Précis: Research on Techniques and Tools for Computer Security: The COAST Project and Laboratory

Director: *Eugene H. Spafford*
Department of Computer Sciences
1398 Computer Science Building
Purdue University
West Lafayette, IN 47907–1398
spaf@cs.purdue.edu
+1 765 494-7825 (office)
+1 765 494-0739 (fax)

Last revision: 2 June 1998

Abstract

The goal of the **COAST** project is to establish a long-term research program exploringing new approaches to computer security and computer system management in a first-class educational environment. The principal focus will be on techniques and tools for common, off-the shelf (COTS) systems without military-grade security (in the traditional DoD TCSEC, "Orange Book" sense), especially legacy systems in wide-spread use. We anticipate that the work we do in this area will also aid in the development of future systems.

A secondary goal of **COAST** is to advance the state of education in computer security, and includes the establishment of a dedicated, dual-use laboratory. We intend for our research to be broad-based and to draw from experience and expertise in other areas of computing such as software engineering, language design, operating systems, computer networking, and databases. It will also draw on expertise in related non-computer areas, as appropriate to advance our goals.

COAST is supported through gifts and by donations of software and equipment. Selected work may be supported through research contracts. Our sponsors get early access to **COAST** products, software and publications, as well as unique opportunities for research collaboration. **COAST** sponsorship benefits (see section 3 for a full list) also include special access to the faculty and students (see section 4) conducting the research. Our sponsors help to focus our research: we expect continuing, direct feedback from them on our research directions and results.

By combining support from many different sources, we help leverage the contributions of individual sponsors, resulting in significantly greater levels of effort. We expect our research to lend new insights into the development and use of tools for enhanced computer security and operational management. By involving our sponsors in our work, we improve its applicability and enhance technology transfer. We expect that our research results will have a significant impact on the current state of practice as our results are brought into use and as our graduates enter the workforce.

1 Introduction

It is clear that computer security is an area of increasing, major concern and that all of society is facing an increasing number of severe challenges related to information security. Incidents related to disclosure of personal and corporate date, wide-scale computer breakins, and the exponential growth in the number of computer viruses being written and discovered all indicate an increasing threat to effective use of computing resources.[1] There have already been many documented cases of economic espionage, vandalism, theft, and other major economic crimes, some of which involve losses in the tens of millions of dollars *per incident*.[36]

Many computer crimes go undetected. Others go unreported because the victims fear that any publicity about their losses (and by implication, their vulnerabilities) will result in a loss of confidence in their businesses. Additionally, there has been a huge number of cases involving smaller losses, most of which may not have been reported to the authorities for a simple reason: nearly everyone is aware that law enforcement is undertrained, underequipped, and understaffed to cope with even a minute fraction of the current flood of computer crime—and this imbalance seems to be steadily improving for the vandals and crooks.

The threat from violations of computer security are numerous and diverse. They include loss from fraud and theft, economic and international espionage, sabotage, terroristic activities, computer viruses, vandalism, and support of other forms of crime. Furthermore, not all of the criminal activities are directed at government, commerce and other organizations: violations of personal privacy, harassment, "stalking," libel, and other activities threaten individuals as well.

A few years ago, the report *Computers at Risk*[40], forcefully outlined several critical security problems facing computer users. Few of the recommendations in that study were addressed, and the problems have become even more pressing in the intervening years. Our increasing reliance on computers for critical applications poses increasing temptation for unauthorized criminal and terroristic activity. The recent report from the Presidential Commission on Critical Infrastructure Protection found that threats to the U.S. computing infrastructure to be one of their most significant concerns.[31] The situation is getting worse: our increased connectivity provided by new network technologies simply amplifies the existing threats that we do not yet completely understand. For example, in 1981, the experimental IP protocol suite was introduced as the number of ARPANET hosts exceeded 210; today, we have a worldwide network of several million machines using the same protocol, designed at a time when thousands of connected machines was a wild, unlikely fantasy.

The increasingly widespread use of computer technologies involving distributed databases and parallel and distributed processing adds new variables that have not, as yet, been adequately examined. Initiatives that link together computing systems from around the world and that provide access to more users will only add to the potential for security problems. In his State of the Union Address in January 1997, President William Clinton voiced a goal of connecting every school and library in the United States into the Internet. Are we prepared for the problems that may arise as a result in addition to the perceived benefits of having such widespread access available by the general public?

As was noted in a recent Office of Technology Assessment report:

> Information networks are changing the way we do business, educate our children, deliver government services, and dispense health care. Information technologies are intruding in our

[1] See, for instance, [35, 36], and the on-going series of advisories from response teams such as the CERT Coordination Center, Department of Energy CIAC, NASA NASIRC, and DISA ASSIST.

> lives in both positive and negative ways.... As businesses and governments become more dependent on networked computer information, the more vulnerable we are to having private and confidential information fall into the hands of the unintended or unauthorized person.... Otherwise, concerns for the security and privacy of networked information may limit the usefulness and acceptance of the global information infrastructure.[32, Forward]

The problems are especially pressing in the arena of national defense. Consider this statement in Duane Andrews' cover letter in the Defense Science Board's November 1996 task force report on Information Warfare – Defense[2] (emphasis ours):

> We conclude that there is a need for extraordinary action to deal with the present and emerging challenges of defending against possible information warfare attacks on facilities, information, information systems, and networks of the United States which would seriously affect the ability of the Department of Defense to carry out its assigned missions and functions. We have observed an increasing dependency on the Defense Information Infrastructure and increased doctrinal assumptions regarding the continued availability of that infrastructure. *This dependency and these assumptions are ingredients in a recipe for a national security disaster.*

It is interesting to note that this conclusion is independent of whether or not there is concern for protection against directed "information warfare." Widespread criminal enterprises, selected actions by anarchists, or random acts of vandalism can also have ruinous effects on our safety as a nation. Furthermore, as more and more commercial entities move to "internet commerce," the potential for serious disruption of our national economy also looms large.

Consider: in 1980, there were under 200 hosts on the ARPANET.[37] A few countries were beginning to experiment with national networks. The first commercial workstations were not yet on the market, and the PC industry was in its infancy. The first, primitive Usenet newsgroups were flowing among a few dozen machines using 30 cps[2] modem technology. And the World-Wide Web was pure science fiction and a dozen years away.

Now, less than 20 years later — slightly more than one-half of a human generation or one-fourth of human lifetime — we have a global network that reaches to over 120 countries on all seven continents. We have tens of millions of people using the Internet daily. Governments are using the Internet to run their daily affairs. Commercial overload of service providers makes front-page news in all the major newspapers. Late night comics and editorial cartoons commonly refer to the WWW and network address. The President's State of the Union address is broadcast live around the world over the Internet. Some people estimate that tens of billions of dollars are already invested and changing hands in commerce facilitated through on-line communications.

Where will we be in another 15–20 years? Although it is difficult for any of us to even imagine the changes in store, there is at least one clear aspect of that future: it will be designed tomorrow, in large part, by today's students using current methods in deployment and development. Some of them will enter the workforce and design the technology that will change our lives. Others will initiate the changes with their research projects soon to be underway. And still others will be wrought by those who are soon to be seeking re-education in high-tech fields so as to be productive employees of the 21st century. We should be asking if we are adequately investing in this vital component of our future.

[2] Characters per second

The urgency of this challenge is well-stated in the summary of the report[3] of the Joint Security Commission (we have emphasized some text from the original):

> Nowhere is this more apparent than in the area of information systems and networks. The Commission considers the security of information systems and networks to be *the major security challenge of this decade and possibly the next century* and believes that *there is insufficient awareness of the grave risks we face in this area*. The nation's increased dependence upon the reliable performance of the massive information systems and networks that control the basic functions of our infrastructure carries with it an increased security risk. Never has information been more accessible or more vulnerable. This vulnerability applies not only to government information but also to the information held by private citizens and institutions. We have neither come to grips with the enormity of the problem *nor devoted the resources necessary to understand fully, much less rise to, the challenge. . . . Protecting the confidentiality, integrity, and availability of the nation's information systems and information assets—both public and private—must be among our highest national priorities*[19, p. 2]

This applies to *every* nation.

Academic Security Education in the U.S.

This incredible pace of technology is changing our world so rapidly, there is clearly little chance to roll back the clock and reimplement decisions that may have negative security implications. To ensure safe computing, the security (and other desirable properties, such as safety) must be designed in from the start. To do that, we need to be sure all of our students understand the many concerns of security, safety, privacy, integrity, and reliability.

Unfortunately, this has not happened in recent years. For instance, consider the production of the software on which we currently depend. Commercial software vendors are *still* writing and releasing software needing patches for "bugs" that were well-known as security problems over 20 years ago![4] Even when highly-publicized problems occur, such as the buffer overflow problem exploited by the 1988 Morris "Internet Worm"[10, 11], or the year 2000 date problem, those same software faults continue to be incorporated into new operating systems and applications.

Systems continue to be built using techniques known to be unsafe. Why aren't these problems avoided? Why is it that our students do not learn better security techniques? It is almost certainly because so few of them have access to appropriate education in such topics.

Information security/computer and network security, as an area of specialization, is difficult to accurately define. Even professionals working in this area have difficulty agreeing on an exact definition that appropriately encompasses the field. Part of the reason that security is difficult to describe is because it draws heavily upon so many areas of computing. In at least one sense, it seems closely related to software engineering — computer security is devoted to ensuring that software and hardware meet their specifications

[3] This commission was composed of personnel of the U.S. Department of Defense, Central Intelligence Agency, and Department of Energy. The charter of the Commission was to evaluate the current state of security in their agencies and the U.S. Government, and to suggest needs and directions.

[4] C.f. [1], [24] and [30].

and requirements when used in a potentially hostile environment. Computer security thus includes issues in computer system specification, verification, testing, validation, safety, and reliability. However, security encompasses much more than these issues, including topics in (at the least) operating systems design, architectural design, information security, risk analysis and prediction, database organization, encryption and coding, formal models of computation, fault tolerance, network and protocol design, supportive interface design, government regulation and policy, managerial decisions, security awareness, law, ethics, and education.

The difficulty in defining computer security is also reflected in the scattered and underdeveloped educational and research programs in the area. Many other fields of computing research have well-defined bodies of educational literature, major research centers funded by government and industry, and a substantial student interest. Meanwhile, the field of computer security has been represented in academic life in the past dozen years by short chapters in textbooks on operating systems, data communications, and databases, and by a few individuals working in isolation in academia. The field currently has only a few widely-circulated archival journals in computer security topics: e.g., *Computers & Security*, *Journal of Cryptology*, and the *Journal of Computer Security*. And the public perception of computer security is shaped[5] by sensationalism such as computer virus scares, stories of 14-year old children breaking into sensitive military systems, and movies such as "The Net" and "Hackers."

Few universities or colleges offer in-depth education in computer security. Until late 1996, there were only three declared, dedicated computer security research centers in degree-granting departments at universities in the United States;[6] in November of 1996, a fourth center came into public existence[7]. When computer security courses are taught, relatively few textbooks on computer security are in use, and several of the most commonly used ones are principally devoted to cryptography (e.g., [4]) or are outdated.

Research in academia is being done by a limited number of faculty at scattered locations working with a few students. What research is being done, in academia or commercially, has traditionally been oriented towards limited military requirements because until recently that is where the major demand has been (and where the funding has been available). The recent trend has been somewhat more open, but is still focused on a few narrow areas such as cryptographic support for electronic commerce and network firewalls. Although these technologies are significant, they are not addressing more important security needs. By way of illustration, Professor Spafford has been using the following analogy in his lectures and seminars on this topic over the past few years:

> Focusing our research on cryptographic protocols for secure electronic commerce is akin to investing all our money to build heavily armored cars. However, those armored cars will spend their lifetimes transferring checks written in crayon by people on park benches to merchants doing business in cardboard boxes under highway overpasses. Meanwhile, there are no traffic regulations, anyone on a skateboard can change the traffic lights with a screwdriver, and there are no police.

This lack of visibility, training, and coordinated research efforts has led to a significant shortage of professionals trained in *practical* computing security, and to a critical shortage of academic faculty prepared to

[5] Warped?

[6] In addition to **COAST**, these are the Computer Security Research Lab at at UC Davis and the Center for Secure Information Systems (CSIS) at George Mason University. The **COAST** group at Purdue is the largest of the three, and is certainly the one with the broadest focus.

[7] The Center for Cryptography, Computer, and Network Security at the University of Wisconson-Milwaukee

offer advanced instruction in this area. This contributes to a lack of consideration of security issues when new computer systems are being designed, thus placing those new systems at risk. As technology propels us into a future where global networks of communicating, multi-vendor computer systems are commonplace, the lack of universally-accepted social norms and laws will lead to difficulties that only well-designed computer security tools and techniques may prevent.[8] To design those tools and train that workforce, we need an experienced, well-educated core of faculty with support for their students, research, and education programs.

Education and research in computer security-related issues has usually been conducted under a number of different rubrics reflecting its cross-disciplinary nature. Work in areas such as computer architecture, operating systems, data communications, database systems, and software engineering has addressed questions of computer security. Despite advances in all these areas, most direct security-related research in the last few decades has been largely directed towards only a few selected topics. For instance, most of the systems-oriented research done to date has been in support of formal trust models for multi-level secure machines employed in military settings, including compartmented-mode workstations. The results of this research is usually of little use in "real-world" computing environments. This is because the traditional focus of such research has primarily been focused on issues of confidentiality [28, 29] (keeping information secret), rather than on related issues such as availability and integrity.[9] Thus, there has been little support for research in the area of designing security tools and techniques for everyday use on commercial and educational computing platforms. Furthermore, as more computer users seek to use COTS (commercial, off-the-shelf) components, we will need better protection methods built in to these common systems.

In particular, considerable research in computer security methods and protocols over the last few decades has largely been focused on theoretical models of secure systems, multi-level (military) systems, covert channels, statistical intrusion detection systems, and communications security issues (e.g., cryptography). Insufficient research has been focused on the development of tools for improving general security, policy formation, audit techniques, availability models, network security, computer forensics, countering malicious software (e.g., computer viruses and worms), reliability, authentication and integrity methods (to name a few). In fact, research in many of these important areas has been discouraged by the government for fear that people might collaterally discover ways of penetrating sensitive systems. Another reason work in these areas has been limited may be because such efforts require an interdisciplinary approach and few researchers and research groups have both the breadth and depth of expertise necessary to conduct such investigation. To conduct good research in this area with application potential requires a broad base of resources and focus.

Education and research tend to track sources of demand. Thus, over the past few decades, research funding was made available by the military to researchers to conduct research issues related to military concerns. This tended to narrow the research done in computing security. Journals and conferences came into being to provide outlets for this research, thus leading to a climate that has not readily accomodated research in other areas. The demand for students also shaped this picture, as the majority of job offers for graduates in security has come from either the government itself, from military contractors, or from vendors supplying the military. The overall demand for such graduates was not large. The Internet "explosion" has

[8] This is not to imply in any way that development of new network-based etiquette and laws will obviate the need for information security professionals!

[9] During the Cold War era, standard military computer security doctrine could be interpreted as allowing classified computers to be blown up, the users shot, and the surrounding building burned to the ground so long as the data contained therein was not disclosed to enemy agents: policies such as these are not currently acceptable in most banks, universities, or commercial establishments.

taken many in the community by surprise, to put it mildly.

As a result of these influences, education in computer and network security in the U.S. (and much of the rest of the world) is currently provided in a narrow, haphazard and inconsistent fashion. Some standard undergraduate and graduate texts in major course areas (e.g., operating systems) may have a brief chapter on security. These chapters often contain vague information about general security properties that are not particularly helpful in actual use. The instructors have not had direct experience or education in security, so they are unable to augment the material in the texts in any meaningful ways. The result, in the usual case, is that the material is presented in a cursory and compressed manner. As the material is in a separate chapter rather than integrated into the rest of the text, students are further given the implicit impression that security is unimportant, is lacking in detail, and is a separable concern.

Luckily, this is not true at every college and university. There are a number of faculty with some deeper background and concern with security. These faculty members do attempt to present information security concepts at greater depth in their courses. Even so, few students are given the opportunity to concentrate in security as a specialty, or to see how it cuts across several areas of study. There are only a few score faculty at institutions in the U.S. who conduct some research or specialized education in computer or network security. There are fewer still who have any experience with front-line security response experience.[10]

At the high-end of this specialization, the **COAST** Laboratory is generally acknowledged as one of the best places in the world to study practical computer security; it is also located in a highly-ranked computer sciences department, according to statistics published by the National Research Council[11]. It is one of the few academic centers in areas related to computer and network security in the U.S. with several faculty whose research specialization is in one or more fields of information security. **COAST** has outside funding, recognition by its home university as a center of education and research, and wide-spread recognition in the community. It is also associated with a functioning computer response team: the PCERT — the Purdue Computer Emergency Response team.

2 COAST Goals and Recent Research

The mission of the **COAST** (**C**omputer **O**perations **A**udit and **S**ecurity **T**echnology) Project and Laboratory is to conduct research and education on general and practical tools and techniques for improving computer and network security. The specific focus of this research is on typical computing environments — systems without multi-level requirements, and without formal levels of trust. In particular, our short-term research is directed to developing approaches of increasing the security of existing systems without severely impacting their usability. Our goal is to explore how to increase confidence in existing systems in a cost-effective and user-friendly manner. Our long-term research is directed to how to integrate better security mechanisms into common computing platforms. Using this research as a teaching mechamism, we are committed to providing a comprehensive and thorough education in security to our students at every level.

[10] For instance, Professor Spafford at Purdue is the only full-time professor in the world to be associated with a FIRST-accredited response team in a day-to-day, active role; there is no incentive within traditional academia to play such a role, as it is unlikely to lead to publications or grants. Ludicrously, this situation is akin to rewarding teaching faculty at medical schools only if they never have seen a patient or perform an autopsy! (FIRST is the international Forum of Incident Response and Security Teams — the network of CERT teams.)

[11] In their study *Research-Doctorate Programs in the United States: Continuity and Change.*

COAST was established as a formal structure in the spring of 1992 by Professor Eugene Spafford after several years of independent research work. Operationally,**COAST** brings together expertise of many faculty from throughout the university environment, provides shared resources in computer security research, and provides a unified approach to the research and education efforts in this vital area. It provides a focal point both for internal and external agencies seeking reliable information about computer and network security, computer crime investigation, and appropriate computer use.

The specific, long-term goals of **COAST** are to have it continue to be:

- A world-recognized center of research excellence. We intend to be known for our research into methods of practical computer security technology, including computer incident response, system management, and network security technologies. We expect most of our research to be based on the real needs of the community, as conveyed to us through interactions with our sponsors and the general user population. **COAST** is already known world-wide, and we intend to build our existing reputation.
- An on-going source of quality graduates with cutting-edge training in computer and network security. We expect our undergraduate and graduate students to receive a broad-based and comprehensive education that will give them a solid foundation for work in computer security, computer systems, and communication networks.
- A resource center for research. We intend to build a comprehensive collection of documents, references, tools, hardware, software, testbeds, and other resources necessary for comprehensive research and experimentation in various areas of computer security. We expect to make the**COAST** Research Laboratory a significant, widely-available resource for visiting scholars, sponsor personnel, and **COAST** researchers.
- A renowned source of educational and training materials in computer and network security. We intend to produce materials for use in computer security training, both for in-service training in government and industry, and for academic use. This includes traditional materials such as texts and lab materials, but may also include leading-edge technology as embodied in hypermedia and distance-learning methodologies.
- A resource center for independent evaluation of products. We intend to be able to provide unbiased, comprehensive testing and evaluation of security tools for computers and networks. By providing detailed test results to sponsors, vendors, and the general user population, we believe we will help improve the overall state of information system security and improve the general state of the art.
- A resource center for information dissemination to the non-technical community. There is a significant need for sources of information for the press and public that is unbiased by commercial interest or government policies. We expect to continue to be known and consulted as one such source. (**COAST** personnel have been quoted on issues of computer security and computer crime over 150 times in the last five years, including quotations in the*New York Times*, *Newsweek*, the *Wall St. Journal, NPR Radio, ABC Radio, Scientific American*, *Chronicle of Higher Education*, *Science* and more.)
- A source of useful tools for system management and security. Although not a primary focus of **COAST**, we expect that we will produce new tools and protocols as useful byproducts of our research activity that will be of wide-spread applicability to the community at large. The**COAST** on-line archive is already acknowledged as the single largest and most comprehensive security repository on the Internet.

Significantly, **COAST** has already made progress in each of these areas. The following text details a few significant results.

Since 1987, **COAST** faculty and students have been exploring issues in *practical* computer security. Their work has included widely-cited work in analysis of malicious code including viruses (e.g., [11, 10, 12, 13, 14]). In 1991, **COAST** director Gene Spafford coauthored the award-winning book *Practical UNIX and Internet Security* [17], now considered the standard reference in the field. He has also been involved in work on static audit and analysis tools. An initial result of this work was the COPS [5] security audit tool for UNIX systems, used worldwide on tens of thousands of computer systems. This tool runs on several dozen varieties of UNIX, and detects scores of configuration and management problems that may lead to security problems. It is still viewed by many as the standard of its type.

Other research in the center has been conducted on integrity monitoring methods for virus protection, intrusion detection, and change management. One of the first results of this work was a portable integrity checking tool to search for unauthorized changes to files, as might be made by the addition of a trapdoor, logic bomb or virus: *Tripwire*®, (currently in release, version 1.2[12])[20, 22, 21]; *Tripwire*[13] has recently been licensed to Visual Computing Corporation for commercial support and enhancement. We have also expended some effort on developing an efficient scanner tool that can be configured to search for indicated bit patterns (e.g., a virus signature); this reconfigurable virus-scanning tool was described in [23].

Two current projects involve audit and supervision concerns. One project, IDIOT,[14] was funded by the Department of Defense. It employs new methods of specifying and detecting anomalous behavior and exceptional events. Unlike a simple intrusion detection mechanism, this work is intend to serve as a general-purpose monitor for all forms of notable behavior without placing undue load on the system being monitored. The IDIOT protoype was developed as part of Sandeep Kumar's Ph.D. work. It was recently enhanced for release to **COAST** sponsors and other users. A related project is examining how to monitor host and network activity in real time and react to designated traffic. The focus of this work is how to scale the monitoring in a cost-effective manner by using small, largely-autonomous monitoring agents.

Other current or pending **COAST** projects are tools for network and host audit; next-generation network firewall technology on ATM networks; new methods of testing software to discover potential security vulnerabilities; methods of secure distribution of vulnerability information; analysis of security flaws; development of mechanisms to counter denial of service attacks; secure outsourcing of large computations; digital watermarking techniques; detection of insider misuse of computer systems; development of incident response and investigation tools; technology for testing firewall devices; and developing a classification of audit information to separate anomaly detection, accounting, and record-keeping functions.

COAST continues to maintain the largest and most comprehensive on-line archive[15] of computer security tools and documents on the Internet. This archive mirrors tools and documentation held on scores of sites around the Internet, and is organized according to topic. The server has been mirrored to sites throughout the world, and regularly sees downloads in excess of 100 megabytes per day. The archive also includes a WWW interface.

Overall, the **COAST** philosophy is to conduct coordinated, cooperative research in the area of security tools and methods. Uniquely, a primary facet of our work is to investigate methods of developing and

[12] ftp://coast.cs.purdue.edu/pub/COAST/Tripwire/tripwire-1.2.tar.Z

[13] Tripwire is a registered trademark of the Purdue Research Foundation

[14] Intrusion Detection In Our Time.

[15] ftp://coast.cs.purdue.edu/pub

employing practical security tools and techniques that are of little or no threat to systems. This research is thus directed towards development of simple approaches and reconfigurable tools that do not require extreme secrecy to be effective, and that cannot be easily used for system penetration, nor be easily altered to introduce "trojan horses." This work is generating new insights into the development of secure software.

COAST research will continue to focus on practical application of the techniques developed, with experimental trials in existing systems, at Purdue and elsewhere, to validate the methods used. We encourage our sponsors will participate in these trials, to provide us with the kind of real-world constraints and environments that are too often not present in the experiments conducted by other academic researchers.

As noted above, another explicit philosophy of the **COAST** group is to provide students with the opportunity to be involved with computer security issues. By providing research and educational opportunities, we hope to encourage good students to specialize in the area. We also hope to use these results to develop new curricular materials in computer security. By encouraging participation of **COAST** sponsors, we further intend to widen the base and reach of the computer security educational materials that we expect to produce. The result should be students better equipped to undertake careers in industry and academe with computer security as their specialty.

3 Sponsors

Companies and government agencies may become sponsors of the **COAST** project and laboratory. Sponsors will help determine the direction of **COAST** research, through regular feedback, and through an advisory panel. Sponsors will receive early releases of prototype software and our reports, whenever practical. We also anticipate other interactions with sponsors, including placing students for internships, personnel exchanges, and tutorials. The nature of the research proposed for **COAST** is such that active collaboration of sponsors will be sought to influence, validate and refine **COAST** products.

Sponsors will be recognized in **COAST** publications and products (unless they request otherwise). Sponsors will also have preferential access to faculty, staff and students involved in **COAST** projects. Sponsors will have opportunity to provide significant input into the direction of our research. At the least, we anticipate the following as significant benefits of sponsorship:

- Sponsors will gain early access to new computer security technology.
- Sponsors will gain a means to influence and guide cutting-edge research in computer security at an early stage.
- Sponsors will gain preferential access to students working in **COAST**. This may give an "inside track" to hiring talented new employees with experience in computer security research.
- Sponsors will have a means of helping influence security in commercial products through **COAST** development, and through contact with other **COAST** sponsors.
- Sponsors will gain visibility in **COAST** activities, and publications.
- Sponsors gain preferential access for product evaluation, bug fixes, comparison testing, etc. done by **COAST** personnel.

- Sponsors may gain early access to warnings of security problems and trends identified by project personnel.
- Sponsors will get early access to prototype tools and technology to be used as a basis for further development for both internal use and for use by sponsor "clients."
- Sponsors gain access to a pool of expertise that can be tapped to help deal with unexpected problems that might not be covered by in-house expertise.
- Sponsors will help improve the general posture of security in the computing community by supporting **COAST** efforts.
- Sponsors will be able to obtain educational materials on security.
- Sponsors will gain early access to technical reports, software prototypes, and other products of **COAST**.
- Sponsors will get automatic access to seminars, workshops, classes, and other activities to be sponsored by **COAST**.

We are interested in joining forces with additional sponsors for **COAST**. We prefer a steady, multi-year commitment of support from each sponsor rather than one-time, large grants; most of the projects planned will be multi-year in scope. Single donations for specific projects and purposes are still appropriate, however, and constitute sponsorship.

Companies and agencies may also become contributors to (instead of sponsors of) **COAST**. This will entail a smaller donation or gift to the project, and will not result in our same levels of support or disclosure as for sponsors. However, we will seek to involve and support **COAST** contributors more closely than non-contributors. Gifts at the level of $50,000 or above per year constitutes full sponsorship; sponsored project research at the level of $100,000 and above will also qualify for sponsorship.

Sponsorship of **COAST** may be in one of four ways:

1. **By unrestricted gift**. A sponsor may make an unrestricted gift of money to Purdue University, on behalf of the **COAST** Project and its director. This may then be used to support worthy students, fund publication and software distribution, send **COAST** personnel to appropriate conferences, and purchase equipment and software for the lab. This is the prefered form of support: gift/grant money is subject to *no* overhead charge by Purdue.[16] Grants and gifts of money and/or equipment may have certain Federal and state tax advantages for the donor, too.

   (NB: As of June 1, 1996, support of COAST no longer qualifies for a discount in becoming a full corporate partner in the CS Department's corporate partner program[17]. We continue to encourage **COAST** sponsors to become departmental sponsors, too, to enjoy the associated benefits of that program.)

2. **By directed contract**. A sponsor may specifically fund one or more particular projects within **COAST**, or fund the project as a whole. This involves negotiating a contract with the Purdue Research Foundation for specific deliverables (e.g., reports or software), and would need to include

[16] Other forms of support involve a university overhead charge of more than 52%.
[17] http://www.cs.purdue.edu/corp

coverage of salaries, equipment, travel, supplies, secretarial expenses, and overhead costs; in general, over 1/3 of every research contract goes to these fixed overhead costs, which is why a directed contract requires a higher level of funding to qualify for support.

To establish a directed project with **COAST** requires a faculty member to agree to direct that project. In general, this means it needs to be related to existing research of that professor, and yield publishable results for the associated faculty and students. Because of the basic philosophy behind **COAST**, and because of the university mandate for publication of research results, offers of contracts for proprietary or classified research are not solicited.

3. **By endowment**. A sponsor may wish to endow multi-year, continuing scholarship support to graduate students (or undergraduates) working on **COAST**-related projects. These fellowships may be named and designated largely as the sponsor wishes, but must meet university guidelines and be directed in support of students involved with **COAST**-related projects. Sponsorships involving a periodic on-site (sponsor location) internship are possible and encouraged.

   Offers to endow one or more named faculty chairs in information security are also welcomed and encouraged. In addition to assisting in the hiring and retention of excellent faculty, such an endowment also helps promote to the general public the sponsor's committment to information security.

4. **By equipment and software donation**. One of the goals of the project is to develop and maintain a state-of-the-art laboratory for computer security research and education. This requires an *on-going* procurement of equipment and software. Sponsors may donate *appropriate* equipment and materials to help stock and maintain this lab. Any equipment so donated will largely define the "baseline" systems used for development and distribution of **COAST** products.

   We note, however, departmental policy that assesses a fixed maintenance cost per software architecture supported. This covers installation and support of third-party software, local operational security, backups, printing support, and other necessary tasks. The **COAST** group is charged at a rate of $15,000 to $25,000 per year for any software architecture not currently present in quantity within the department. This means that a donation of equipment might be reluctantly declined unless it includes sufficient money for support.

Prospective sponsors are invited to contact Gene Spafford for further information about opportunities to sponsor **COAST** projects and become **COAST** sponsors.

Current sponsors of **COAST** include:[18]

AT&T Labs (GeoPlex)
DARPA
Global Integrity Solutions (an SAIC subsidiary)
MITRE, Inc.
Microsoft Corporation
Schlumberger, Limited
Sprint
Sun Microsystems
U.S. National Security Agency

[18] One current sponsor has requested anonymity.

Current supporters of **COAST** include:

Cisco Systems, Inc.[19]
Internet Security Systems, Inc
O'Reilly & Associates
Thomson Consumer Electronics

Past Sponsors and Contributors include:

Baseline Software, Inc.
Enigma Logic, Inc.
FSA Corporation
GTE Laboratories
Haystack Labs
Hewlett-Packard Company
Hughes Research Laboratories/Hughes Aircraft Company
IBM
InternetOne, Inc.
Nortel (formerly Bell Northern Research)
Raxco
Security Dynamics Corporation
Trident Data Systems
U.S. Air Force Information Warfare Center
Xerox PARC

4 Personnel and Resources

The Department of Computer Sciences at Purdue University is the nation's oldest computer science department (founded in 1962), and has a long tradition of research and service in computer systems, software engineering and computer security. Computer security, in particular, has a long history at Purdue. A graduate course in computer security has been taught here since 1980, after being started by Dorothy Denning when she was at Purdue.

The Department of Computer Sciences is considered by many to be in the top twenty programs overall in computer sciences in the country. The department includes dedicated labs for several major projects and centers, including the SoftLab project, the Software Engineering Research Center (SERC), and the Parallal and Distributed Systems (PADS) group.

The department offers degrees at all levels, including an active PhD program graduating about a dozen new PhDs in computer science each year from an outstanding graduate student population. The program currently enrolls over 750 undergraduates and 175 full-time graduate students. The current faculty of almost three dozen researchers includes internationally-known individuals with interests that include networking, computer systems, databases, algorithms and data structures, user interfaces, graphics and visualization,

[19]Cisco will resume as a **COAST** sponsor on August 1, 1998.

parallel computation, computational theory, software engineering, performance evaluation, artificial intelligence, and robotics. The faculty represent, literally, dozens of international journal editor positions, and more than a score official positions in professional and technical organizations including the ACM, the IEEE and IEEE Computer Society, SIAM, and the Computing Research Association.

Principal Investigator and Director

Gene Spafford attended the School of Information and Computer Sciences (now the College of Computing) at Georgia Institute of Technology, holding both a Georgia Tech President's Fellowship and a National Science Foundation Graduate Fellowship. He received the Ph.D. in 1986 for his design and implementation of the original Clouds reliable, distributed operating system kernel, and for his contributions as one of the original members of the Clouds design team. Next, Dr. Spafford spent a year and a half as a research scientist with the Software Engineering Center at Georgia Tech. His duties there included serving as a principal software engineer with the Mothra software testing project.

In 1987, Professor Spafford joined the academic faculty of the Department of Computer Sciences at Purdue University, where he now holds the rank of full professor. At Purdue, he has taught courses in operating systems, compiler and language design, computer security, computer architecture, software engineering, networking and data communications, and issues of ethics and professional responsibility. Professor Spafford has twice been cited as one of the 10 best undergraduate instructors in the School of Science.

At Purdue he has conducted research on methods of increasing the reliability of computer systems, and the consequences of computer failures. This has involved a particular emphasis on computer security, software testing and debugging, and issues of liability and professional ethics. Spaf has also been an active member of the Software Engineering Research Center (SERC) at Purdue — an NSF-founded Industry-University Cooperative Research Center.

Professor Spafford is currently on the advisory and editorial boards of the the*Journal of Artificial Life*, the journal *Network Security*, *Computers & Security* (as Associate/Academic editor), and the *Journal of Information Systems Security*. He has written extensively in the field of computer security, including coauthoring an award-winning book on UNIX Security ([17]) and making major contributions to widely-cited books on computer viruses ([13]), computer crime ([18]) and WWW security ([6]). He has published over 100 papers and reports on his research. He has also spoken internationally at panels, conferences, symposia, and colloquia on these issues.

Dr. Spafford is a *Fellow* of the Association for Computing Machinery (ACM) and has served as chairman of both that organization's Self-Assessment Committee and of its ISEF Awards Committee, as a charter member of the Technical Standards Committee; he is currently a member of the Committee on U.S. Public Policy (USACM), and is one of the ACM's two members on the Computing Research Association Board of Directors. He is a Senior Member of the IEEE and of the Computer Society of the IEEE; he was named as a charter member of the Computer Society's *Golden Core* for his past service to the society. He is also a member of the Usenix Association. He has been elected to both Sigma Xi and Upsilon Pi Epsilon in recognition of his research activities.

Spaf has served as member of the advisory boards of both FIRST and the CERT/CC; he is currently a member of the advisory board of IBM's Emergency Response Service. He was a founder of the Purdue Computer Emergency Response Team (a FIRST member organization). Spaf is a member and immediate past chair of the IFIP Technical Committee 11 Working Group on Network Security (WG 11.4), and a mem-

ber of Working Group 11.8, Information Security Education and Training.He is a member of the advisory board of the National Research Center on Computing and Society.

Professor Spafford is an alumnus of the fifth class of the Defense Science Study Group and is currently a member of the Information Security Technology and Science Study Group. Over the past few years, he has served in an advisory or consulting capacity on information security and computer crime with several U.S. government agencies and their contractors, including the FBI, National Security Agency, U.S. Attorney's Office, the Secret Service, the Department of Energy, and the U.S. Air Force. He has also been an advisor to several Fortune 500 firms, and state and national law enforcement agencies around the world.

Other information may be found on Professor Spafford's WWW homepage[20].

Other Personnel

In academic 1998, thirty-five students were working on security-related projects with **COAST** faculty. This group included two Fulbright Fellows and students holding several other awards; recent graduates include an NSF Graduate Fellow and a GM Fellow. Many of these students are members of international honor societies such as Phi Beta Kappa, Phi Beta Phi, Upsilon Pi Epsilon and Tau Beta Pi, and most currently hold competitive fellowships or assistantships.

Several Purdue faculty members are also interested in computer security topics. What follows are brief descriptions of a few of the faculty who have expressed strong interest in collaborating on projects as part of the **COAST** effort, or who have been actively involved in **COAST** research. Their interests and areas of expertise help provide additional depth and potential to the **COAST** effort. There are a number of other faculty interested in contributing to **COAST** goals if and when appropriate opportunities arise.

Mikhail Atallah

Mikhail J. Atallah received the Ph.D. in Electrical Engineering and Computer Science from Johns Hopkins University, Baltimore, MD, in 1982. He then joined the faculty of Purdue University, West Lafayette, Indiana, where he is currently Professor of Computer Science. In 1985 he received a Presidential Young Investigator award from the National Science Foundation. In the summer of 1988 he was a Visiting Scientist at the NASA Ames Research Center (RIACS Institute, Center for Advanced Architectures).

Dr. Atallah is a member of the ACM, is a Fellow of the IEEE, and is a member of the Society for Industrial and Applied Mathematics (SIAM). He currently serves on the editorial boards of the journals *Computational Geometry: Theory & Applications*, *Information Processing Letters*, *Int. J. on Computational Geometry & Applications*, *J. of Parallel and Distributed Computing*, *Methods of Logic in Computer Science*, *Parallel Processing Letters*, *SIAM J. on Computing*, and of the Wiley *Handbook of Parallel and Distributed Computing*.

Professor Atallah's main research interests are the design and analysis of combinatorial algorithms, parallel computation, and computational geometry. He has coauthored over 85 technical papers in these areas, e.g.[16, 15] He is currently working with Professor Spafford on the application of special-case pattern matching algorithms with temporal characteristics to intrusion detection systems.

[20] http://www.cs.purdue.edu/people/spaf

Professor Atallah is also interested in audit trail reduction. Audit trail files have a special structure, with many repetitions of a certain kind, so the challenge is to (i) Do better for compressing them than using generally known data compression techniques such as Lempel-Ziv, and (ii) How to search for a pattern in a compressed file without having to "decompress."

More information may be found on Professor Atallah's WWW homepage.[21]

Carla Brodley

Carla Brodley received the Ph.D. in Computer Science in 1994 from the University of Massachusetts. She then joined the faculty of Purdue University, West Lafayette, Indiana, where she is currently an assistant professor in the School of Electrical and Computer Engineering.

Professor Brodley's main research interests are in in the areas of machine learning, pattern recognition and knowledge discovery in databases. She has worked in the areas of anomaly detection (detecting observations that differ from the normal cases), classifier formation (forming a discrete classification system to perform prediction), and feature selection (determining which measurable features are predictive of the output of interest). She has applied techniques from these areas to problems from a variety of fields including computer vision, remote sensing and medical diagnosis.

Professor Brodley is a member of AAAI and IEEE. She currently serves on the editorial board of the journal *Intelligent Data Analysis*, a new (1997) electronic journal.

More information may be found on Professor Brodley's WWW homepage[22]

Tony Hosking

Professor Hosking received the B.Sc. from the University of Adelaide, the M.Sc. from the University of Waikato, and the Ph.D. in 1995 from the University of Massachusetts. His dissertation work in persistent programming languages demonstrated that the fundamental mechanisms of persistence can be supported efficiently on stock hardware, even for very fine-grained data.

Having joined Purdue as an assistant professor in 1995, Professor Hosking is now exploring the advantages of compiler support for high-performance persistent and distributed programming languages and systems [7, 8, 9]. These interests mesh naturally with issues of security in distributed systems, especially for environments that support the distribution and mobility of very fine-grained objects. Maintaining security in such environments can be particularly difficult to achieve without compromising performance.

More information may be found at Professor Hosking's WWW homepage.[23]

John T. Korb

Dr. Korb received his Ph.D. from the University of Arizona in 1979, where he participated in the design and implementation of the Icon programming language. He has held both academic and research positions with

[21] http://www.cs.purdue.edu/people/mja

[22] http://mow.ecn.purdue.edu/~brodley/

[23] http://www.cs.purdue.edu/people/hosking

the Computer Sciences faculty at Purdue in the past dozen years. He is currently the Director of Research Facilities for the department. His research interests include operating systems, programming languages, networks, and user interfaces. He has lectured widely on Internetworking and the X Window System, and is involved with many of the research projects at Purdue.

Supervising the departmental computing facilities, Dr. Korb's responsibilities make him acutely aware of the current problems with practical computer system security, both from technical and administrative points of view. He is keenly interested in performance monitoring and testing tools to detect and correct security problems, including intruders and viruses. Additionally, he is interested in the development of new techniques and tools for managing a multi-vendor, networked installation: currently, he is responsible for almost 400 machines running over a dozen different operating systems.

More information may be found on Dr. Korb's WWW homepage.[24]

Aditya P. Mathur

Professor Mathur is Director of the multi-university NSF-sponsored Software Engineering Research Center. His research interests lie in the area of software testing and reliability. His notable contributions are: techniques for the efficient utilization of parallel machines in software testing, coverage-based approaches to reliability estimation, experimental evaluation of coverage-based testing criteria, a grammar based error classification technique, hardware support for testing embedded software, a new approach to the testing of fault tolerant software, the use of sound in programming environments, and techniques for the use of computer programs as generators of polyphonic music (e.g., [25, 26, 27]).

His most recent work deals with the estimation of reliability of large software systems given the reliability of its components and interfaces. He is collaborating with several notable researchers in statistics to experiment with a 2-phase method for reliability estimation. This method is applicable in the early as well as later phases of the software development cycle. His current research program has taken an integrated view of the issues in reliability, availability, and security. Techniques of formal specification, error classification, testing, and fault injection are being studied to determine how one might estimate and improve the reliability, availability, and security of a software system.

He has developed and delivered a commercial strength compiler, a Parser Generating System, and an educational information system and has supervised the development of three prototype testing and reliability estimation tools. He has taught Computer Science for over 20 years, published over 70 research articles, and has delivered more than 50 invited talks at various forums. He is the author of two celebrated text books on Microprocessor Architecture and Introductory Programming.

More information may be found on Professor Mathur's WWW homepage.[25]

Kihong Park

Kihong Park received his B.A. from Seoul National University, Korea, and his Ph.D. degree in Computer Science from Boston University (1996). Presently, he is an assistant professor in computer science at Purdue University. His research centers around control issues in high-speed networks including congestion control,

[24] http://www.cs.purdue.edu/people/jtk
[25] http://www.cs.purdue.edu/people/apm

quality of service provision, routing and the facilitation of adaptive, fault-tolerant computing on large-scale distributed systems. He has published over 20 technical papers and has served on two international program committees. He was a Presidential University Fellow at Boston University and is a member of several professional societies including ACM and IEEE.

Professor Park's expertise lies in control issues related to managing traffic in high-speed networks and facilitating resilient computing on large-scale distributed systems. His '93 SIGCOMM paper on congestion control showed that achieving both optimality and stability in a generic congestion-susceptible system is intrinsically difficult and he proposed a protocol suite called Warp Control to find near-optimal solutions. In more recent work, Professor Park and his collegues have shown that traffic self-similarity is a robust, instrinsic property of any distributed system where large objects are exchanged with nonnegligible frequency. They also showed the detrimental effect of self-similarity on network performance including quality of service. [34] Professor Park's thesis [33] looked at the theoretical problem of achieving fault-tolerant, adaptive computations on regular, many-body systems called cellular automata, one of the simplest models of distributed computing. He showed that fault-tolerance is not achievable for a class of well-known self-healing controls (dating back two decades) when subject to biased, transient, probabilistic faults.

Professor Park's interest in security lies in two developing areas, one, in the design of efficient mechanisms for the facilitation of prioritized network services (using tagged traffic streams) while meeting security constraints, and two, in the application of rescaling tools and percolation theory for building/analyzing secure large-scale distributed systems using a form of hierarchical intrusion detection by which malicious attacks are "locally" contained and corrected. He is also involved in a**COAST** project, led by Professor Spafford, for systematically testing and evaluating firewalls.

More information may be found on Professor Park's WWW homepage.[26]

John R. Rice

Professor Rice received his Ph.D. in 1959 from the California Institute of Technology. He has been on the faculty at Purdue since 1964, and served as head of the Department of Computer Sciences from 1983–1996. In 1989 he was named as the *W. Brooks Fortune Distinguished Professor of Computer Sciences* Professor Rice is founder of the ACM Transactions on Mathematical Software and is on several other editorial boards. He is the past chair of the Computing Research Association, a fellow of the AAAS, of the ACM, and he is a member of the National Academy of Engineering.

For the past 15 years, Professor Rice has been analyzing numerical methods and problem solving environments for scientific computing. He has created a general methodology for performance evaluation of mathematical software and developed the ELLPACK system for elliptic problems. It is now being extended to Parallel ELLPACK and PDELab. Professor Rice has published 19 books. Among recent ones are*Solving Elliptic Problems with ELLPACK* (Springer-Verlag, 1985), *Mathematical Aspects of Scientific Software* (Springer-Verlag, 1988), and *Expert Systems for Scientific Computing* (North Holland, 1992).

Professor Rice's research in security involves techniques to disguise scientific computations so they can be outsourced (e.g., sent to a supercomputer server) without revealing any data or problem information. These disguises involve a variety of mathematical transformations as well as overlaying data with known but random values.

[26] http://www.cs.purdue.edu/people/park

More information may be found on Professor Rice's WWW homepage.[27]

H. J. Siegel

H. J. Siegel received his Ph.D. degree in 1977 from the Department of Electrical Engineering and Computer Science at Princeton University. In 1976, he joined the faculty of the School of Electrical Engineering at Purdue University, where he is a Professor and Coordinator of the Parallel Processing Laboratory.

Professor Siegel is a Fellow of the IEEE, was a Coeditor-in-Chief of the *Journal of Parallel and Distributed Computing*, and is on the Editorial Boards of the *IEEE Transactions on Parallel and Distributed Systems* and the *IEEE Transactions on Computers*. From 1979 to 1982, he was an *IEEE Computer Society Distinguished Visitor*, giving invited lectures across the country about his research. He is a member of the Eta Kappa Nu electrical engineering honorary society and the Sigma Xi science honorary society.

Professor Siegel has coauthored over 150 technical papers and has presented his work at conferences in the USA, Europe, Japan, and Israel. He wrote the book *Interconnection Networks for Large-Scale Parallel Processing* (second edition 1990) and has coedited five other volumes.

His current research focuses on interconnection networks, heterogeneous computing, and the use and design of the PASM reconfigurable parallel computer system (a prototype of which is supporting active experimentation). He is interested in exploring some of the security challenges posed by parallel machines, and the possible benefit of applying parallel technology to security concerns in more traditional architectures.

More information may be found on Professor Siegel's WWW homepage.[28]

Samuel S. Wagstaff, Jr.

Professor Wagstaff obtained his Ph.D. in mathematics at Cornell University in 1970. He has worked in Computational Number Theory ever since then. For the past dozen years his research has focused on Public Key Cryptography, especially on the computational complexity of factoring integers. While at the University of Georgia in 1981–83, he helped to design and build a special processor for factoring large integers. In 1983, he joined the faculty of the Department of Computer Sciences at Purdue University. He has taught a course in Cryptography and Data Security almost every year since then, as well as teaching courses in several other areas.

Dr. Wagstaff is an Associate Editor of the journal *Advances in the Theory of Computation and Computational Mathematics*. He is coauthor of a popular book on factorization of numbers of the form $b^n \pm 1$[3]. He has given dozens of lectures around the world on Computational Number Theory, and has published extensively in this and related areas (e.g., [38, 39]).

Sam's interests include most areas of computer security. He is working with several the **COAST** projects on matters related to cryptography. His interests mesh especially well with **COAST** projects involving authorization, applying zero-knowledge proofs techniques, and integrity checking methods. One current project, being investigated jointly with Professor Spafford, involves finding mechanisms for the safe distribution of sensitive software patches and upgrades.

[27] http://www.cs.purdue.edu/people/jrr

[28] http://dynamo.ecn.purdue.edu/Faculty/siegel.html

More information may be found on Professor Wagstaff's WWW homepage.[29]

5 For More Information

The current PostScript version of this document. is available for anonymous ftp[30]. A version[31] of that document is also available in Adobe Acrobat PDF. A hypertext, somewhat expanded version of this précis is available as http://www.cs.purdue.edu/coast/precis/precis.html. There is also a WWW homepage[32] for **COAST**. Professor Spafford may also be contacted directly for further information about **COAST**.

References

[1] R.P. Abbott et al. Security Analysis and Enhancements of Computer Operating Systems. Technical Report NBSIR 76-1041, Institute for Computer Science and Technology, National Bureau of Standards, 1976.

[2] Defense Science Board. Report of the task force on information warfare (defense). Government report, November 1996.

[3] John Brillhart, D. H. Lehmer, J. L. Selfridge, Bryant Tuckerman, and S. S. Wagstaff, Jr.. *Factorizations of $b^n \pm 1$, b = 2, 3, 5, 6, 7, 10, 11, 12 up to high powers*, volume 22 of *Contemporary Mathematics*. American Mathematical Society, Providence, RI, second edition, 1988.

[4] Dorothy E. R. Denning. *Cryptography and Data Security*. Addison-Wesley, Reading, MA, 1983.

[5] Daniel Farmer and Eugene H. Spafford. The COPS security checker system. In *Proceedings of the Summer Conference*, Berkeley, CA, June 1990. Usenix. Also available as http://www.cs.purdue.edu/homes/spaf/tech-reps/993.ps.

[6] Simson Garfinkel and Gene Spafford. *Web Security & Commerce*. O'Reilly & Associates, Inc., Cambridge, MA, 1997.

[7] Antony L. Hosking, Eric Brown, and J. Eliot B. Moss. Update logging for persistent programming languages: A comparative performance evaluation. In *Proceedings of the Nineteenth International Conference on Very Large Data Bases*, pages 429–440, Dublin, Ireland, August 1993. Morgan Kaufmann. Also available as ftp://ftp.cs.umass.edu/pub/osl/papers/vldb93.ps.Z.

[8] Antony L. Hosking and J. Eliot B. Moss. Object fault handling for persistent programming languages: A performance evaluation. In *Proceedings of the Conference on Object-Oriented Programming Systems, Languages, and Applications*, pages 288–303, Washington, DC, October 1993. ACM. Also available as ftp://ftp.cs.umass.edu/pub/osl/papers/oopsla93.ps.Z.

[29] http://www.cs.purdue.edu/people/ssw
[30] ftp://coast.cs.purdue.edu/pub/COAST/What_is_COAST.ps
[31] ftp://coast.cs.purdue.edu/pub/COAST/What_is_COAST.pdf
[32] http://www.cs.purdue.edu/coast/coast.html

[9] Antony L. Hosking and J. Eliot B. Moss. Protection traps and alternatives for memory management of an object-oriented language. In *Proceedings of the Fourteenth ACM Symposium on Operating Systems Principles*, pages 106–119, Asheville, North Carolina, December 1993. ACM. Also available as ftp://ftp.cs.umass.edu/pub/osl/papers/sosp93.ps.Z.

[10] Eugene H. Spafford. An analysis of the internet worm. In C. Ghezzi and J. A. McDermid, editors, *Proceedings of the 2nd European Software Engineering Conference*, number 387 in Lecture Notes in Computer Science, pages 446–468. Springer-Verlag, September 1989. Also available as http://www.cs.purdue.edu/homes/spaf/tech-reps/933.ps.

[11] Eugene H. Spafford. The Internet Worm: Crisis and aftermath. *Communications of the ACM*, 32(6):678–687, June 1989. An expanded version is available as http://www.cs.purdue.edu/homes/spaf/tech-reps/823.ps.

[12] Eugene H. Spafford. Computer viruses as artificial life. *Journal of Artificial Life*, 1(3):249–265, 1994. Also available as http://www.cs.purdue.edu/homes/spaf/tech-reps/ALife.ps.

[13] Eugene H. Spafford, Kathleen A. Heaphy, and David J. Ferbrache. *Computer Viruses: Dealing with Electronic Vandalism and Programmed Threats*. ADAPSO, Arlington, VA, 1989.

[14] Eugene H. Spafford and Stephen A. Weeber. Software forensics: Can we track code to its authors? In *Proceedings of the 15th National Computer Security Conference*, pages 641–650, Washington, DC, October 1992. National Institute of Standards and National Computer Security Center. Also available as http://www.cs.purdue.edu/homes/spaf/tech-reps/9210.ps.

[15] M. J. Atallah, P. Jacquet, and W. Szpankowski. A probabilistic approach to pattern matching with mismatches. *Random Structures and Algorithms*, 1994.

[16] M. J. Atallah, C. Lock, D. C. Marinescu, H. J. Siegel, and T. L. Casavant. Models and algorithms for co-scheduling compute-intensive tasks on a network of workstations. *Journal of Parallel and Distributed Computing*, 16:319–327, 1992.

[17] Simson Garfinkel and Gene Spafford. *Practical Unix & Internet Security*. O'Reilly & Associates, Inc., Sebastapol, CA, second edition, 1996. Summary information available as http://www.ora.com/item/pus2.html.

[18] David Icove, Karl Seger, and William VonStorch. *Computer Crime.* O'Reilly & Associates, Inc., 1995. contributing editor Eugene H. Spafford.

[19] Joint Security Commission. Report of the joint commission. Technical report, U.S. Government, 1994.

[20] Gene H. Kim and Eugene H. Spafford. Monitoring file system integrity on unix platforms.*InfoSecurity News*, 4(4):21–22, July 1993. Available as http://www.cs.purdue.edu/homes/spaf/tech-reps/gkim.

[21] Gene H. Kim and Eugene H. Spafford. Experiences with tripwire: Using integrity checkers for intrusion detection. In *Systems Administration, Networking and Security Conference III* Usenix, April 1994. Available as http://www.cs.purdue.edu/homes/spaf/tech-reps/9412.ps.

[22] Gene H. Kim and Eugene H. Spafford. Writing, supporting, and evaluating tripwire: A publically available security tool. In *Proceedings of the Usenix Applications Development Symposium*, Berkeley, CA, 1994. Usenix. Available as http://www.cs.purdue.edu/homes/spaf/tech-reps/9419.ps.

[23] Sandeep Kumar and Eugene H. Spafford. A generic virus scanner in C++. In *Proceedings of the 8th Computer Security Applications Conference*, pages 210–219, Los Alamitos CA, December 1992. ACM, IEEE Computer Society. Also available as http://www.cs.purdue.edu/homes/spaf/tech-reps/9262.ps.

[24] Richard Linde. Operating system penetration. In *National Computer Conference*, pages 361–368, 1975.

[25] A. P. Mathur, M. Chen, and V. Rego. Effect of testing techniques on software reliability estimates obtained using a time-domain model. *IEEE Transactions on Reliability*, 44(1):97–103, March 1995.

[26] A. P. Mathur, M. Delamaro, and J. Maldonado. Integration testing using interface mutations. In *Proceedings of the Seventh International Symposium on Software Reliability Engineering*, pages 112–121. IEEE Computer Society Press, 1996.

[27] A. P. Mathur, F. Del Frate, P. Garg, and A.Pasquini. On the correlation between code coverage and software reliability. In *Proceedings of the Sixth International Symposium on Software Reliability Engineering*, pages 124–132. IEEE Press, 1995.

[28] National Computer Security Center. Trusted computer system evaluation criteria. Technical Report DoD 5200.28-STD, U.S. Department of Defense, 1985.

[29] National Computer Security Center. Computer security subsystem interpretation of trusted computer system evaluation criteria. Technical Report NCSC-TG-009, U.S. Department of Defense, 1988.

[30] Peter G. Neumann. *Computer-Related Risks*. Addison-Wesley, 1995.

[31] President's Commission on Critical Infrastructure Protection. *Critical Foundations: Protecting America's Infrastructures*. U.S. Government, 1998.

[32] Information security and privacy in network environments. U.S. Office of Technology Assessment report, September 1994.

[33] Kihong Park. *Ergodicity and mixing rate in one-dimensional cellular automata* PhD thesis, Boston University, 1996.

[34] Kihong Park. Self-organized multi-class QoS provision for ABR traffic in ATM networks. In *Proc. 15th IEEE International Phoenix Conference on Computers and Communications*, pages 446–453, 1996.

[35] Richard Power. Current and future danger. Technical report, Computer Security Institute, San Francisco, CA, 1995.

[36] Richard Power. Current and future danger. Technical report, Computer Security Institute, San Francisco, CA, 1996. Second Edition.

[37] Peter H. Salus. *Casting the Net: From ARPANET to INTERNET and Beyond*. Addison-Wesley, Reading, MA, 1995.

[38] J. W. Smith and S. S. Wagstaff, Jr.. How to crack an RSA cryptosystem. *Congressus Numerantium*, 40:367–373, 1983.

[39] J. W. Smith and S. S. Wagstaff, Jr.. Methods of factoring large integers. In D. V. Chudnovsky, G. V. Chudnovsky, H. Cohn, and M. B. Nathanson, editors, *Number Theory, New York, 1984-85*, volume 1240 of *Lecture Notes in Mathematics*, pages 225–234. Springer-Verlag, 1987.

[40] National Research Council System Security Study Committee. *Computers at Risk: Safe Computing in the Information Age*. National Academy Press, 1991.