libraries and tools to support application of model-based testing and sequence-based specification. Software developed by this program is in use by over 30 organizations. Prior to working at the university, he served as a consultant in the software industry. His research interests include rigorous software specification methods, automated statistical testing, and function-theoretic analysis of program behavior. Prowell holds a PhD in Computer Science from the University of Tennessee and is a member of the ACM, IEEE, and Sigma Xi.

Kirk Sayre

Kirk Sayre is an expert in the function-theoretic mathematical foundations that are the basis for function extraction technology. He is currently working on development of the core rewriting engine for the FX system, as well as on formal testing for the system.

Prior to joining CERT, Sayre was a research professor at the University of Tennessee, where he developed an automated testing framework for the certification of generic scientific computing libraries. In his position at UT, Sayre also developed a CASE tool to support the editing and creation of rigorous sequence-based software specifications. This tool is currently being used on software projects at Oak Ridge National Laboratory and Bosch. Sayre has developed software in many different areas, including educational web applications, automated testing tools, CASE tools, medical devices, and weapons systems. He received a BS in Computer Science from Bucknell University in 1992, an MS in Computer Science from American University in 1994, and a PhD in Computer Science from the University of Tennessee in 1999.

Timothy J. Shimeall

Timothy J. Shimeall is a senior member of the technical staff with the CERT Network Situational Awareness group, where he is responsible for overseeing and participating in the development of analysis methods in the area of networked systems security and survivability. This work includes development of methods to identify trends in security incidents and in the development of software used by computer and network intruders. Of particular interest are incidents affecting defended systems and malicious software that are effective despite common defenses. Shimeall is also an adjunct professor of the Carnegie Mellon University Heinz School of Public Policy and Management, with teaching and research interests focused in the area of information survivability. He is an active instructor in information security management and information warfare and has led a variety of survivability-related independent studies.

James Stevens

James Stevens is a member of the Survivable Enterprise Management group and a senior member of the CERT technical staff. As a thought leader on the information security assessment and evaluation team, Stevens conducts research, development, and process improvement activities in risk, threat, and vulnerability management methodology related to information security management.

Stevens' current focus is to develop strategies and provide support for national and international critical infrastructure protection initiatives. This work includes the exploration of capabilities, frameworks, and models for managing security at and above the business process level. In other work at the SEI, Stevens has been active in developing and delivering methods, tools, and techniques that assist organizations in the identification, analysis, mitigation, and management of security risks. Stevens has also held positions on the infrastructure team and the incident response team within the CERT Program.

Stevens holds an MBA from Carnegie Mellon University and a BS in Electrical Engineering from the University of Notre Dame.

Randall F. Trzeciak

Randy Trzeciak is currently a senior member of the CERT technical staff. Trzeciak is a member of a team in CERT focusing on insider threat research. The studies analyze the physical and online behavior of malicious insiders prior to and during network compromises. Other insider threat research uses system dynamics modeling for risk analysis of the impacts of policy decisions, technical security measures, psychological issues, and organizational culture on insider threat. Trzeciak also is an adjunct professor in Carnegie Mellon's H. John Heinz School of Public Policy and Management.

Prior to his current role in the CERT Program, Trzeciak managed the Management Information Systems (MIS) team in the Information Technology Department at the SEI. Under his direction, the MIS team developed and supported numerous mission-critical, large-scale, relational database management systems.

Prior to his time working at the SEI, Trzeciak was a software engineer for the Information Technology Development Center of the Carnegie Mellon Research Institute (CMRI), responsible for a variety of information networking projects. These projects included the design and development of large-scale databases and Internet-based systems that adhered to data privacy and security requirements; the design and implementation of multi-organizational portals for preparation and response to weapons of mass destruction; and collaboration among public health department epidemiologists.

Prior to his career at Carnegie Mellon, Trzeciak worked for Software Technology, Incorporated (STI) in Alexandria, Virginia. For nine years, he was a consultant to the Naval Research Laboratory (NRL) working on numerous projects designing, building, and supporting large-scale relational database management systems. During his employment with STI, Trzeciak also filled the role of Information Systems Business Manager.

Trzeciak holds an MS in Management from the University of Maryland, a BS in Management Information Systems, and a BA in Business Administration from Geneva College.

Cal Waits

Cal Waits is a member of the CERT technical staff. As a member of the Forensic team, Waits develops digital forensic training material for law enforcement and intelligence agencies. His research also focuses on emerging trends in the forensic field and tool development.

Before joining the SEI, Waits worked for the National Security Agency. He holds an MS degree in Information Security from Carnegie Mellon University.

Gwendolyn H. Walton

Gwendolyn H. Walton is a senior member of the CERT technical staff. As a member of the Survivable Systems Engineering group, she is currently involved in research on theoretical foundations for computation and automated analysis of software security attributes and function extraction for malicious code.

Prior to joining the SEI, Walton held faculty positions at Florida Southern College and the University of Central Florida. She published over 30 journal and conference papers and directed the research of 2 PhD students, 15 MS students, and 4 undergraduate students. Previously Walton served as President of Software Engineering Technology Inc.; Assistant Vice President, Division Manager, Project Manager, and Senior Systems Analyst for Science Applications International Corporation; Senior Data Systems Programmer for Lockheed Missiles and Space Company; and Research Associate for Oak Ridge National Laboratory.

Walton received her PhD in Computer Science, MS in Mathematics, and BS in Mathematics Education from the University of Tennessee. She is a senior member of IEEE and the IEEE Computer Society, a senior member of the Society of Women Engineers, and a member of the ACM.

Rhiannon Weaver

Rhiannon Weaver joined the CERT Network Situational Awareness group in September 2006 and provides support to various research initiatives in the group through statistical and mathematical modeling.

Weaver holds a BS in Mathematics and BS in Computer Science from Penn State University and an MS in Statistics from Carnegie Mellon University. She is currently working on her PhD in Statistics at Carnegie Mellon. Her interests include scientific computing, Bayesian statistics, and dynamic hierarchical modeling.

David W. White

David White is a senior staff member in the CERT Program. White is responsible for developing and implementing strategies that lead to the widespread dissemination and use of methods, techniques, and tools that help organizations manage information security risks. He is also a member of the development team for the CERT Resiliency Engineering Framework, a process improvement framework that provides guidelines for managing security and business continuity from an enterprise risk management perspective.

White has a bachelor's degree in civil engineering and public policy from Carnegie Mellon University and master's degree in civil engineering with a specialization in robotics from Carnegie Mellon University. He is currently based in New York City.

Bradford J. Willke

Bradford Willke is a member of the Survivable Enterprise Management group and a senior member of the CERT technical staff. Willke is responsible for leading the assessment and evaluation team. He conducts research, development, and process improvement activities in risk, threat, and vulnerability management methodology related to information security management. Willke also leads projects to develop strategies and provide support for national and international critical infrastructure protection initiatives.

Before joining the SEI, Willke was a technical intern with Southern Illinois University at Carbondale, where he installed, managed, and maintained the university's first firewall, which protected the university's multi-million dollar Oracle investment. He also provided technology and security management for computing resources of the 90th Security Police Squadron, Francis E. Warren Air Force Base, Wyoming. Willke served in the United States Air Force as a law enforcement specialist and organizational computer security officer from 1993 to 1997.

Willke holds a professional certificate in information protection and security from the University of New Haven, and received a BS in information systems technologies from Southern Illinois University at Carbondale in 1999. He received an AAS in criminal justice from the Community College of the Air Force in 1997, and has been a Certified Information System Security Professional (CISSP) since 2005.

Carol Woody

Carol Woody is a senior member of the CERT technical staff. Her research is focused on ways to address software design and development that improve the security of the implemented results. She is leading several research projects for the DoD and other federal agencies that are expanding the usage of the Survivable Analysis Framework.

Woody is a member of the Survivable Enterprise Management group in the CERT Program. She participated in the development of the Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE®) methodology for applying good security practices through risk management. In addition she developed and piloted a version of OCTAVE for use in K-12 schools and school districts.

Woody has over 25 years of experience in software development and project management covering all aspects of software and systems planning, design, development, and implementation in large complex organizations. Before coming to the SEI, she consulted for New York City as a strategic planner for the Administration of Children's Services addressing the financial technology needs of the $2 billion organization during the formulation of the agency and its transition through Y2K. She also managed the user testing for a timekeeping application purchased by NYC to handle 160,000 employees in over 100 agencies; activities included work force scheduling for police, fire, sanitation, and correction.

Woody has a biographical citation in Who's Who in American Women and Who's Who in Finance and Industry. She is a member of IEEE, the ACM, and PMI.

Woody holds a BS in Mathematics from The College of William and Mary, an MBA with distinction from Wake Forest University, and a PhD in Information Systems from NOVA Southeastern University. She was selected as 2007 alumni of the year by the NOVA Graduate School of Computing and Information Sciences.

Lisa Young

Lisa Young, senior member of the SEI technical staff, has 20+ years of experience in the information technology and telecommunications industry. She holds the designation of Certified Information Systems Auditor (CISA), Certified Information Systems Security Professional (CISSP) and is experienced in IT governance, information audit and security, and risk management.

Young teaches the Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE) risk-based security assessment methodology at the Software Engineering Institute. Her current line of research provides guidelines for improving the way organizations manage the processes of security, IT Operations, business continuity, compliance, and audit to support the organization's mission and critical success factors.

CERT | Software Engineering Institute
Carnegie Mellon

The 2007 CERT Research Report was produced by SEI Communications.

Executive Editor
Richard Linger

Editor-in-Chief
Pamela Curtis

Design
Cat Zaccardi

Production
David Gregg
Melissa Neely

Software Engineering Institute
Carnegie Mellon University
4500 Fifth Avenue
Pittsburgh, PA 15213-2612

Phone: 412-268-5800
Fax: 412-268-5758
www.cert.org

CERT | Software Engineering Institute
Carnegie Mellon