INFORMATION SECURITY: AN OVERVIEW

Alf L. Andreassen, William J. Leighton III, and David F. Schreiber

***Alf L. Andreassen** is with AT&T's Federal Systems Division, Washington, D.C. **William J. Leighton III** and **David F. Schreiber** are with AT&T Bell Laboratories, Whippany, New Jersey. Mr. Andreassen is Command, Control, Communications and Intelligence Systems vice-president. He was previously director, Information Systems Center, responsible for systems engineering, development, and research efforts for government customers. He received a Ph.D. in physical chemistry from Cornell University. Mr. Leighton is head of the Information Systems Engineering Department, responsible for research, design, and development of secure systems and networks. He manages the development of System V/MLS, a multilevel secure version of the UNIX® operating system. He joined*

(continued on page 8)

Information security technology encompasses all measures used to protect information from unauthorized disclosure, modification, or destruction. In an age when information is widely recognized as a valuable commodity, information security has become particularly vital. This issue of the *AT&T Technical Journal* brings together papers on various aspects of digital electronic information security, representing work in many areas of the company. This paper presents a brief overview of the problems addressed by information security technology and how its focus has changed with the advance of computing science.

The Secure Systems Concept

The elements of security technology may be realized at many levels in a system hierarchy. A representative system is illustrated in Figure 1. At a low level in the hierarchy, individual links in a network may be secured against eavesdropping by using pairs of encryption devices. At a higher level, individual general-purpose processors, workstations, and database processors may use access control mechanisms to protect themselves against unauthorized users. At yet higher levels, clusters of processors and even entire wide-area networks can use combinations of network security servers and localized control mechanisms to enforce overall system security. The key to achieving security in a system of any size is to analyze the security problem starting at the highest possible level, the system level. From that beginning, requirements for lower-level components can be specified to ensure that their collective effect is the desired level of system security and preservation of system performance. An example of the success of this approach is System V/MLS, AT&T Federal Systems' multilevel secure operating system. As a result of careful system-level analysis and implementation of security features, System V/MLS suffers only a 3-percent performance degradation relative to the standard UNIX® System V operating system (Figure 2). Some of the decisions that went into the design of

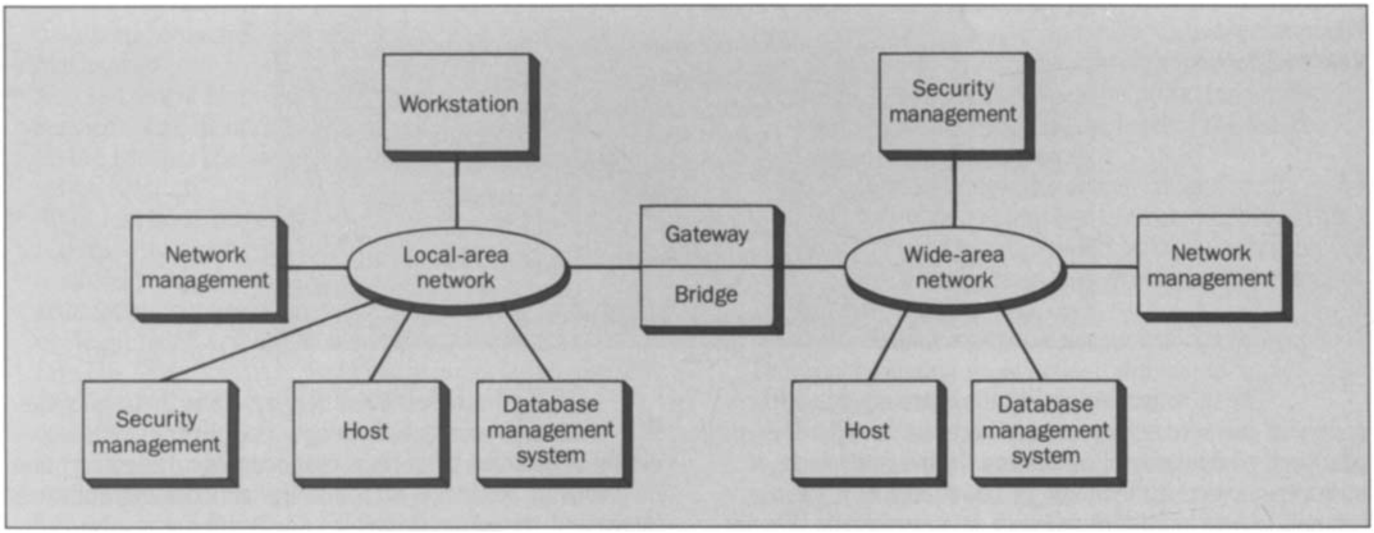

Figure 1. Secure system elements.

System V/MLS are described in this issue (see Reference 1).

The security-related activities at AT&T, as mirrored in this issue of the *AT&T Technical Journal*, span all levels of the hierarchy, from devices to processors to systems. At each level, however, function and design are governed by the principle that a system view is essential to achieving the goal of overall system security.

Evolution of Security Technology

Information security technology has progressed through three phases: data protection, system protection, and system verification. The onset of each phase has been marked by the application of digital electronic computing to a problem previously addressed by other means.

Encryption has been used since antiquity to protect data from unauthorized disclosure. With the twentieth century, however, came the first practical advances in techniques for automatically generating cryptographic transformations. Not surprisingly, these techniques relied on two predecessors of digital electronic logic: analog electronics (to scramble human-voice signals) and electromechanical logic (to encrypt alphanumeric data).

Security technology, however, must do more than protect data. Systems that handle the data must themselves be protected to safeguard their operating principles as well as the data they contain. In the analog/electromechanical era, equipment was protected simply by restricting direct physical access. As long as the equipment could not be operated remotely, this effectively solved the problem.

A security system, however, requires more than access protection. It must be verified that the system itself is properly implemented and that it operates as intended. Otherwise, the system could be altered secretly and maliciously. In effect, the system's *design*, as well as the physical system itself, must be protected against subversion. In the era before digital electronics, encryption devices were relatively simple, consisting of hundreds of computational elements. Visual inspection was the practical and effective way to verify correctness.

The three phases of security technology are related to three kinds of protection. In phase 1, digital electronic computing was applied to the problem of protecting data. Phase 2 brought computing technology to bear on the system protection problem. Phase 3 is system design protection, or verification.

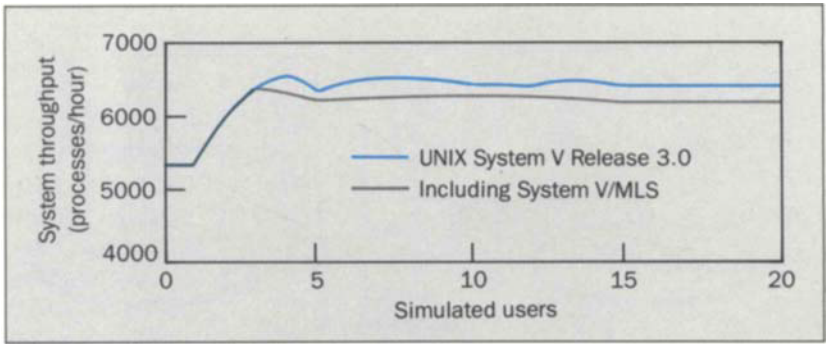

Figure 2. System V/MLS performance.

The introduction of digital electronics brought orders-of-magnitude increases in both the speed and complexity of computational machinery. In the first phase of information security technology, this machinery was brought to bear on the problem of data encryption. Digital electronic encryption processors could be installed at both ends of a circuit to secure data flowing through it against eavesdropping. A digital electronic encryption processor enjoys two major advantages over its analog and electromechanical antecedents. First, its speed lets it function transparently in a circuit, even at high data rates. Second, its scale and complexity allow construction of highly secure encryption mechanisms (now, algorithms) using many thousands of computational elements.

Although digital electronics revolutionized encryption machinery, the strategy for protecting that machinery remained largely unchanged: keep it behind a locked door. Again, this approach worked as long as the equipment had no remote control interfaces.

Because of its complexity, a digital system is like a two-edged sword: it makes highly secure encryption algorithms practical, but it frustrates verification. Even though the design is accessible, the equipment itself is often potted, integrated, or otherwise difficult to inspect. Thus, verification becomes a three-step process. First, the correctness of the design is verified by inspection. Next, convincing arguments are used to show that the equipment faithfully realizes the design. Finally, the equipment is tested extensively in the hope that residual errors will be uncovered.

Despite these difficulties, systems that typify the first phase of security technology carry over certain essential qualities from their predecessors: the encryption mechanisms are based on hardware or firmware, and, to the extent that programmable computers are involved, access is restricted to a few individuals. Thus, opportunities to subvert these systems are limited.

Two developments in computer applications—time-sharing and networking—acted together to usher in the second phase of security technology. Time-sharing dramatically increased the number of individuals with direct access to a computer, and networking brought another dramatic increase in that number. With these developments, the door to the computer room was effectively unlocked and thrown wide open. As the role of physical security was reduced, it became necessary to incorporate protective measures directly into the hardware and software of the computer itself. Thus, security technology advanced as digital electronics took on the task of system protection.

Software-based protection measures are those with which computer users everywhere have become familiar: password protection to control system access and privilege mechanisms to control access to specific resources within the computer.

These protections may be grouped into the following general categories:

- Authentication includes functions whose purpose is to ascertain the identity of an individual or a piece of computing machinery; the standard UNIX system login/

password sequence is an example of an authentication mechanism.

- Access control functions determine and control which services shall be offered to a given individual or machine on the basis of the identity established by the authentication function.
- Audit functions record and analyze all security-relevant activities on a system. Audit data collection and analysis is aimed at uncovering these classes of activities: (1) attempts to commit unauthorized acts and (2) patterns of "legitimate" activity whose collective effect is sinister. The latter target is particularly important because it represents the security officer's only handle on the activities of malicious yet fully authorized users.

Unfortunately, software-based protection mechanisms have proven highly vulnerable to attack. Because software is inherently fluid, the smallest flaw in the system offers the potential for changes in the protection mechanisms, changes that can completely compromise system security.

Related to the weakness of computer-based system protection is the difficult problem of verification. The programmable computer is, by definition, flexible. Any direct effort to verify its functionality is doomed to fail. Instead, the verification concept must be applied at a higher level of abstraction: not to the tasks the system performs but, rather, to the controls that the system enforces. In the phase 2 environment, system verification remains a manual task. Although formal methods may be used to advantage even at this stage, the transition to full use of formal verification represents a significant advance in the scope of security technology and properly belongs in the next phase.

The difficult task of maintaining security controls over a programmable computer is further complicated by the proliferation of networks among processors, peripherals, and user terminals. The difficulty of applying security controls in a network environment stems from these factors:

1. Networks and their connected processors are exposed to a large community of potential attackers.
2. Networks are dynamic.
3. Networks frequently span administrative domains and tend to be administered haphazardly, if at all. Thus, resources on a network tend to be protected at the "low-water mark," that is, at the level of the least secure system on the network.
4. Networks use a variety of communications media, some of them potentially susceptible to eavesdropping. This problem is sometimes addressed by using encryption, a solution that introduces its own problems in the area of key administration.
5. Networks use a variety of communications protocols. Differences among them make it difficult to apply strong, network-wide security features.
6. The scope of a network grows erratically with new network connections: a new connection may introduce one new user to a network, or it may introduce a new network of wider extent than the first one. This problem, coupled with weak administration, permits new network connections without knowledge of the attendant risks. The result is uncontrolled exposure.

Network security is such a difficult problem because of an underlying contradiction: the objectives of networking and those of security are fundamentally at odds. Networking fosters general, flexible access, whereas security seeks to impose limited access rules under rigidly controlled conditions.

The mechanisms used to protect networks are analogous to computer security measures, including elements of authentication, access control, and audit. Implementing these mechanisms in a network is complicated by a conflict: distributed processing gives power and flexibility, while centralized control inherently gives security.

The third phase of security technology is just getting under way. It represents a consolidation of the advances that precipitated the previous phase. As it is currently evolving, the latest phase is characterized by two major developments: standardization of the network environment and automation of system verification.

The significance of standardization of the network environment goes far beyond the security problem. Although any pair of network components eventually can be made to work together, experience has shown that both the economy and reliability of the process can be significantly advanced if both components adhere to a common standard. This observation is all the more important for security functions. Custom implementation of security features where none previously existed and patching together incompatible features on separate systems are costly and time-consuming endeavors. They are less likely to produce an error-free security system than standard components would.

Several efforts are currently underway to develop sets of security standards for specific environments. Two of the articles in this issue, "Security Standards—Government and Commercial"[2] and "The Secure Data Network System,"[3] address this topic.

The second development, automated verification, finally brings protection of system design under the umbrella of modern security technology. Automated verification is an implementation of the underlying technique of formal verification, the application of rigorous mathematical logic to verify the security of system design.

A system may be represented at different levels of abstraction, from top-level requirements, through various levels of specification, down to the implementation logic in hardware and software. In the formal verification paradigm, the elements of a lower-level abstraction (*axioms*) are used to prove the elements of a higher-level abstraction (*theorems*). Theoretically, it is possible to obtain a set of proofs that go level by level from code to requirements; in practice, the most stringent security verification requirements in the Department of Defense (DoD) *Trusted Computer Systems Evaluation Criteria* (*Orange Book*) call for only one layer of proofs.[4]

So far, none of this has anything to do with automation. Proving something, however, is generally more difficult than simply stating it; the proofs used to verify the correspondence between two descriptions of any nontrivial computer system tend to be immense. Creating and testing these proofs by hand is tedious and conducive to error. Fortunately, techniques have been developed to test and, to a limited extent, to create mathematical proofs using a machine.

Even if proofs could be generated completely automatically, human intuition is still required to properly define each level of system specification properly. Thus, the role of automation in system verification remains that of a useful tool rather than a complete solution.

Although the third phase of security technology is still new, the widespread propagation of standards, both for security and other purposes, seems assured, although their final form remains largely to be determined. Automatic verification techniques are seeing early applications, although the technology is not yet mature and the tools have yet to make the transition from research grade to industrial strength. As this phase unfolds, it will become clear which of the current elements will make sustained contributions and what new requirements and technologies will emerge.

Applications of Security Technology

Having sketched the evolution of security technology, we proceed to outline some of its applications.

Customers for security technology are frequently categorized as either government or commercial. It is useful to subdivide these categories, identifying the defense community and the financial community as particular centers of interest within the government and commercial establishments, respectively.

The defense community's data security concerns fall into three areas: privacy, integrity, and reliability. Privacy includes preventing disclosure to the outside world as well as separating data on a need-to-know basis within a given defense agency. Currently, these tasks are performed largely by encryption technology and physical isolation of computers that support different security domains. As networking and verification technologies mature to the point of providing adequate levels of interoperability and assurance, these domains will probably be able to share processing and communications resources.

Data integrity measures protect data from corruption while the data move from source to destination. Although transport and storage media may provide the basis for maintaining high fidelity in data, the responsibility for testing for accuracy falls on the endpoints. There, it tends to be addressed by using application code in programmable computers.

In the context of security technologies, reliability is a property by which a system operates correctly, doing all that it should and nothing that it should not. As a verification issue, this form of reliability is beginning to be addressed by automated verification technology. Currently, design review and implementation testing are the main assurance methods.

The data security needs of the commercial sector span a wide range of information types, including:

- Electronically stored documents
- Strategic business data
- Accounting data
- Personnel information
- Technical product data: designs, test results
- Software to support internal operations
- Software products.

As in the defense area, commercial security includes privacy, integrity, and reliability, although the emphasis on the latter category is not as strong as for defense security. Commercial data security generally relies on the system protection technologies of phase 2. However, particularly sensitive sectors (e.g., payroll) will resort to the technology of phase 1— that is, isolated computers with physical access controls.

The requirements of the financial community pose some of the most interesting challenges currently faced by security technology. Here, the concern for data privacy is equaled and even surpassed by the need for strong data integrity measures. For example, any party to a financial transaction should be able to not only convince itself that the transaction is authentic, but also to establish its legitimate claims before a judicial authority. Functions of this sort rely on end-to-end protocols involving cryptographic elements. They will depend heavily on the development of industry-wide standards for electronic financial transactions. This development falls into the phase 3 category.

In This Issue

The papers in this issue describe projects that draw on various phases of security technology to satisfy requirements in the government, industrial, and financial communities.

"Security Standards—Government and Commercial"[2] surveys and compares activities among various bodies now establishing and promoting standards for information security. This paper highlights the security requirements and accompanying standards in the financial community.

"The Secure Data Network System"[3] (SDNS) reviews a cooperative industrial/government program now under way to define standards for secure communications protocols and establish the basis for a commercial line of secure data communications products. SDNS security functions are being defined within the framework of the Open Systems Interconnection (OSI) Reference Model. These functions are intended to be added to various kinds of processors (personal computers, workstations, host computers). At the heart of the SDNS concept is an end-to-end encryption mechanism based on an advanced key management system.

"A Security Methodology for Computer Networks"[5] describes a life-cycle approach to system security based on two phases: the design and implementation phase and the operational phase. In the design and implementation phase, the methodology focuses on defining protected resources, security policy, and threats; developing countermeasures; assessing residual risk; and building the basis for operational approval. In the operational phase, the focus shifts to processing unusual security events, including detection, evaluation, and decision on subsequent action.

"Design of a Certifiable One-Way Data-Flow Device"[6] reports on the design of an asymmetric filter that has high forward channel bandwidth, but permits reverse channel bandwidth to be kept below any desired limit (the practical example used is 0.1 bit per second). The subtlety in this task derives from the following requirements: that protocols, including acknowledgments, be passed at a substantial fraction of full speed, and that the filter design be simple enough to be readily verifiable without the aid of automation. This device is meant to act as a trusted interface between two untrusted systems of different security levels.

"Sandia's Terminal Switching Network and Fiber Optic Loop"[7] offers an overview of a secure campus-scale data network currently in use. It describes how the network media, architecture, and special switch firmware are used to build a highly secure network that nevertheless retains much essential flexibility.

"System V/MLS Labeling and Mandatory Policy Alternatives"[1] describes AT&T Federal Systems' first multilevel secure UNIX system-based product that will be evaluated against the criteria of the National Computer Security Center (NCSC). Under evaluation for a B1 rating from NCSC, System V/MLS adds several security enhancements to the standard UNIX system, including mandatory access controls based on labels consistent with the DoD classification scheme, improved protection of passwords, extensive auditing, boot-time assurance measures to detect the introduction of malicious code, and restriction of certain capabilities that historically have been responsible for security failures. The first applications of System V/MLS will be in the government area, although it is also suitable for protection of private commercial information.

"Quest—A Security Auditing Tool"[8] presents the capabilities of a collection of auditing tools that check for security problems on UNIX systems. These problems include nonsecure login identifications and passwords, nonsecure settings of file and directory permissions, unauthorized attempts to acquire extended privileges, potential "Trojan horse" programs, and nonsecure settings of the user's environment. Quest is designed for use in commercial UNIX system installations.

These papers represent a sample of the security-related activities at AT&T, not an exhaustive survey. Continuing development and growth of information security technology at AT&T will play an essential role in serving the needs of our corporation, our customers, and the national defense.

References

1. C. W. Flink II and J. D. Weiss, "System V/MLS Labeling and Mandatory Policy Alternatives," *AT&T Technical Journal,* Vol. 67, No. 3, May/June 1988, pp. 53-64.
2. L. K. Barker and L. D. Nelson, "Security Standards—Government and Commercial," *AT&T Technical Journal,* Vol. 67, No. 3, May/June 1988, pp. 9-18.
3. B. C. Karp, L. K. Barker, and L. D. Nelson, "The Secure Data Network System," *AT&T Technical Journal,* Vol. 67, No. 3, May/June 1988, pp. 19-27.
4. *DoD Trusted Computer Systems Evaluation Criteria,* United States Department of Defense, Publication No. 5200.28, December 1985.
5. L. G. Pierson and E. L. Witzke, "A Security Methodology for Computer Networks," *AT&T Technical Journal,* Vol. 67, No. 3, May/June 1988, pp. 28-36.
6. R. L. Sharp, "Design of a Certifiable One-Way Data-Flow Device," *AT&T Technical Journal,* Vol. 67, No. 3, May/June 1988, pp. 44-52.
7. L. F. Tolendino, S. D. Nelson, and S. A. Gossage, "Sandia's Terminal Switching Network and Fiber Optic Loop," *AT&T Technical Journal,* Vol. 67, No. 3, May/June 1988, pp. 37-43.
8. S. A. Kapilow and M. Cherepov, "Quest—A Security Auditing Tool," *AT&T Technical Journal,* Vol. 67, No. 3, May/June 1988, pp. 65-71.

Biographies (continued)

AT&T in 1976. He received a B.S. from Villanova University and an M.S.E. and Ph.D. from Princeton University, all in electrical engineering. Mr. Schreiber is a distinguished member of technical staff in the System Architecture Group. Since joining AT&T in 1984, he has worked on many security-related systems engineering projects, including secure local- and metropolitan-area networks, operations support systems, and message switching systems. He received an A.B. from Harvard University and an M.A. and Ph.D. from Princeton

(Manuscript received March 23, 1988)