Int. J. Engng Ed. Vol. 19, No. 1, pp. 206–212, 2003
Printed in Great Britain.

0949-149X/91 $3.00+0.00
© 2003 TEMPUS Publications.

Vulnerabilities and Failures of Complex Systems*

S. J. LUKASIK
Sam Nunn School of International Affairs, Georgia Institute of Technology, Atlanta, GA 30332, USA.
E-mail: stephen.j.lukasik@saic.com

The generic product of engineering is a system. Systems interact with each other as a result of the choices of their designers, owners, and users. These complex systems of systems fail in unanticipated ways. The impacts of those system failures are amplified by the often unplanned and unappreciated interdependencies among systems. The result is an increasing frequency and magnitude of system failures that have major impacts on regional economies and on the physical well-being of the populations they serve. Compounding this concern is the emergence of malevolent acts resulting from irresponsibility, disaffected employees and users, criminal motives, terrorism, and state-supported strategic attack. This paper first examines some vulnerabilities encountered in the 'federation' of systems through the widespread application of information technology. It proposes a defensive design paradigm that recognizes the unavoidable occurrence of failures resulting from complexity and from malice. Finally, the implications for engineering design are examined and proposals are made for ways to introduce such an approach to design into curricula.

INTRODUCTION

ENGINEERS design systems and, rather obviously, the designs include the necessary controls to enable them to operate as desired under the range of conditions they encounter. Controls have changed with the level of technology employed. When systems were purely mechanical, with either manual or water power for their operation, sensors were eyeballs and controls were mechanical. When electric power was introduced in the late 19th century, it was possible to employ analog electrical sensors, and control could be either manual or electrical. When digital electronic technology became available in mid-20th century, the analog outputs of system status sensors were converted to digital form and the control systems involved algorithmic computation based on models of system performance implemented in software.

With the addition of long distance communication capabilities to digital computation, the technology enabled the creation of *networks*, networks not only for the delivery of service but for the control of the network itself. Two (pre-digital) examples of network control are the use of the rails for the transmission of block occupancy status information in railroad signaling systems, and the use of the telephone network to not only carry calls, but set up the call circuit.

Networks have become both the triumph and the bane of contemporary society. They enable the central control of distributed elements, provide global backup for local failures, and increase efficiency by allowing system optimization over larger geographical and functional domains. The most prominent example is the ensemble of TCP/IP-based information networks we call the Internet.

Networks are described by their architecture, and by the principles incorporated into their architecture such as being heterogeneous, asynchronous, and open. While initially allowing the central control of distributed resources, the same efficiency arguments led naturally to further efficiencies through distributed control of distributed resources.

Distributed systems display some characteristics that are not desirable. Such systems are complex, to the point where no single person or organization is fully cognizant of all their parts. Configuration management becomes increasingly difficult as their complexity increases. Much as one might prefer not to admit it, no one really controls them. The consequence of this is that no single organization can fix them when they fail. And by the nature of things, as captured by Murphy's Law, they will fail [1].

Another awkward fact of modern technology is that the ultimate control technology for the networked world is software. The good news is that this is a rapidly changing technology, increasing in functionality and efficiency and benefiting from the concurrent increases in the power of the computing hardware it controls. The bad news is that old software, euphemistically called 'legacy' software, is so archaic that its creators have vanished and it is no longer supported by contemporary hardware and software environments. The scramble for COBOL programmers to remedy past sins based on the use of two-digit years is a recent case in point.

* Accepted 2 September 2002.

To all this, one can respond 'So?' Are not these simply the consequences, some admittedly unintended, of progress? To a degree, yes. Except for the factor, previously largely ignored, of *malice*. As long as the operation of a system depends on a few skilled people, malice can be controlled. Consider the case of the financial system of banks that handle cash. Bank employees are limited in number and they can be vetted. To limit embezzling there is a system of auditors, both internal and external. And to control physical theft there are bank guards and vaults. The system works because there are relatively few unprincipled people skilled enough to attempt embezzling or safe-cracking or brave enough to engage in armed robbery. It also helps that the skilled operators and skilled protectors outnumber the 'bad guys'.

Electronic network technology, enabled by powerful software, has changed all that. It has introduced vulnerabilities into the systems on which society depends for its everyday functioning. These vulnerabilities have been exploited to date by several types of people.

EXPLOITING THE VULNERABILITIES OF NETWORKS

Hackers are at the low end of the scale of maliciousness. They break into systems or deface websites for fun, to show off, or for the challenge of problem-solving. Frequently they are young, and psychologists explain that their ethical sense and appreciation of consequences is not yet fully developed. On the scale of skill, they can be quite amateur, but they are greatly assisted by hacking tools created and distributed by people who are beyond the age of hiding behind such psychological cover. The February 2000 distributed denial of service attacks have also been traced to teenagers, as were the attacks on DoD computers in 1997. But the software tools used were written by a skilled programmer who calls himself 'Mixter'. On 7 Feb 2000 two websites were subjected to a distributed denial of service attack. To mount such an attack, the attacker secures access to a number of unprotected computers and instructs them to send a large number of messages to the target website, either requesting information and hence saturating the target's input capacity or transmitting invalid information that causes the target site to crash. The first attack was on Yahoo at 1:10 p.m. EST and shut the site down for 5 hr. Yahoo is visited by 8.7 million users per day and it is an important part of the Internet because it serves as a portal site used to locate other sites or information on the Internet. At 2 p.m. on the same day, Buy.com was attacked and closed down for 6 hr. This site is an e-commerce sales site visited by 122,000 users per day. On 8 Feb Amazon.com, a retail sales site visited by 892,000 users per day was closed for 3.75 hr; the CNN news site, with 642,000 users per day was closed for 3.5 hr; and the eBay auction site, with 1.68 million users per day was closed for 5 hr. This pattern was repeated on 9 Feb. The E*Trade brokerage site, with 183,000 users per day was closed for 2.75 hr and ZDNet, with 734,000 users per day closed for 3.25 hr.

Another group of attackers, the virus writers also impose substantial costs on society. Melissa, Love Bug, and other viruses have cost businesses, in terms of lost time or lost e-commerce sales, billions of dollars [2]. The article reports on the time sequence of events following the first appearance of the ILOVEYOU virus on 4 May 00. The effect of the virus on a large e-mail system was noted by the system operators within 2 hours. The ISP was brought in and diagnosed the problem within a few minutes and antivirus patches were made available immediately. Nevertheless, the virus caused damages exceeding $1.5 billion in 48 hours. Another estimate of the damage is as much as $10 billion [3]. A perverse nature of the problem is illustrated by the fact that the necessary antivirus updates frequently exceeded 1.5 MB and the Internet itself became clogged by the volume of the updates. The writer of the virus, a Philippine computer science student, was identified but could not be prosecuted because no Philippine laws were violated. In the case of the Melissa virus released on 26 Mar 99, the damage was stipulated in a federal plea by the writer, a 30-year-old programmer, as exceeding $80 million. Mutations of the Melissa virus continue to appear. The AnnaKournikova virus that appeared 12 Feb 01 was noteworthy since it was written by a Dutch hacker using a virus 'toolkit', the Visual Basic Worm Generator that required no knowledge of computer programming. The virus infected e-mail systems for millions of users worldwide [4].

The next step up on the scale of maliciousness are those protesting government policies and intend harm in order to be recognized. They are persistent, organized, and are motivated by a larger purpose than simply individual entertainment. A recent UK law includes them in the category of terrorists. The UK Terrorism Act 2000 defines terrorism to include actions that 'seriously interfere with or seriously interrupt an electronic system'. The Act only applies to actions 'designed to influence the government or intimidate the public' [5].

The exploitation of networks for criminal purposes is a next stage of malice. Its occurrence is well established and losses from it are mounting. While crime committed with the aid of a computer is not fundamentally new, the numbers are astonishing. While the average bank robbery at gunpoint yields $9,000, and ordinary commercial embezzlement $25,000, the average computer-assisted theft yields $650,000 (Bob Friele, US Secret Service Financial Crimes Division [6]). Trends in criminal activity are reported in an annual survey of businesses undertaken jointly by the FBI and the Computer Security Institute [7].

Losses due to criminal acts include both financial fraud and theft of intellectual property. These and other abuses of information networks are discussed in more detail elsewhere [8].

The above suggests that violations of computer systems are not only becoming more costly to the victim but more lucrative for the criminal. Furthermore, the number of people who can acquire the necessary skills to exploit information systems is growing as the number of computer-literate people grows, and as the power of the attack tools available to them increases.

Finally, consider those intent on strategic attacks on a nation's infrastructure. If damage such as that noted can be accomplished by unskilled or relatively low skill people largely intent on fun or private gain, what might highly skilled attackers, supported by the financial and intelligence resources of an adversary state, be able to accomplish? National security analysts believe the answer to be a great risk of disruption and loss of life and property. The National Security Advisor, Condoleezza Rice, in the first major policy address by a spokesperson of the Bush Administration on the protection of infrastructures against cyber attack, noted on 22 Mar 01, 'Corrupt those networks and you disrupt this nation. US businesses, which own and operate more than 90% of the nation's banks, electric power plants, transportation systems, telecommunications networks, and other critical systems, must be as prepared as the government for the possibility of a debilitating attack in cyberspace.'

Compounding the vulnerability of such systems is their interdependencies, with the result that impacts of attacks on one system can cascade into other systems. A particularly insightful study was that undertaken for the UK Cabinet Office [9]. UK infrastructures were studied by Ernst & Young for the Cabinet Office, as part of the Y2K remediation effort. The study was directed to 11 infrastructures: fuel, utilities, transport, finance, supply of food and goods, communication, emergency services, social services, justice, health services, and weather services. These infrastructures were decomposed into 59 'processes', each of which was modeled to identify the generic actions required for its operation. It was thus possible to identify for each process what other processes it depended on. The result is a 59 59 element dependency matrix that shows, for the *ij* cell, whether process *j* depends critically on process *i*. The matrix also records where there is a non-critical dependence, one that would be detrimental to process *j* even though it might not result in its complete breakdown.

This analysis enables one to identify as the most critical processes those on which the greatest number of other processes depend. Table 1 shows the most important 12 of the 59 processes. Each process is characterized by three numbers: C is the number of critical dependencies of other processes on that process; N is the number of non-critical dependencies of other processes on that process; and T is the sum of C and N. There are four processes in the most critical category, defined as those for which T > 40, and eight in the next tier of criticality, defined as 11 < T < 30.

Not surprisingly, telecommunications and electric power are the most critical, with virtually all other processes of society dependent on these two. The supply of transport fuel and the road infrastructure rank next since most material goods move from producer to consumer by road. At the next tier are such systems as the supply of water, gas, the movement of funds, the provision of emergency services and the like. These are the central systems and hence these are the systems a state-supported attacker can be expected to target.

IMPLICATIONS FOR SYSTEM DESIGN

Designers of information systems, and the subsystems and components of the systems that incorporate information technology, must explicitly recognize from the beginning that such systems are being penetrated at an alarming and growing rate. The view that users are benign, that there are only a few 'bad apples', and that system failure is the result of infrequent random accidents must be abandoned as a premise in system design. The penetration of information technology into society makes large numbers of people capable of attacking those systems and the systems, as currently designed and operated, are so porous that failure due to malicious action is guaranteed. Not only are users, their personal and proprietary information, and their liability as system operators at risk, but the vulnerabilities of systems dependent on information technology puts the security of the nation and its economy at risk also.

While not a cyber attack, the exploitation of the air transport system by terrorists on September 11, 2001 is a case in point. Air travel presumes that passengers are interested in arriving at their destination and airliners are designed to be operated as a bus. Passengers carrying box cutters who take over the controls and turn the aircraft into a guided cruise missile were not part of the design

Table 1. Infrastructure process interdependencies

Infrastructure Process	C	N	T
Provide telecommunications	49	9	58
Provide electricity	56	1	57
Supply transport fuel	45	4	49
Provide road infrastructure	43	6	49
Supply clean water	26	3	29
Transfer funds	17	3	20
Provide postal service	14	3	17
Supply gas	13	3	16
Manage sanitation and waste disposal	11	4	15
Provide fire and rescue service	11	2	13
Provide weather information	9	10	19
Provide rail transportation	8	11	19

environment. The air traffic control system has not been designed to be able to assume control of the aircraft from the ground nor would the air defense system have been able to intercept the aircraft prior to impact. The only system that worked as it was designed to was the cellular phone system, that enabled passengers on the last plane to achieve 'situation awareness' and thereby overcome the attackers and thwart their plans. To do this, however, the cell phones had to be used while in flight, a violation of another rule of the air transport system.

Two changes in approach are needed if information technology is to be incorporated safely into the systems on which society depends. First, systems must be designed *defensively*. And second, they must be designed for robustness under *planned malicious assault*, not simply random failure.

Defensive design is nothing new in areas involving physical processes. Aircraft engines and canopies are designed to withstand bird impacts, and structures are designed for the one-hundred year storm and magnitude 8 earthquakes. In information systems there seems to be no equivalent stressing design criterion, and such systems are easily overwhelmed by nothing more sophisticated than just driving up the rates at which system resources are accessed.

Malicious assault is a more difficult design criterion. The human mind is ingenious in surmounting obstacles, and the protection one mind can devise can be defeated by another. Ironically, the successful attacker will eventually know more about the design than its designer, because the designer has to deal with the whole while the attacker need only concentrate on finding the hole. Thus bank vaults are penetrated and prisoners escape.

Both attackers and defenders can, and do, learn from experience. Thus design is not a matter of an initial definition of requirements, and entering into a contract to create a system to satisfy those requirements. Instead design is a never-ending cycle of measure and countermeasure. A current management concept is that of the learning organization. In this view, organizations are seen as dynamic. Designs, on the other hand, are too often static. A satisfactory design can, in principle, embrace a capacity for growth, evolution, and adaptability. Curiously, the designs of information systems are frequently imagined as having such features. But as information systems become embedded in organizations, they become critical for operations. And as information systems accumulate layers of legacy software, they become increasingly inflexible and hence vulnerable to attackers who only need to be expert in circumvention and penetration and the exploitation of often known flaws.

A paradigm for defensive design is shown in Fig. 1. This shows that protecting a complex system involves five coupled issues. First, one attempts to influence potential attackers to desist from attacking. Second, if attacked, one seeks to thwart the attack and hence to prevent damage to the system. Third, since one will not always be successful in either preventing or thwarting an attack, one limits the damage as much as possible. Fourth, having sustained some level of damage from an attack, the system must be reconstituted to the pre-attack state. Finally, since both offense and defense are influenced by changing technology and incentives to attack, the final step is for the defender to learn from failure in order to improve performance, just as attackers will learn from their failures. Strategies for the protection of systems based on this schema are elaborated in [11].

There will be trade-offs between the various steps. Preventing or thwarting attacks imply costs, both explicit as well as possibly reduced system performance. The more successful one is in limiting damage, the less will be the amount of damage to be repaired. If limiting damage is difficult, prudence suggests that investments be made to assist in reconstitution. Damage limitation can be viewed on two time scales. One can seek to limit the damage from a single attack, or to minimize losses from multiple attacks over time. There will be trade-offs here also. They will involve

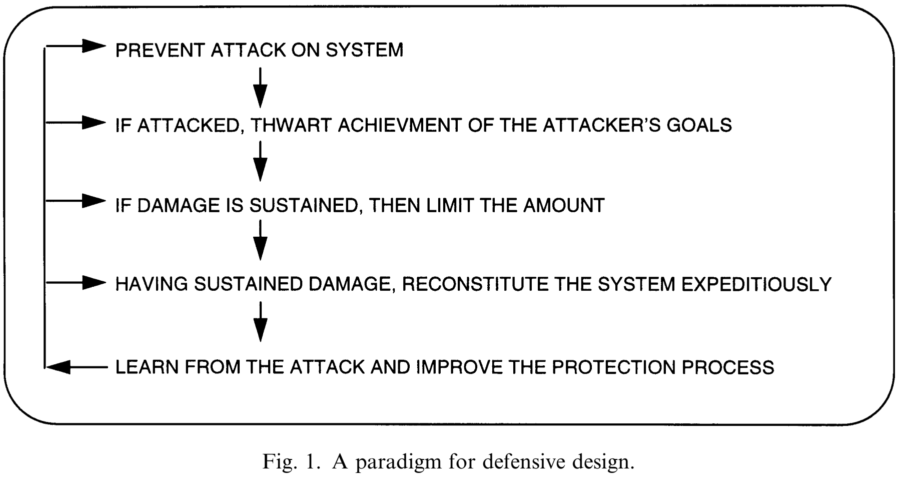
PREVENT ATTACK ON SYSTEM
IF ATTACKED, THWART ACHIEVMENT OF THE ATTACKER'S GOALS
IF DAMAGE IS SUSTAINED, THEN LIMIT THE AMOUNT
HAVING SUSTAINED DAMAGE, RECONSTITUTE THE SYSTEM EXPEDITIOUSLY
LEARN FROM THE ATTACK AND IMPROVE THE PROTECTION PROCESS

Fig. 1. A paradigm for defensive design.

choosing detailed and potentially costly scrutiny of individual transactions, and identifying and punishing attackers over the longer term, recognizing this implies sustaining some amount of damage along the way.

Since complex systems may be a mix of public and private ownership, the various owners are likely to have different views of investments in protection. Private owners, faced with loss of revenue and loss of confidence by customers, regulators, investors, and insurers, will seek to restore revenues and confidence in their stewardship. Governments may pursue policies that focus on longer term aspects of protection, seeking to reduce cumulative losses, protecting markets, and maintaining law and order. Industry and trade groups may adopt both perspectives, the long view for the benefit of the collective group they represent, but reflecting also the individual financial and competitive concerns of their members with respect to successful single attacks.

Thus the system design philosophy that is adopted is likely to represent a balance of short and long term approaches, as well as a balance between levels of investments in prevention, damage limitation, and reconstitution [12].

There are several domains that offer possibly useful models for a defensive design process.

Military systems

Military systems are always a response to a 'threat'. This is a very formal part of the design process, and in the US there is an agency outside the procuring military service that 'validates' the threat. This means that the service can not on its own make up a threat to which it is allegedly responding as a way of securing more financial resources. The threat comes from the intelligence community and is, if the system works right, based on what the adversary is actually doing or is assessed to be capable of doing in the future. Military services organize themselves around 'missions' and the system being designed must demonstrably improve the service's accomplishment of the mission in question. Part of that demonstration is detailed analysis of the adversary systems and the tactics that the new system will confront. Elaborate modeling and simulation, field exercises, and operational tests and evaluation are used to convince Defense management and Congress that the system will prevail in combat. In practice there are lots of ways for this process to succumb to 'threat creep' and to political and bureaucratic gaming. But the essential point for the present discussion is that no system is procured without the most intense study and understanding of resourceful opponents bent on defeating it.

Criminal investigation

The objective of criminals is to make identification, location, and prosecution as difficult as possible. Furthermore, the assumption of innocence and the protections of due process put constitutional limits on criminal investigations and prosecutions. In the present context, this means that the collection of forensic evidence must be done in ways that satisfy strict requirements regarding probable cause for search and seizure; qualifications and training of the collector; the chain of custody of evidence collected; the adequacy of the analysis of the evidence; and the protection of the rights of both the accused and those of victims, bystanders, and witnesses. Computer forensics is a technically and legally intricate specialty. If the owners of systems expect to receive legal protection and the deterrent benefit that can come from successful criminal, or civil, sanctions against violators, provision for that must be recognized in the initial design. Other concepts of civil and criminal law are applicable, including exercising due diligence in protecting property rights and in avoiding liability for civil or criminal negligence.

Collective security

Designers of complex systems can reasonably expect that the operators of the systems they design will work together against common adversaries. Thus sharing of information about the occurrence of attacks, attack modes, and even attackers can help to improve the balance between attackers and defenders. For this to work, the system design must incorporate 'sensors' capable of collecting, analyzing, and, if appropriate, transmitting information on a timescale useful for common defense. The anti-virus community works in such a mode, despite the natural competitiveness of the vendors of security products. Automated network intrusion detection is a current R&D topic, and the Internet Engineering Task Force's working groups are addressing the necessary protocols and other technical standards to support collective efforts. Other concepts developed by the arms control community can be applicable, such as non-aggression pacts, verification of compliance, notification of tests, and consultative procedures for investigating disputes and resolving conflicts. The effectiveness of such approaches can be enhanced if systems are designed in ways that can assist in their implementation.

PEDAGOGICAL PROPOSALS

There are several ways these ideas can be implemented in engineering curricula. Earlier work has addressed the need for presenting systems of systems as a central concept in contemporary engineering education [13]. In another, a proposal for teaching the concept of emergent properties of complex systems based on the complexity and ubiquity of the Internet was offered. I recall that one of the earliest documented pursuits of a cyber criminal resulted from the investigation of a 75-cent anomaly in a computing account [14].

While the central focus of concern here is on information systems or on information-based control subsystems, the idea of defensive design is not simply a matter for computer science departments. While the technical tools to combat cyber attack are the domain of that discipline, balanced systems will require the use of many disciplines: those related to physical vulnerabilities; the application of financial audits; the use of management system design; and the psychology of motivating insiders. The failure of the Tacoma Narrows bridge in 1940 was photographed and considerable data can be extracted from the video record.

Consider the following suggestions as to how the design philosophy discussed here might be presented to engineering students:

1. *A clinic project that requires the analysis of a failure of a complex system.* The failure could be a recent one of concern to the sponsor or it could be a past failure captured in a way that allows thorough *ex post facto* analysis. Even where professional analysis of the failure is available in the literature, the students could be graded on the degree to which they are able to formulate and analyze potential failure processes [15]. The material in [1] also provides rich ore to mine.
2. *A clinic project of the same type as (1) but where the failure is the result of malicious action.* The official investigations of terrorist bombings, as augmented by investigative journalism, usually provides a wealth of documentation in its search for those responsible. The project could examine ways the incident could have been circumvented, and could model the official recommendations to test their adequacy.
3. *Construction of case studies.* Case studies are heavily used in management curricula to illuminate the multi-dimensional issues that arise in business. A point of departure could be the wealth of material in the writings of Petroski [15]. (The material in [1] also provides useful material.) Student projects could be aimed at producing first drafts of such case studies, testing studies for adequacy, and updating completed case studies, as well as using them in courses.
4. *A design project focused on protecting student privacy while online.* The project could also explore ways of assuring the integrity of academic records maintained by the university. It could evaluate institutional privacy policies with the intent of making recommendations on how to protect both the student and the institution.
5. *Student participation on penetration teams to test the vulnerabilities of existing systems.* The systems could be part of clinic projects or they could be the university's own systems. Since students will relish the opportunity to 'get inside' systems, part of the project could be to design safeguards to prevent misuse of the access they are provided and the vulnerabilities they discover. Such exercises could also provide good opportunities to illustrate issues relating to professional ethics.
6. *Student participation as part of the university's Computer Emergency Response Team.* While one can hope that real intrusions will be infrequent, they are likely to be ill-timed in terms of the academic schedule. Forensic data collected as part of actual investigation could be preserved for off-line student exercises.
7. *Seminars, theses, and projects on the history of technology.* Past technologies and their failures could be used to examine their social, economic, and environmental impacts, especially where there were major unanticipated consequences. Such studies could fit into humanities and social science courses as well as in courses devoted to current engineering practice.

Engineers will eventually learn the techniques discussed here through some combination of formal education, perhaps at the graduate level, through professional experiences, and by on-the-job training (OJT). Learning by experience shifts the costs to users who should not be expected to underwrite the education of the designers of the systems they believe to be commodities. OJT shifts the costs to employers, which in one sense is fair, but does not provide the measure of quality control that formal educational institutions provide. Additionally, less-than-adequate attention to OJT responsibilities simply shifts the suffering from failures to users. To the extent that universities shirk the responsibility for teaching defensive design, one can at least hope that professional organizations, licensing bodies, and accreditation agencies will choose to direct attention to the need.

REFERENCES

1. Charles Perrow, *Normal Accidents: Living With High Risk Technologies*, Basic Books (1984).
2. www.mailscan.deerfield.com/helpdesk/mw-art.cfm
3. www.gallup.com/poll/releases/pr000607.asp
4. www.asia.cnn.com/2001/TECH/internet/02/14/kournikova.01/
5. www.zdnet.com.au/news/dailynews/story/0,2000013063,20205108,00.htm
6. www.netrail.net/~sunburst/investigations/fletcc6.htm
7. Richard Power, *Computer Security Trends & Issues*, **V**(1) Winter 1999; **VI**(1) Spring 2000. www.gocsi.com
8. Stephen J. Lukasik, Protecting the global information commons, *Telecommunications Policy*, **24**, 2000, pp. 519–531.

9. www.citu.gov.uk/2000/ey_study/ey_menu.htm
10. J. Stephen, *Lukasik, Public and Private Roles in the Protection of Critical Information-Dependent Infrastructure*, Center for International Security and Cooperation, Stanford University, May 1997.
11. Stephen J. Lukasik, Seymour E. Goodman, and David Longhurst, *Strategies for Protecting National Infrastructures against Cyber Attack*, International Institute for Strategic Studies, London, Adelphi monograph series (in press).
12. Stephen J. Lukasik, Systems, systems of systems, and the education of engineers, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, **12**, 1998, pp. 55–60.
13. Stephen J. Lukasik and Michael Erlinger, Educating Designers of Complex Systems: Information Networks as a Testbed, *Int. J. Eng. Educ.*, **17**, April 2001.
14. Clifford Stoll, *The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage*, Doubleday (1989).
15. Henry Petroski, *To Engineer is Human: The Role of Failure in Successful Design*, Vintage Books, Random House, first edition 1982, vintage edition 1992.

Stephen J. Lukasik received a BS in physics from Rensselaer Polytechnic Institute and an MS and Ph.D. in physics from the Massachusetts Institute of Technology. While a member of the Department of Defense Advanced Research Projects Agency (ARPA), he was responsible for research in support of nuclear test ban negotiations and subsequently served from 1967–1974 as Deputy Director and Director of the Agency. Later government service was as Chief Scientist of the Federal Communications Commission responsible for advising the Commission on technical issues in communication regulation and for the management of non-government use of the electromagnetic spectrum. He has taught physics and engineering at Stevens Institute of Technology, and technology policy at the RAND Graduate Institute and the Pepperdine University School of Business and Management. He has been a Visiting Scholar at the Stanford University Center for International Security and Cooperation, and a Visiting Professor of International Affairs at the Sam Nunn School of International Affairs, Georgia Institute of Technology, where his research was directed to technical and policy issues related to the protection of critical infrastructures against cyber attack. Dr. Lukasik's business and management experience includes positions as Vice President and Manager of the Systems Development Division at the Xerox Corporation, Vice President for National Security Research and Chief Scientist at the RAND Corporation, Vice President and Manager of the Northrop Research and Technology Center, Corporate Vice President for Technology at Northrop, and Vice President for Technology at the TRW Space and Defense Sector. He has served on numerous advisory committees of the federal government, of several universities, and of the National Research Council. He was a founder of the Software Productivity Consortium and served as the Chairman of its Board of Directors in 1988. Dr. Lukasik currently serves as Assistant to the CEO of SAIC participating in programs related to counter-terrorism. He was a founder of *The Information Society: An International Journal* and a member of the Board of Trustees of Harvey Mudd College.