10TH NATIONAL COMPUTER SECURITY CONFERENCE, BALTIMORE, SEPT. 21-24, 1987.

SOCIAL ASPECTS OF COMPUTER SECURITY

Dorothy E. Denning, Peter G. Neumann, and Donn B. Parker

SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025

Introduction

The problem of computer misuse (intentional and accidental) has been a growing concern as the number of computers and users increases, and as computers become an integral element in areas such as medicine, finance, and defense. This concern has led to advances in computer security technology, and to the Department of Defense Trusted Computer System Evaluation Criteria[1], which gives criteria for evaluating the security of computer systems in terms of the policies to be enforced and the assurance one can obtain in the correct enforcement of those policies. The "Criteria" represents a significant step forward in the computer security area.

The objective of this paper is to examine social aspects of computer security, particularly with respect to some of the technologies being developed. We believe that the problem of computer misuse must be addressed within a broader context that includes the people who regulate and use the system, and the information resources that are external to the system. Security policies and mechanisms must be evaluated in terms of their effect on privacy and productivity, and in terms of the actual and perceived threats they address. If we ignore these social aspects, then there is the danger of developing technologies that are not cost effective, do not address the actual threats, or jeopardize human rights.

In an article on system safety, Leveson [2] observes that "Safety is a system problem," and goes on to show that one cannot make systems safe just by focusing on software and hardware issues. Instead, one must examine the total system and its social aspects, including political, legal, and ethical issues.

The same is true of computer security. We must pay greater attention to the issues of user productivity, privacy, ethics, acceptance of security measures, the nature of the threats, and the role of computer security within the broader context of information security.

In the remainder of this paper, we elaborate on four topics relating to the social aspects of computer security: security policy definition and awareness, user productivity, privacy, and the broad area of information security. For each of these topics, we make specific recommendations aimed at improving overall information security.

Security Policy Definition and Awareness

In many environments, information managers and workers lack the knowledge, motivation, and support to apply basic security controls and practices. This is particularly true in business, where there are few written rules about how the computers may be used. It is not surprising that the systems are misused, because the users and their organizations are not clear what the rules are. Also, many users are not consciously aware of how their carelessness can deleteriously affect other users who are sharing the same resources (including computer networks).

There have been many violations of data privacy and integrity, with a wide variety of motivations – personal gain, greed, curiosity, harassment, etc. Documented cases include external system break-ins; internal fraud and embezzlement; implantation of destructive Trojan horses, software time-bombs inserted for blackmail, spoofing, jamming, and so on. Hiding of knowledge about system security vulnerabilities often (e.g., by system purveyors) creates a head-in-the-sand attitude, ripe for underground dissemination of the vulnerabilities (which are usually known anyway) and abuse. Open discussion of such knowledge also creates problems, as it whets the appetites of would-be perpetrators.

The federal computer crime law and 47 state statutes define as crimes unauthorized acts with, within, or to computers. This makes it imperative for computer systems managers to make clear what is unauthorized, such as personal use of electronic mail and other computer resources. All employees should have explicit requirements to protect information assets in their job descriptions and performance evaluation criteria. Adequate motivation to support security will not be achieved until there are well-defined security policies and until security is considered part of one's job, since security can otherwise be viewed as an obstacle to productivity.

In order to assist organizations develop security policies, policy guidelines can be developed for various types of organizations and various degrees of risk. These guidelines could be developed through industry and professional associations, such as the ACM External Activities Board, the IEEE, and the Data Processing Management Association.

The guidelines would suggest possible policies about what is considered to be acceptable use of an organization's information resources, including personal computers. The policy guidelines should address the broad social issues such as user productivity and privacy rights, discussing tradeoffs as they arise. Based on the guidelines, each organization would formulate its own specific policies in accordance with the sensitivity and value of the information (and other resources) to be protected, and the threats, vulnerabilities, and risks.

When a user is given an account on a system, or other information-related responsibility, the user might be asked to read and sign the organization's policy statement. Making the security policy clear, together with asking all users to make a commitment to the policy, could help eliminate much computer misuse (both internally and externally), while at the same time helping the users appreciate the need for security and the benefits to be gained by it, including making them of greater value to their organizations.

Policies for using personal computers can be developed for elementary and secondary schools. Such policies should contain a clear statement that personal computers are not to be used for unauthorized entry into other computer systems (when in doubt, ask for permission). This could help reduce the malicious hacker problem. Since break-ins are often performed more out of challenge than malicious intent, alternative challenges can be presented to the students in the schools.[1]

Overall there is a great absence of pervasive and credible ethical principles; on the other hand, there are many incentives (e.g., weak technology) for violating such a code of ethics, even if it did exist. Nevertheless, such a code should be established, widely taught, and thoroughly practiced within the context of an overall security policy.

Productivity

A main purpose of computers is to aid the productivity of people and organizations. Many users respond negatively towards computer security, because they view it as interfering with their productivity. At least two factors contribute to this attitude: First, many users are not consciously aware of how security helps them with their work, for example, by protecting their files from accidental or malicious destruction, and by allowing selective on-line access to sensitive information. Second, many security mechanisms are overly complicated or tedious to use or install.

In addition, many organizations are reluctant to install security mechanisms that degrade the performance of the system or otherwise interfere with productivity. Indeed, many security mechanisms should not be installed for this very reason, because they are not justified by a cost/risk trade-off. Organizations are also reluctant to switch to more secure systems if the more secure systems are not compatible with the existing systems or provide less functionality. UNIX, for example, has remained popular despite its security weaknesses, because its functional properties contribute to user productivity. Because of its popularity, several secure versions of UNIX are under development (e.g., see [1]). In many environments, compatibility, performance, and functionality take precedence over security when upgrading to a new system.

If our goal as computer security professionals is to make systems more secure, then we must pay greater attention to the impact of our policies and mechanisms on productivity. In particular, we should strive for policies and mechanisms that, within the scope of threats they address, are transparent to users, simple to install and use, and offer positive benefits to the user community. To illustrate, we will discuss two broad classes of security controls: identification and authentication of users, and discretionary and mandatory access controls.

Identification and Authentication

A variety of different mechanisms has been developed to identify and authenticate users, including passwords, challenge/response protocols, biometrics, keystroke dynamics, access cards, and smart cards. These mechanisms vary considerably both in terms of the security they provide and their impact on productivity. For example, long meaningless *passwords* may offer greater security than short, easy-to-remember ones (if the users do not write them down in obvious locations), but are also more annoying to users. Some security experts have proposed using super-long, but meaningful, passwords, but we do not know whether these are preferred by users over shorter, nonsense passwords, because they require extra key strokes. Moreover, simply lengthening passwords does not protect them from possible exposure during transmission. Cryptographic-based *challenge/response protocols*, such as the PFX system developed by Sytek, can protect against certain threats not addressed by passwords alone (including the exposure threat during transmission), but at the same time lengthen the time required to login. *Biometrics*, such as signature verification, hand geometry, voice prints, and electronic fingerprints can add significant security, but can be expensive and generally require special equipment. Authentication through *keystroke dynamics* is attractive in terms of user productivity, because it is totally passive, low-cost, and transparent, requiring no action on the part of users. In addition, it offers continuous authentication, thereby protecting a user's session while the user is absent from the terminal. On the other hand, because of its passivity, it might raise privacy issues under certain circumstances if the users are not aware of its presence (we will return to this in the next section). *Smart cards* also can provide a high level of security without the need for much user interaction during login, but again require special equipment.

In addition to the various identification and authentication mechanisms, various strategies are applied when a user requests access to a subsystem or remote host. In many environments, the user must supply a separate password for each subsystem or remote host. Because this places an extra burden on the user, these additional passwords are frequently stored on the system, unencrypted, where they are vulnerable to exposure. Mechanisms that provide a high degree of security without requiring any additional information from the user better support the concept that computers are there to aid people.

[1] A recent study on the antisocial behavior of certain members of the computer community [3] concluded that rather different approaches to education are required: "... the cost of these educational environments may be considerably less than the losses being incurred." One particular recommendation was this: "Access to real computing power should be established for interested users, both students and their parents. Empowerment can lead to increased responsibility."

Access Controls

Discretionary and mandatory (multilevel) access controls can aid productivity by allowing sensitive information that serves the needs of different users to coexist on a single host computer or network. Without adequate host or network access controls, it is necessary both physically and logically to isolate the information, which interferes with a user's ability to access and integrate information. For example, because no commercial system supports a multilevel-secure database system, users who are cleared for information having different access classes (e.g., different sensitivity levels and/or different compartments) cannot access that data from a common database or manipulate it in a single session.

Discretionary access controls are often complicated, making it difficult to grant or revoke access to an individual user, and difficult to understand the implications of doing so. The former is due in part to inadequate user interfaces. For example, on some systems one must remember obscure commands for granting access and even what bit patterns correspond to what access modes! Search-path strategies further complicate matters. The latter is due in part to the inherent limitations of discretionary controls [5,6], and their lack of policy about information flow, including copies of information. The "setuid" facility of UNIX, for example, attempts to provide a mechanism for enforcing the principle of "least privilege," but has dire consequences if not used correctly. Because of the complications associated with discretionary controls, many users, accidentally or intentionally, grant access to all users rather than to those with a need for access.

Network access controls are often inadequate and difficult to analyze. For example, some network facilities have all sorts of special conventions whereby a user can remotely login or copy files from one machine to another without giving a password. However, there is no clear security policy or model underlying the mechanisms, and the result can be total confusion and misapplication of the functionality. Reid [7] describes how intruders broke into a network of UNIX systems by exploiting vulnerabilities in system directories and permission files. These vulnerabilities often arose from shortcuts taken by programmers to improve their own productivity, thus demonstrating the importance of providing secure mechanisms that do not burden the users, and the importance of making users aware of the consequences of break-ins.

Several studies [8,9,10] have shown the value of multilevel, lattice-based policies for controlling direct and indirect (via information flow) access to information of different sensitivities — that is, for enforcing multilevel security. Such policies are relatively easy to understand, avoid the need for users to grant and revoke access, and avoid the inherent limitations of discretionary policies. Moreover, because of their simplicity, it is possible to build systems that enforce multilevel security with a high level of assurance (B3 or A1), and such systems are now becoming commercially available. These systems are based on the concept of a reference monitor or security kernel. Examples include the Honeywell SCOMP and the Gemini GEMSOS [11]. Systems with a lower level of assurance (B1 or B2) could have enormous practical value in environments where the threat is not great, but the simplicity of multilevel security is desirable.

Applications are under development that can exploit the properties of a system enforcing multilevel security. For example, under sponsorship by the U.S. Air Force Rome Air Development Center (RADC), a team at SRI International and Gemini Computers is developing a formal policy model and design for a multilevel-secure database system, which is to be implemented on top of a reference monitor (e.g., GEMSOS) in order to provide A1 assurance [12,13,14]. The development of such applications will enable users to integrate sensitive data of different classifications, thereby improving user productivity.

Although most of the early work on multilevel security was aimed at protecting classified data, some was aimed at protecting sensitive data in the public sector [10], including proprietary and confidential data. Lipner [15] has shown how multilevel policies can be applied to commercial data, and Cohen [16] has argued that such policies help protect against computer viruses. Although we do not claim that multilevel policies and mechanisms can replace discretionary ones, we believe that their potential in the commercial sector has largely been ignored. While some organizations in the public sector have made efforts to classify information, few if any have attempted clearance of their users. Both classification and clearance must be rigorously and comprehensively accomplished in order to obtain the full benefits of multilevel security.

While multilevel security can improve productivity by allowing the integration of sensitive data having different sensitivity markings, if misused, it can inhibit productivity by restricting the flow of information, thereby interfering with the needs for efficient, timely, and effective analysis of information. For example, attempting to eliminate all covert channels in a system improves security, but also impairs communication and the flow of information; similarly, attempting to solve all possible inference and aggregation problems improves security, but makes data integration and analysis more difficult. When security and productivity compete, the appropriate balance can be determined only by examining the particular application environment.

Discretionary access controls are useful as a means of providing a finer granularity of control in order to enforce "need-to-know" constraints within the assigned classifications. However, because they are inherently more complicated and weaker than mandatory ones, they should not be relied upon to control the flow of sensitive information. The limitations of discretionary controls are particularly evident in databases, where access controls may be at the view level (or transaction level) so that authorization can be value-dependent, context-dependent, or history-dependent.

Other types of controls are also needed in order to ensure the consistency or integrity of data, and to enforce other security policies. Our formal model of a multilevel-secure database system, for example, supports database consistency through integrity constraints, transactions, and a mandatory integrity policy [17,18]. Clark and Wilson [19] argue that integrity is more important than multilevel secrecy in most commercial environments, and go on to argue that such a policy should include controls that enforce separation of duty among employees.

Privacy

Computer security is essential for enforcing state and national privacy laws. At the same time, the process of detecting threats, vulnerabilities, and abuses may result in violations of privacy and other human rights, leading to a conflict between the use of computer security to guarantee privacy and its use to invade privacy. These privacy issues became particularly apparent when backup files for a computer operated by the National Security Council were used to reconstruct and expose electronic mail messages regarding the Iran arms deal.

One area where this conflict is especially noticeable is *threat monitoring* — that is, analyzing system activity with the objective of detecting computer break-ins and abuse. We have identified several types of monitoring, listed in order of increasing privacy implications:

1. Continuous authentication, such as through keystroke dynamics.

2. Monitoring unusual activity on the system through system status information (e.g., tracking password failures and looking for sudden rises in system or network activity).

3. Maintaining an audit trail of user activity for the purposes of enforcing user accountability. User events recorded in an audit trail may include login times and locations, commands executed, and file accesses. This type of auditing is required by the Criteria [1] for systems that are rated at the level of C2 and above.

4. Analyzing user events as recorded in an audit trail in terms of abnormal behavior, where "normal" may be defined in terms of a user's past behavior or in terms of acceptable behavior. Under sponsorship from the Space and Naval Warfare Command (SPAWAR), we are developing at SRI a real-time Intrusion-Detection Expert System (IDES) that would detect various types of intrusions by looking for abnormal behavior on the system[20,21].

5. Monitoring the contents of files and messages (e.g., as for the Iran arms case). Any backup system potentially gives a mechanism for implementing this type monitoring, though they are generally not used for this purpose.

6. Complete surveillance of a user's terminal session — i.e., all information transmitted to and from a user's terminal (except possibly passwords). Limited forms of surveillance that provide this type of monitoring have been installed in some systems, and Clyde Digital Systems has developed a surveillance tool called the "Surveillance-Kernel."

Monitoring has many advantages. For example, it has been used to catch outsiders who have broken into computer systems, and it could potentially detect other forms of computer misuse that go undetected by other security controls. Monitoring might be especially attractive in environments where the systems themselves lack adequate security controls commensurate with the sensitivity of the information handled by them. By protecting confidential information about individuals from unauthorized access, monitoring can help enforce privacy rights and protect information assets.

While recognizing the benefits of monitoring, we have some concern that monitoring could foster a chilling and suspicious attitude in the working environment, especially if it is misused. In particular, the users could feel that they are not considered trustworthy or that their privacy and other rights are violated [22]. We are also concerned that threat monitoring could have an escalating effect as additional monitoring capabilities are developed in order to protect against a wider range of threats, while at the same time the user community becomes increasingly less satisfied with the working environment. Further, monitoring can aggravate the security problem if the data that are accumulated are sensitive but not adequately protected. For example, many audit logs accidentally expose user passwords, such as when a password shows up instead of the user identifier. Finally, the centralization of sensitive audit data that is not otherwise available in an integrated form has social implications.

Because real-time threat monitoring systems are not yet generally available, it is difficult to determine the extent to which these concerns are justified. We can get some insights from a study by Irving, Higgins, and Safayeni [23] on computerized performance monitoring, which showed that "workers perceive increased stress, lower levels of satisfaction, and a decrease in the quality of their relationships with peers and management as a consequence of computerized monitoring." At the same time, however, those authors found that the cause of the dissatisfaction was not so much the monitoring per se as that "managers overemphasize the importance of quantity [over quality] ... in evaluating employee performance." Thus, that study concluded that it is "not the technology itself, but rather how it is used by management that determines an individual's reaction."

We believe that any threat monitoring must be carefully applied to preserve the rights of privacy and freedom from intrusion, and avoid creating an atmosphere that leads to employee and other user dissatisfaction. When a computer system is being shared, users should not expect that they can function privately, in isolation; yet limits must be put on monitoring lest it become oppressive. These issues might be partially resolved by comparisons with analogous situations such as the sanctity of employee's desks and lockers, inter-office mail, television monitoring, use of work-place informants, and telephone eavesdropping practices. If suitably restricted and administered, monitoring of computer activity could be viewed as a benefit by the user community in much the same way that security monitoring of personal luggage at airports is viewed as a benefit by air travellers.

We recommend that a policy be developed regarding threat monitoring that addresses such areas as limits on threat monitoring, use of the results obtained from monitoring, obtaining informed consent of users, and providing due notice of intent to monitor. The development of a monitoring policy should not be limited to security experts, but should involve users, as well as psychologists, sociologists, constitutional lawyers, and human rights groups. We believe that this task should be assigned high priority in order that we do not find ourselves with threat monitoring systems that foster social problems in the work place. Our ultimate

goal must be to create an atmosphere that motivates people to behave responsibly and with confidence that both their rights and information assets are protected.

Protecting Noncomputerized Information

Although our society is still heavily dependent on information that is spoken and printed on paper, we often ignore the security of these other forms of information in favor of the technological challenges associated with automated information. Interviews of approximately 100 computer criminals, while not necessarily representative of all loss experience, indicate a skewing of emphasis across all forms of information [24]. Except for some of the malicious hackers, these people were attempting to solve their intense, unsharable, personal problems with the easiest, safest, and surest methods, constrained by their own skills, knowledge, and resources. Their preferred forms of information were the spoken word first, printed information second, and computerized information third. Computerized information received their focus of attention only when the other forms of information were not accessible to them or amenable to their knowledge and skills[25]. They did not need the computer as a tool to modify, disclose, or manipulate large amounts of information.

Protectors of information must assign similar priorities in applying security, while not overlooking computerized information in anticipation of the few, unusual perpetrators who do not fit the general pattern. Limited security resources would dictate "spoof-proofing" of key employees so that they are not deceived into giving information to outsiders who lack a need-to-know, and protecting printed paper and removable computer media before protecting information stored in computers or data communications [26].

Summary and Recommendations

The pursuit of technology, in the absence of a broad policy that addresses the social aspects of computer security, operates in a vacuum that may lead to violations of human rights, abuse, or other unwanted consequences. Attention must be given to the social aspects, and we make the following specific suggestions:

1. That the social aspects of specific computer security policies and technologies be examined in depth. Areas that should be addressed include identification and authentication, access controls (including those provided by add-on security packages), encryption, and threat monitoring. The technologies should be examined in terms of their actual and perceived effect on productivity and privacy.

2. That generic security policies be developed for different types of organizations and environments, taking into account the social aspects of information protection. The generic policies could serve as guidelines for formulating specific security policies within an organization.

3. That a national policy be developed specifically for threat monitoring that recognizes the rights of the users as well as the potential threats.

Even though the emphasis in this paper is on the social aspects, it is vital that the technological and the social considerations be balanced. They must go hand in hand. Either one without an understanding of the other is likely to create serious problems.

Moreover, security must be tempered with many other requirements that we have not addressed here, such as reliability, safety of use, and real-time responsiveness. To address a broad spectrum of requirements requires a holistic approach. At lower layers of system abstraction we tend to optimize rather locally to ensure that the technology satisfies rather specialized properties such as file privacy and integrity. At the higher layers the optimization may produce completely different results when all of the requirements are considered (technological and human, health and welfare, costs of automating, costs of not automating, etc.)[27].

Acknowledgments

We are grateful to Peter Denning and John Rushby for their constructive comments.

References

[1] *Department of Defense Trusted Computer System Evaluation Criteria.* Dept. of Defense, National Computer Security Center, Dec. 1985. DOD 5200.28-STD.

[2] N. Leveson. Software safety: why, what, and how. *ACM Computing Surveys*, 1986. to appear.

[3] J.A.N. Lee, G. Segal, and R. Steier. Positive alternatives: a report on an ACM panel on hacking. *Comm. ACM*, 29(4):297–9, Apr. 1986.

[4] V. D. Gligor et al. On the design and the implementation of secure XENIX workstations. In *Proc. IEEE 1987 Symp. on Security and Privacy*, pages 102–117, IEEE Computer Society, Apr. 1986.

[5] M. A. Harrison, W. L. Ruzzo, and J. D. Ullman. Protection in operating systems. *Comm. ACM*, 19(8):461–471, Aug. 1976.

[6] D. E. Denning. *Cryptography and Data Security.* Addison-Wesley, Reading, Mass., 1982.

[7] B. Reid. Risks: lessons from the stanford UNIX breakins. *ACM SIGSOFT Software Engineering Notes*, 11(5):29–35, October 1986. A shortened version of this material appears in *CACM* 30(2), February 1987, pp. 103-5.

[8] C. Weissman. Security controls in the ADEPT-50 timesharing system. *Proc. Fall Jt. Computer Conf.*, 35:119–133, 1969.

[9] D. E. Bell and L. J. LaPadula. *Secure Computer Systems: Mathematical Foundations and Model.* Technical Report M74-244, The MITRE Corp., Bedford, Mass., May 1973.

[10] D. E. Denning. A lattice model of secure information flow. *Comm. ACM*, 19(5):236–243, May 1976.

[11] R. R. Schell, T. F. Tao, and M. Heckman. Designing the GEMSOS security kernel for security and performance. In *Proc. 8th National Computer Security Conf.*, pages 108–119, 1985.

[12] D. E. Denning, S. G. Akl, M. Heckman, T. F. Lunt, M. Morgenstern, P. G. Neumann, and R. R. Schell. Views for multilevel database security. *IEEE Trans. on Software Eng.*, SE-13(2):129–140, Feb. 1987.

[13] D. E. Denning, T. F. Lunt, R. R. Schell, M. Heckman, and W. Shockley. A multilevel relational data model. In *Proc. 1987 Symp. on Security and Privacy*, IEEE Computer Society, 1987.

[14] T. F. Lunt, D. E. Denning, , R. R. Schell, M. Heckman, and W. Shockley. Element level classification with A1 assurance. 1987. Computer Science Lab, SRI International.

[15] S. B. Lipner. Non-discretionary controls for commercial applications. In *Proc. 1982 Symp. on Security and Privacy*, pages 2–10, IEEE Computer Society, 1982.

[16] F. Cohen. Computer viruses, theory and experiments. In *Proc. 7th DOD/NBS Computer Security Conf.*, pages 240–263, Sept. 1984.

[17] R. R. Schell and D. E. Denning. Integrity in trusted database systems. In *Proc. 9th National Computer Security Conf.*, pages 30–36, 1986.

[18] D. E. Denning, T. F. Lunt, P. G. Neumann, R. R. Schell, M. Heckman, and W. Shockley. Security policy and interpretation for a class A1 multilevel secure relational database system. Nov. 1986. Computer Science Laboratory, SRI International.

[19] D. D. Clark and D. R. Wilson. A comparison of commercial and military computer security policies. In *Proc. 1987 Symp. on Security and Privacy*, IEEE Computer Society, Apr. 1987.

[20] D. E. Denning and P. G. Neumann. *Requirements and Model for IDES - a Real-Time Intrusion Detection System.* Technical Report, Computer Science Laboratory, SRI International, 1985.

[21] D. E. Denning. An intrusion-detection model. *IEEE Trans. on Software Eng.*, SE-13(2):222–232, Feb. 1987.

[22] G. T. Marx and S. Sherizen. Monitoring on the job: how to protect privacy as well as property. *Technology Review*, 63–72, Nov./Dec. 1986.

[23] R. H. Irving, C. A. Higgins, and F. R. Safayeni. Computerized performance monitoring systems: use and abuse. *Comm. ACM*, 29(8):794–801, 1986.

[24] D. B. Parker. Consequential loss from computer crime. In *Proc. IFIP/Security 86*, 1986.

[25] D. B. Parker. *Fighting Computer Crime.* Charles Scribner's Sons, New York, 1983.

[26] D. B. Parker. *Managers Guide to Computer Security.* Reston, Reston, VA, 1981.

[27] P. G. Neumann. On hierarchical design of computer systems for critical applications. *IEEE Trans. on Software Eng.*, SE-12(9):905–920, Sept. 1986.