Security for Infinite Networks

Ruth Nelson
Information System Security
48 Hardy Ave.
Watertown, MA 02172

Hilary Hosmer
Data Security, Inc.
58 Wilson Rd.
Bedford, MA 01730

Abstract

Although network security theory forbids many connections to large networks as being too risky, the reality is that large numbers of sensitive systems are connected to the Internet and that connectivity is incresing at a rapid rate. Firewalls and host protection mechanisms are used in a somewhat arbitrary fashion, depending more on the availability of products than on a clear understanding of security principles. We need to expand security theory to protect large networks.

This paper proposes a new paradigm for security in large networks, based on an understanding of the sometimes conflicting requirements for security, connectivity and functionality. The paradigm, called FICS-IT, consists of a philosophy, an approach, a framework , and a collection of components. It is based on an understanding of security as risk management and includes local resource control; multiple, tailored security policies; layered, functional access control; and recognition of heterogeneity in architecture, ownership and policy.

Introduction

Network connectivity and functionality are in a period of explosive growth. Corporations and government need connectivity with large networks, including the Internet, to support their business and provide their members with access to public information. They also need security for their valuable information assets.

Current network security approaches are not adequate to mitigate the risk of network attachment. They restrict connection and communication to isolated domains with single security policies and they require a security assessment of the entire connected network.

With the support of a Small Business Innovative Research (SBIR) contract from Rome Laboratory,[1] we have developed a new paradigm for infinite network security, which addresses the new question. The new paradigm, which we call FICS-IT, considers security as risk management rather than an absolute goal. It includes connection strategies and architectures; supports multiple policies for access control for connection and function as well as data, and assumes a heterogeneous environment without a central authority. FICS-IT stands for **F**unctional, **I**nformation, and **C**onnection **S**ecurity for **I**nformation **T**echnology.

The new FICS-IT paradigm is based on an understanding of the connectivity, functionality, operation and management of today's large networks; it is extendible to new network architectures and technology. It defines security as risk management, rather than in absolute terms. It allows resources to be protected in networks that are heterogeneous on architecture, ownership and policy.

The FICS-IT approach includes the concept of local resource control: that the owner or holder of an information resource makes decisions about the release or dissemination of that information. This means that most FICS-IT functionality is located at or near processing systems. Decisions are made, based on local policies, whether to allow specific external *connections* to the protected processing system, what system *functions* are available to particular requesters, and what *information* can be released from the system and sent to particular destinations over particular network communications facilities.

FICS-IT functionality may be embedded in the processing system itself, or it may reside in a front end to a system or an application gateway between two connected networks.

1 This work was supported under Contract #F30602-94-C-0183, from the Air Force Materiel Command.

0-8186-7318-4/95 $4.00 © 1995 IEEE

The Infinite Network Security Problem

Large (or infinite) networks are not under control of a single administration, are constantly changing, include many differing policies, and are open to potential attacks from a large population of users. The most promising approach to deal with this fact is not to try to enforce centralized control, but to emphasize local control and administration plus coordination between local authorities to permit needed interactions. The question is not whether we can make a secure network; it seems obvious that we cannot assure the security of an entire large net. Rather, we should ask how to attach computers to networks so that the risk of attachment is minimized and how to exchange information usefully and securely among networked computers.

Our approach changes an impossible problem into a tractable one by redefining and clarifying the issues and requirements. It is impossible to define a single security policy applicable to infinite networks; it is impossible to determine whether a large network is secure (much less to make it secure); and it is impossible to separate infinite networks into isolated domains each representing a single security community. It is possible to reconcile conflicting security policies and make agreements between systems to exchange and label data. It is also possible to design computers that can attach to infinite networks without enormous risk and to exchange information usefully and securely among networked computers.

We have divided the problem into three areas. The first of these is the problem of secure communications: moving data from one computer to another through the network. There are a number of good, mostly cryptographic, solutions for this, and so we do not plan to expend much more effort on this area.

The second problem is how to connect a computer securely to a network. The issues in this area are functional limitation of access and authentication of access requests. In general, computers attached to a network receive and act on requests for access to data and processing resources under their control. These access decisions must be enforceable, and this requires strong mechanisms within the computer to limit access to authorized functions as well as authorized data. We need to define modes of computer security that go beyond current MLS schemes in order to reduce the risk of attachment to infinite networks.

The third problem is management and dissemination of the information required for access control. The information must be available to the computer controlling the processing or data resource before the resource can be released to a requester. This requires that policy issues be resolved, that access decisions be made and communicated to the resource holders, and that sufficient authentication information be available for effective enforcement of the decisions.

Infinite Network Security Issues

Infinite networks cannot be secured as single entities. They are constantly changing, with new systems and even networks being added, all with great frequency. There is no centralized or hierarchical management. There are no boundaries to the net, and there are as many policies as service providers. Many current users don't appreciate the value of their assets or the extent of their vulnerabilities. The Internet culture of openness, many free services and unlimited connectivity is antithetical to security.

Current network security approaches cannot deal with infinite networks. The approach based on computer security theory and embodied in the Trusted Network Interpretation (TNI) [1] of the Trusted Computer System Evaluation Criteria (TCSEC) [2] assumes that the entire network must be secure. It prescribes a system-wide network security policy requiring an administrator with an overall view and control of the entire network. This view precludes the interconnection of secure assets with the "outside world," isolating assets rather than controlling and protecting the necessary connections.

Ad hoc security solutions using firewalls pose problems because they operate on unauthenticated data, violate protocol architectures, and do not provide all the needed user functionality. Users behind firewalls are precluded in most cases from accessing useful public data, while the protection against outside misuse of resources is incomplete.

The Finite Paradigm

Current network security theory is based on the idea of securing the whole network, which means that one must be able to characterize the whole network. As part of this study, we examined the basis of this

paradigm, so that we could understand both its source and also the ramifications of changing it to fit the infinite network problem. Since many of the most basic assumptions of this (and other) models are not stated, they are difficult to examine. A historical look has proved useful.

Part of the difficulty with current finite models is that they assume that secrecy can be guaranteed or at least measured. In an infinite environment, it becomes very clear that security is a fuzzy concept. It is extremely difficult to define requirements, even for a small part of the network. This is especially true when it is necessary to connect two systems or subsystems with different goals, functionality or management. The assumption that there can be a single policy, approach, and set of security services for a large network is simply not correct.

A New Infinite Network Security Approach - FICS-IT

FICS-IT, integrates the Mutual Suspicion [3] work of Ruth Nelson, the Multipolicy Machine work of Hilary Hosmer [4], the complex network management paradigms of David Bailey [5], and the transaction-based integrity model of Clark and Wilson [6].

FICS-IT defines security as risk management, not in absolute terms. Its wider, more flexible definition of security is useful in designing and analyzing systems that must connect to, and interact with, large networks and less secure systems. It considers the tradeoffs between security and the needed network connectivity and functionality. FICS-IT allows connections between domains with differing security policies, mechanisms and assurance levels. It integrates multiple policy considerations with a decentralized security architecture.

FICS-IT emphasizes functional limitation of access, so that connections are permitted for certain purposes but not for others. This concept was initiated by Clark and Wilson [6] for commercial applications, and was extended to more general environments by GTE's VISE [7] work. Functionally restricted connections are safer than login access, which underlies even some file transfer connections in the Internet protocol suite. Enforcing this limitation in a server machine allows it to provide needed access without excess risk of losing control of all the machine's resources.

Mutual Suspicion and the GTE Architectures

GTE performed two research projects in network security from 1985-1992,[2] under the sponsorship of NSA. One result of this research was an Internet Security Architecture, which tied the security services and mechanisms to specific Internet protocol layers and functionality. It emphasized a more functional approach to security than previous efforts. The architecture led to development of a prototype network layer encryption protocol and the definition of the Internet Security Service (ISS), which is the minimal service necessary to deliver individual data objects across the Internet over a real-time communications connection.

A second result of the GTE research was the creation of the Mutual Suspicion Model for network security. This allocated services and mechanisms, not by protocol layer, but rather to processing and communications components of the network. The communications security service is the ISS. The processing systems attached to the network are responsible for protecting the resources under their control. There is no concept of secure network, since the network is assumed to be heterogeneous, large and not controlled by a single authority. The Mutual Suspicion concept, with its Local Resource Control, is one of the foundations of the work on Security for Infinite Networks.

Functional Security Model

A truly secure and effective system would allow information to flow as needed for system functionality without allowing unauthorized release of secrets. A security approach that considers processing function as well as information flow provides for the required information flow, but controls the functions that cause this flow to happen. This is significantly more flexible than the current computer security models that consider system activity only in terms of data access.

GTE developed a paradigm called VISE [7] based on work by Clark and Wilson, which is based on controlling access to particular processing functions as well as to data. This allows processing systems that are attached to networks to control access to their resources in a more precise and realistic fashion. The

2 Internetwork Security Research, Contract MDA904-89-C-6030, 1989-1993

VISE work is the basis for the functional access control portion of FICS-IT.

The Clark and Wilson model recognized that the functionality of a computer system depends on its application software, and that correct functionality depends on correct software accessing the appropriate data for that program. It also recognized the need for functional limitation of access, that is, for allowing users to use some software in the system and not other software. Their model went beyond subject-object to the triple of user, program, data. The Clark and Wilson work was intended to capture the way that operational business data processing is done. The application software for commercial systems is also usually developed on separate development machines, is installed and becomes operational through controlled procedures and is not permitted to be changed or updated in an uncontrolled fashion. In addition, auditing is commonly done in business computing so that transactions are checked and books balanced.

The Clark and Wilson model incorporates the business computing paradigm. Only some of the software is configuration and access controlled. Auditing and balancing functions are included in the model. At GTE, we recognized the similarities between the Clark and Wilson paradigm and the DoD mission-oriented systems we were developing. We extended the model and removed the constraints of business application.

Firewalls

Security solutions based on "firewalls" have recently become popular. Firewalls were suggested in the Mutual Suspicion paper as part of the security solution, with multiple firewalls providing possibly imperfect filtering and authentication. Our work emphasized that the effectiveness of security mechanisms placed at intermediate points in the network depends on the integrity and authentication of the data used for filtering decisions. In general, end-to-end authentication and integrity measures like source-to-destination encryption are preferred over intermediate system solutions. Firewalls at intermediate points in the network are essential, however, to allow for interconnection of systems using different means of providing network security and to limit spread of damage from a single compromise.

Current firewalls perform either a router or a gateway function. Both approaches pose architectural and security problems. In the router case, the filtering is based on IP addresses and/or port identifiers. Without end-to-end encryption or other strong mechanisms, these are not authenticated data. They can be and have been spoofed. The gateway or Bastion Host firewall terminates connections and acts as a proxy for user communication. These pose problems because they do not permit use of many of the newer information access protocols popular today. Users behind such a firewall are precluded in most cases from accessing useful public data.

Elements of the FICS-IT Paradigm

The FICS-IT paradigm is based on some realistic principles about how infinite networks behave and how they can be managed. These are:

- *Local Resource Control*

  The infinite network cannot be defined, let alone secured. Protect the assets under your control. Resources to be protected include information, processing capability (access to processing systems and to programs), and communications capability (access to connections between machines).

- *Multiple, Tailored System Policies*

  Policies for authentication, audit, transmission, marking and access control can be tailored by the local site to reflect the specific intentions of the system managers and the specific services offered by the system. The policies can accurately reflect the local decisions and the agreements with users and other domains on sharing of system resources.

- *Layered Access Control*

  Provide multiple checks and mechanisms. Disallow connection with all systems except those with which there is a policy to connect. Restrict connections that are allowed to specific purposes and particular data only.

- *Connection Control*

  Disallow connection with all systems except those with which there is a policy to connect. Allow connections for specific purposes and to access particular data only. This is the security principle of least privilege applied to networks. Local policy administrators can specify connectivity to other systems as well as access privileges of network users.

- *Functional Access Control*

  Login access is a security risk which has been exploited in a number of recent break-ins. It is safer, if possible, to define the services offered by a server on the network and to restrict user access to those offerings. This requires that policies and enforcement mechanisms be generalized from the current subject-object model to include user, function and information as separate primitives. If a system can protect the software that provides its application functionality, then it is safe from viruses and worms.

- *Application-specific Security*

  Application-specific security policies and mechanisms allow finer grained control of system resources than owner or label-based policies which ignore the information content and particular structure. These have become increasingly useful and necessary as information resources have become more sophisticated.

- *Connectivity as Needed for Mission*

  Emphasis on the traditional security models has led to a concept of single security environment domains, which have uniform access within the domain and are totally separated from other domains. This is the approach recommended in the DoD Goal Security Architecture [8]. The separation into discrete domains does not address the requirement of moving data between domains or between systems with different security policies. Requirements for this information transfer exist and must be included in the infinite network security solution.

- *Security as Risk Management*

  The definition of security and the policies embodying that definition are not absolute; network and computer security is always a tradeoff between controlling risk and permitting needed access. The functional access control and layered access control included in FICS-IT allow the security tradeoffs to be explicitly defined and the risk of network attachment to be carefully managed at each local system. Problems such as inference and aggregation can be managed in the context of specific access to specific information through specific programs; this is easier than addressing the unsolvable general problem. Application-specific policies and mechanisms can also allow detection or prevention of anomalous behaviors, as well as notification of authorities and application-specific audit.

- *Allocation of Security Mechanisms*

  Security mechanisms work only if they are allocated to network components carefully and in accordance with a security architecture. In our approach, we separate the problems of moving data between processing systems, controlling access to the resources within those systems, and managing the assets of the systems. We differentiate between security mechanisms not only on the basis of what security service they provide (confidentiality, integrity, etc.) but also on the particular information resources they protect (all communication data, messages, use of processing capabilities, etc.)

- *Security Planning for the Global Village*

  In the infinite network, heterogeneity is a primary characteristic. Most architectures assume commonality of mechanism throughout the entire system. We cannot. There will always be interfaces between dissimilar systems which need to share data. In general, the workings of these interfaces are a matter of agreement between the parties and will not follow a single pattern. However, it is useful and important to achieve clarity and consistency in the connected network, for both security and functionality. The security planners need information on functionality, protocols, and available and compatible security

mechanisms. They also need guidance in assessing security requirements and risk.

Decomposition of the Network Security Problem

Some of the previous network security research has assumed that the network can be viewed as a large processing system with many users. This view does not lend itself to practical approaches for infinite networks, because of their heterogeneous, changing and uncontrollable nature. FICS-IT is based on the understanding that security is not a global, general or abstract property of systems. Security is specific: a secure system is one that behaves the way it was designed to behave, even when there are attempts to make it behave differently. This definition may include the usual security properties of MAC and DAC,[3] if those are part of the system policy, but is a more flexible definition of security. It covers protection of all system resources, including processing time and communications capability as well as data.

The functions of the communications facilities and the processors connected through them are different; they can and should be considered and analyzed separately, in the light of possible attacks on those functions and components. We can consider the requirements for connecting a computer to a communications network and for allowing particular kinds of communications to occur. The requirements can be allocated to the processing machine and the network.

This decomposition separates the problem of moving data securely between machines in the network from the problem of processing that data within a machine and managing the communications. The problem of moving the data is not totally solved, but there are some useful definitions of requirements and some strong, encryption-based techniques. Our new research has focused on issues in the processing realm. These include:

- *Functional Limitation of Access*

  Application-oriented access
  Message communication as limited access
  Non-login access - is it possible to enforce?
  Restriction of access using front ends

- *Authentication of Access Requests*

  Authentication uncertainty
  User authentication vs. machine authentication

- *Management of Multiple Policies*

  Resolution of conflict
  Subject-object policies vs. user-method-data policies
  Policies for authentication, data export, data labeling

- *Architectural Issues*

  Firewalls - functions and effectiveness
  Options for allocation of requirements
  Measuring assurance

This decomposition of processing system security, as well as analysis of the security issues of moving information between processing systems, is useful for evaluating security in current systems attached to networks. It is also useful for design of new systems and system components.

Practical guidance is sorely needed in network security, particularly in the following areas.

- Strategies and requirements for securely connecting a computer to a communications network.
- Allocation of security requirements to the processing machines and the network.
- Strategies for managing communication between attached computers.
- Strategies for setting and enforcing network communications policies.
- Guidance for implementing a layered security approach for controlling access to a computer's resources.
- Aids for defining metapolicies to resolve conflicts between policies.

3 See TCSEC, Reference 2, Mandatory and Discretionary Access Control

- Guidance for implementing functional access control using the Multipolicy Machine and (user, program, data) tuples.

Login is Dangerous

Most operating systems in use today are based on time-sharing operating systems of the 60s and 70s. These were designed for human users at terminals, who logged in to access the computer resources. Today, most users have workstations or personal computers acting as terminals, and these are more powerful than the old timesharing systems. Most systems are still designed for login access, and this is a key security vulnerability. The flexibility and uncontrolled nature of this access method means that the system software must be extremely resilient and able to handle any possible sequence of user commands and input. Many penetrations have come from exploiting this flexibility and from taking advantage of the fact that users are not restricted to particular system functions.

The most vulnerable aspect of a UNIX system is root access. Most of the successful attacks include gaining access to root and then changing protection mechanisms, etc. so that the attacker can use the system resources freely. Protection of root is usually the same as for all other user protections, often a simple password. Some of the attacks have used password snatching; others have been more sophisticated. A vulnerability that has been exploited a number of times is that users can cause the execution of programs that run with privilege. If the code of those programs can be changed, or if the program can be made to execute code that it has read as data, then it is possible to gain access.

Avoiding login access is a strong protection against this form of attack. Part of the FICS-IT approach is to limit access to only those functions (and programs) which are specifically authorized and to limit logins, at least to the actual operating system controlling the machine. However, current Internet protocols, even those for file transfer and database access, often require login.

The necessity for login access has been assumed, at least in the academic and DoD community, for many years. It provides the most flexibility to remote users, but also the most risk. Recently, there have been some examples of systems that restrict access. A Sidewinder system, developed by SCC,[4] is on the Internet as a challenge. SCC has advertised its availability as a firewall and offered a prize to anyone who can break through the security protection to the "inside." Part of the protection is that users can appear to get login access to the system, but in reality this is only to a shell program operating in user space, still on the "outside."

Another example of restricting login for security is in the Electronic Key Management System (EKMS), built by GTE for NSA. This system performs very sensitive key management functions, filling electronic orders for cryptographic keying material. It has a network interface which is unclassified. The architecture includes a separate front-end system, called the Message Processor, attached to the network, which is set up to send and receive electronic mail and some other specialized messages. The sensitive functions are in a separate machine, with extremely restricted login access and no direct network connection. Messages from the network are validated and reformatted before being passed to the protected machine for processing.

The EKMS and the Sidewinder challenge are worked examples showing that functional limitation of access and restriction of login can be done. Pursuing this approach in designing service providers has the potential for improving both the functionality and the security of networked computing systems.

Placement and Use of FICS-IT Components

The FICS-IT approach looks at interfaces and connections to the network. This allows attachment of dissimilar network domains as well as addition of new technology, without excessive risk to protected portions of the network.

FICS-IT components may be separate machines or functional parts of existing machines. The idea is to apply the functional, connection and information protection to the system resources. Resource management is at a granularity that is as fine as is feasible. If the component is at an end-system server (Figure 1), then it is possible to manage access to individual files and programs, using the triples of User, Program, Data as defined in the VISE work.

4 Thomsen, Dan, SCC, presentation and discussion at CMAD III, Sonoma, CA, January 1995

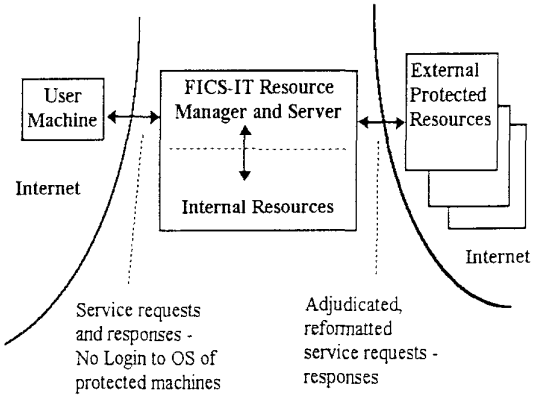

Figure 1: FICS-IT Server at an End System

Physical connectivity does not necessarily imply data communications. Though the protected user machine is connected physically to the Internet, it employs FICS-IT connection security to refuse network connections except from the FICS-IT server. The user "sees" only the FICS-IT server and is not able to access the protected external resources directly. Requests for external resources are interpreted by the FICS-IT server and translated into appropriate requests to resource holders behind the server. These may include file servers, data base servers, etc.

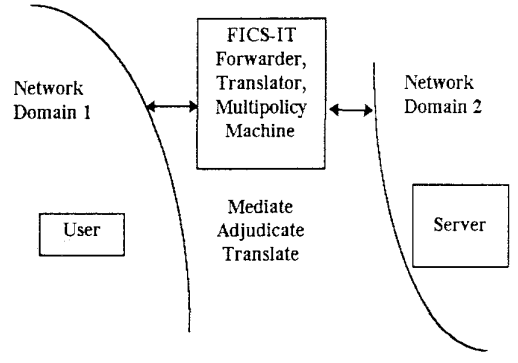

Figure 2: FICS-IT Forwarder at a Domain Interface

There are other possible and useful placements for FICS-IT components. Figure 2 shows a FICS-IT placed between two network domains. In this position, it acts somewhat like the Firewalls described in Nelson's Mutual Suspicion paper [4] or the Application Gateway firewalls described by Cheswick and Bellovin [9]. The current firewall approach is much more limited than FICS-IT, however, because it does not provide much functional access control and because it cannot handle multiple policies. The FICS-IT forwarder is also configured with as much knowledge about the resources behind it as possible. It may primarily provide the Connection Security service of permitting or disallowing information flow, but it may also be capable of much more sophisticated service. It may even act much like the FICS-IT server, knowing the remote users' privileges, and adjudicating and reformulating service requests and responses. We see this version of FICS-IT as a natural extension of a FICS-IT server.

Security Policies and Resolution

System security policies reflect a set of management decisions about allocation and use of system resources. Only some of these decisions can be made automatically in real time during operation of the system. Others are designed into the system or must be enforced by people. The machine-enforceable policies can be expressed as a set of rules determining use of resources. In "classic" (TCSEC [2]) secure systems, these are rules for access of subjects to objects, expressing MAC and DAC access control policies. In our FICS-IT approach, based on the GTE VISE [7] model, we also include rules about access to specific programs or system functions that operate on specific information. In addition, we include rules for connectivity between our system and other systems on the network.

If several policy rules apply to the same resources, then conflicts are possible and must be resolved. The usual security solution is to resolve access requests by denying the request unless all rules are met. Hosmer's Multipolicy [4] work includes more flexible resolution of conflicts through meta-policies. This added flexibility allows the codification of policies involving priority of need and availability, and the integration of these policies with pure access control.

Conclusions

The infinite network is and will continue to be heterogeneous. As new information and communication systems, protocols and methods are

developed, security mechanisms and architectures must continue to change with them. The change and heterogeneity do not mean that understanding of network architecture is extraneous; there is structure to the network, at least locally. They do mean that we cannot solidify our security solution, but must continue to examine and understand the relationships between network functionality and network vulnerability.

The problem of security in large networks is complex and multifaceted. It does not have a single, simple solution. There is no single panacea for securing the infinite network, and we do not expect to find one. Our efforts so far have convinced us of the feasibility of reducing the risk of connecting valuable assets to an open network.

We found in our research that there is presently a dichotomy between theory and practice in network security. On the one hand, the theory, based on computer security, forbids connections to large networks and the dissemination of information outside closed security domains. The connectivity is theoretically too risky, and theories based on computer security models do not offer any effective strategy for mitigation of that risk. On the other hand, large numbers of connections are being made without a basis for understanding the implications of the network interactions. Firewalls and some host protection mechanisms are mandated and used in a somewhat arbitrary fashion, based on the availability of products rather than on established security principles.

The FICS-IT paradigm is a step in the direction of integrating theory and practice. FICS-IT does not attempt to provide a rigid solution, but it does offer some structure, a framework for examining the security problems of networked systems and for designing more security into those systems. Our goal is to use this structure to develop guidelines that are applicable to the design of specific systems and components, and also to develop a collection of modular, customizable FICS-IT components. Use of these components will reduce network security risk and still allow the flexibility of function needed and demanded by network users.

A paradigm, such as FICS-IT, is an abstract concept. More work is needed to apply this concept to specific problems and demonstrate its usefulness.

An important area for future work is the development of both broad and specific guidance for network security. This guidance should cover policies, network functionality, security principles and mechanisms, and methods of assessing security risk. It should include both general principles and specific analysis of some current and new network solutions. Network and security technology are changing very quickly; it is important to allow and provide guidance for the incorporation of new services and features as they become available. Some security principles are basic; a knowledge of these is a foundation for analyzing new elements.

References:

1. National Computer Security Center, *Trusted Network Interpretation of the Trusted Computer System Evaluation Criteria*, July 1987.

2. *Department of Defense Trusted Computer Security Evaluation Criteria*, DoD 5200.28-STD, National Computer Security Center, December 1985.

3. R. Nelson, D. Becker, J. Brunell and J. Heimann, "Mutual Suspicion for Network Security," *Proceedings of the 13th National Computer Security Conference*, Baltimore, MD, September 1990.

4. Hilary Hosmer, "The Multipolicy Paradigm," *Proceedings of the 15th National Computer Security Conference*, Baltimore, MD, October 1992.

5. David Bailey, "Managing Complexity in Secure Networks," *Proceedings of the ACM SIGSAC New Security Paradigms Workshop*, Little Compton, RI, September 1992.

6. D. Clark and D. Wilson, "A Comparison of Commercial and Military Computer Security Policies," *Proceedings of the 1987 IEEE Symposium on Security and Privacy*, Oakland, CA, April 1987.

7. C. Limoges, R. Nelson, J. Heimann, D. Becker, "Versatile Integrity and Security Environment (VISE) for Computer Systems," *Proceedings of the New Security Paradigms Workshop II*, Little Compton, RI, August 1994.

8. DISA/CISS, "*DoD Goal Security Architecture*," Version 1.0, August 1993.

9. William R. Cheswick and Steven M. Bellovin, "*Firewalls and Internet Security*," Addison-Wesley Professional Computing Series, 1994.