Published in the Proceedings of the 21st National Information Systems Security Conference, Crystal City, VA, pp 421-431, October 1998

High Assurance Multilevel Services For Off-The-Shelf Workstation Applications

Cynthia E. Irvine
Naval Postgraduate School

James P. Anderson
James P. Anderson Co.

LT. Dion A. Robb, USN
SPAWARSYSCEN, Charleston

Cpt. Jason Hackerson, USMC
Naval Postgraduate School

ABSTRACT

The need for multilevel secure (MLS) systems still exists yet, the popularity of desktop systems has resulted in the imposition of new requirements. To be useful, a system must employ commercial-off-the-shelf (COTS) operating systems and office productivity software. We describe the preliminary architecture for a COTS-driven local area network that will provide MLS services to users while permitting them to employ standard office productivity tools on standard workstations. Our ongoing development centers on the provision of multilevel mail and messaging to the desktop.

1. Introduction

The problem of timely access to information at multiple security levels has challenged the computer security research and engineering community for several decades. During the 1960s and 1970s attention was focussed on the development of high assurance security kernels by vendors. The resulting systems permit controlled sharing of sensitive information by users at multiple security levels. Often these reference validation mechanisms present primitive interfaces upon which the service-rich interfaces demanded by users and modern applications must be built.

By the early 1980s researchers had turned their attention to the support of complex applications in the context of multilevel systems. Challenges included allowing users to view multiple security levels simultaneously, while minimizing, if not completely avoiding, modifications to underlying security kernels used to enforce mandatory security policies. Considerable success in demonstrating that complex functionality can be provided outside of the reference validation mechanism was obtained by several projects including both database systems, e.g. [11], and modern file systems, e.g. [9]. In most cases, these results applied to custom-built systems. This precluded their rapid evolution and update as vendors developed new, more capable applications.

Personal computers and workstations have become increasingly important tools of office automation and productivity during the last decade. The use of commodity software exploded and automated information technology was no longer the province of scientists and engineers, but of the entire work force. When connected to local area networks (LANs), these desktop systems can be supported by centralized hosts providing a wide variety of services such as mail, networking, accounting, engineering applications, etc. DoD is making enormous in commercial-off-the-shelf (COTS) software and commodity PC products. This shift to COTS has lead to new requirements: the ability to incorporate patches and updates to existing COTS products and the ability to enlarge the desktop-based software suite as new products become available. Unfortunately, multilevel security has been left in the wake of these changes.

Today, the problem for DoD systems includes not only the provision of control of access to and movement of data based on fixed sensitivity levels, but the preservation of compatibility with COTS application software as well. When security is paramount, it has been achieved at the price of compatibility. In contrast, when compatibility with COTS applications takes precedence, then instead of employing trusted systems for timely sharing of information, each access class is relegated to a separate information system. Independent system-high enclaves are established and sharing is

achieved through: manual, "sneaker-net" techniques; automated guards for which no notion of sufficiency or completeness with respect to security policy enforcement can be demonstrated; or replication systems [8]. If the number of access classes to be supported is high, then the use of physical separation becomes cumbersome, costly, and inflexible in terms of both equipment and administration.

A solution is the implementation of a COTS-driven LAN supporting a family of high assurance trusted servers [1]. Through the use of existing evaluated high assurance components, it is possible to develop a security architecture that supports a fully functional multilevel secure environment for workstations with a high level of security for information. A major advantage of this architecture is that it is intended to provide compatibility with COTS PC or workstation operating systems and applications. Several factors contribute to the feasibility of our effort:

- The use of "diskless", rather than multilevel, workstations eliminates the need to develop true multilevel workstations enforcing mandatory security policy. Here "diskless" means that volatile storage at the workstation may be purged under trusted computing base (TCB) control; workstations may include non-volatile, read-only disks.
- The use of "Wintel" workstations making the LAN attractive to end users, who will be able to continue to use their favorite applications, but within the context of a high assurance multilevel environment.
- Use of an evaluated, commercially available, high assurance TCB reduces both risk and cost, while leveraging already significant DoD investments in high assurance products.

Here we will describe an ongoing effort to develop components for a high assurance multilevel secure local area network constructed using evaluated products as well as commercial-off-the-shelf workstations and office productivity software.

To demonstrate the feasibility of a family of multilevel secure servers, we have chosen to focus on a mail service. Ongoing academic research in the area of mail server user- and transport-agents permits us access to existing source code as a basis for the server application. Ultimately, we envision a variety of applications supported by one or more high assurance servers.

The prototype system being built consists of the following components: a trusted mail server based on a adaptation of a free mail software to a high assurance TCB; a Wintel-based COTS workstation and/or a thin, *network computer* client; a TCB extension at the client to provide a trusted path; and components to provide secure communications between the workstation and the trusted mail server. Users accessing the system at the client will be able to execute unmodified COTS mail interfaces. Once the TCB has been achieved on the LAN, it will be possible to enhance the high assurance server to support high assurance label-based selection of encryption for messaging services. This trusted communications service (TCS) can provide high assurance that the correct cryptography is applied to sensitive data planned for the exportation of mail beyond the server.

The organization of this paper is as follows. Overall system requirements are presented in Section 2. Section 3 will provide a description of the overall architecture including the high assurance base, the mail server application, the client workstations, and the TCB extension. We will provide a comparison of our architecture with a few other approaches in Section 4. In Section 5 we will outline continuing work on this effort and possible future extensions. A summary in Section 6 will complete this paper.

2. System Requirements

Here we present high-level system requirements. They fall into two categories: functional requirements and non-functional requirements. Functional Requirements for the server are:

- COTS Clients: The server should support a COTS PC client that has been enhanced with a TCB extension. COTS operating system software should be unmodified,
- COTS Application Independence: The server environment must support unmodified COTS mail client software,
- Client Extensibility: The server should identify an engineering path for introducing

TCB extensions to additional client platforms, and

- Server Application Extensibility: An engineering path for extending the server to support additional server-based applications should be identified.

Non-Functional requirements driving our system architecture include: Multilevel Security, Discretionary Access Control, Cost, Authentication, Audit Support, Reliability, and Performance.

3. Preliminary System Architecture

This section is intended to provide a top-level description of the trusted server architecture. The architecture consists of two major elements: the LAN and the non-local messaging support. We sketch both here, but will provide detailed discussion only of the LAN, the principle focus of our current efforts. We are using a mail server as our prototype application.

The goal of the trusted server development is to utilize a LAN with a trusted server and workstations to provide a secure mail processing environment in which user functions or programs can be securely integrated at virtually any time while still preserving the security of existing data. This will be achieved by defining secure operation in terms of allowed accesses.

The overall system architecture, shown in Figure1, consists of the following components:

- A high assurance mail server composed of untrusted mail server instances constrained by a high assurance TCB, that will ensure enforcement of critical security policy at the server.
- Untrusted COTS client mail applications that will issue requests to the server. A security level will be associated with each client instance and a server instance at an identical security level will handle the client's request. The high assurance TCB will permit server instances to respond to requests for information at or below the security level of the instance.
- A messaging server will insure the proper labeling of information leaving and entering the system. Label-based encryption can be selected to ensure the protection of information transiting unprotected networks.
- Clients may communicate with the server from protected system high enclaves, across the multilevel LAN, or across unprotected networks. For each of these cases, the session level of the clients will be determined.
  - The security level of a protected, system high enclave connected directly to the server will be statically defined when the system is configured.
  - Each LAN client will be fitted with a TCB extension that will provide high assurance services for object reuse, a trusted path to the high assurance server, and support for client-server session cryptography. LAN clients will be "diskless" workstations or network computers.
  - Across unprotected networks, sensitivity labels will be bound to the transmitted information. Network encryption will be used to protect communications.

3.1. Security Policy Enforcement

In our architecture, all mandatory access control policy enforcement is delegated to the high assurance TCB used as the underlying server platform. The high assurance base has been chosen so that it enforces mandatory policy using a label-based mechanism that maps to a lattice [5]. Since all mandatory policies can be expressed as a lattice, a label-based enforcement mechanism can be used in all cases. The initial implementation of the system will enforce a DoD secrecy and/or integrity policy as formally stated in the Bell and LaPadula [2] and a Biba [3] models, respectively. Although it is not envisioned that the TCB will enforce a variety of mandatory security policies simultaneously, it is desirable that the TCB be adaptable. Thus through modifications to its non-discretionary security policy enforcement module, alternative mandatory policies can be enforced, such as commercial secrecy or integrity policies in which information can be separated into either hierarchical and/or non-hierarchical equivalence classes [10] or mandatory role-based policies. This would enhance prospects for commercialization of the server effort.

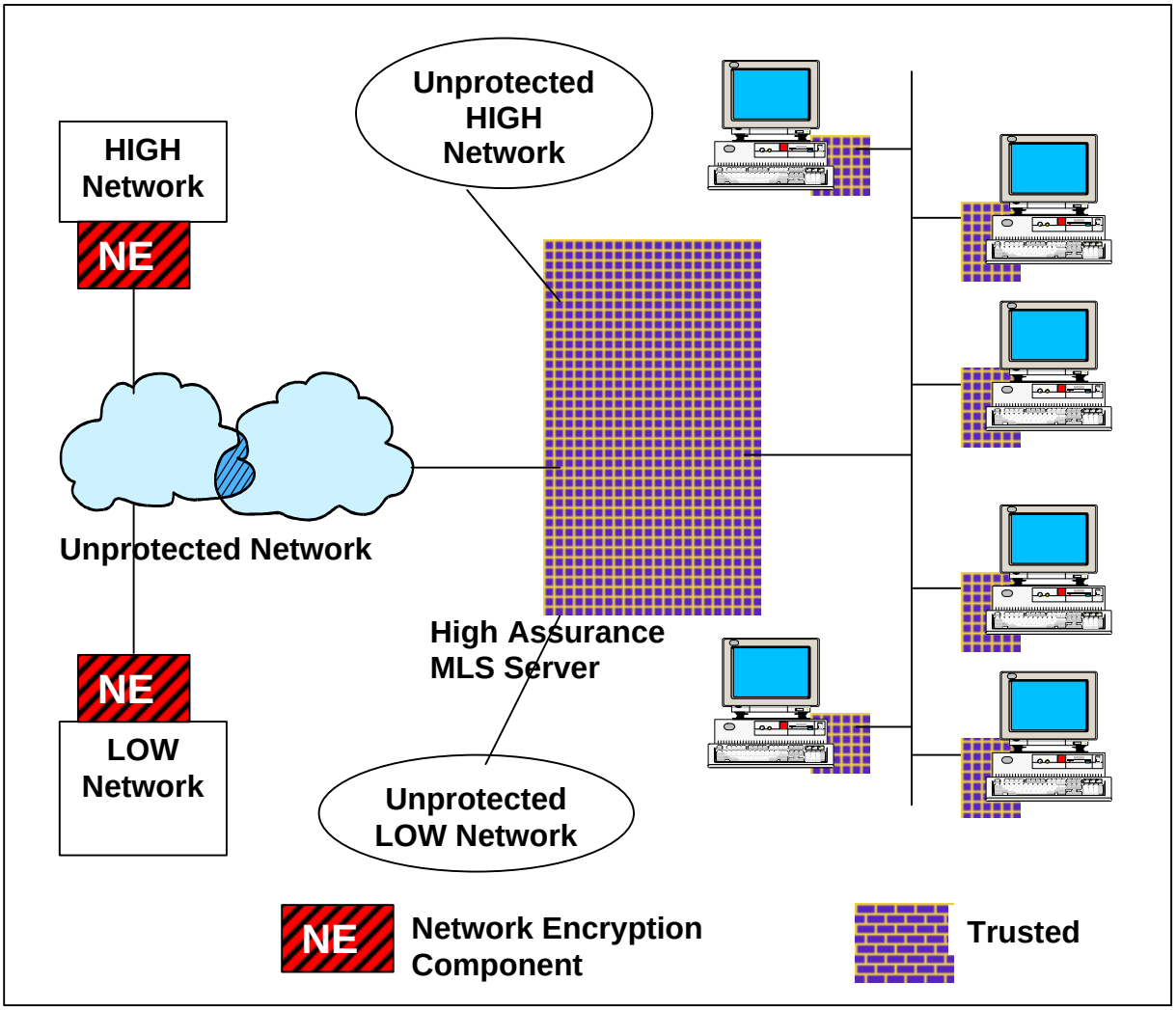

Figure 1. High Assurance Trusted Server Architecture

Enforcement of mandatory policy is allocated to the high assurance server. In addition to relying on the high assurance TCB to enforce mandatory confidentiality policy, its effectiveness as an integrity enforcement mechanism is also being explored. Two forms of integrity will be investigated: data integrity [3] and program integrity [14].

Since the mail server applications under consideration do not enforce discretionary security policies, there is no need to layer discretionary policy enforcement mechanisms using TCB subsets [15]: we depend entirely upon the discretionary mechanism provided by the underlying high assurance base. We note, however, that our architecture would not preclude the use of TCB subsets and balanced assurance. Beyond the problem of mandatory policy enforcement, there are many factors that affect server security. Each service application has a unique set of security issues. Those associated with the configuration of the application are beyond the scope of this investigation. Others relate to the software engineering rigor applied to the application during its creation. Some mail programs are notoriously fragile and we are exploring the most effective techniques to leverage the high assurance TCB protection mechanisms to mitigate some of these vulnerabilities.

3.2. Trusted Mail Server

The Trusted Mail Server will consist of a high assurance TCB and untrusted mail server instances constrained by the TCB, which enforces critical

security policy. The advantages of this approach are that the server will support sharing and labeling and will be functionally equivalent in terms of application-level protocol support to an existing popular package. This provides compatibility with existing COTS software.

3.2.1. High Assurance Base

The choice of the high assurance base for the design, adaptation, and implementation of the server and associated multilevel secure communication services is of critical importance to the favorable outcome of this effort.

It is defined by the broad properties needed for a viable commercial product. Our definition of a high assurance base is a TCB that is already on the Evaluated Products List (EPL) with a Class B3 or higher digraph based upon an evaluation against the TCSEC [12] or the TNI. We considered ONLY products on the EPL. We have selected the most recent model of the Wang XTS-300. Both business and technical considerations affected our deliberations.

The principal non-technical consideration was the availability of software and hardware maintenance support to insure that the systems could be held in an evaluated configuration.

The principal technical consideration was the ability of the high assurance base to enforce security policy. The high assurance TCB, by virtue of the protection domains it creates, provides confidence that any possible malicious code will neither cause the exfiltration of sensitive data nor the corruption of high integrity information. Other factors included ease of use, available software tools, and interfaces.

3.2.2. Mail Service Application

If the mail service application is instantiated within single level subjects at several access classes, the result is a multilevel mail server supporting controlled sharing of individual files. Of course, we must modify the application to view files dominated by the subject's access class. The user view of the system is of mailboxes at or below the user's current session level. Our intention is to analyze the application structure required for a mail server consisting of single level instances intended to manage a multilevel structure of mail resources. Based on this analysis a preliminary free-ware mail program can be adapted to the multilevel environment. Users will log on to the server, establishing an identity for audit and access control purposes. The mail server will be the locus of several other capabilities including security logging and audit, as well as secure downgrading and connection services.

The problem of adapting an existing server to the high assurance base can be divided into two sets of issues: those concerned with moving the server software to any new platform, and those specifically concerned with targeting it to a high assurance platform. While these issues are interdependent, they are discussed separately below.

Earlier work on application partitioning for high assurance platforms, e.g., [9][11] is being leveraged. One approach to separating information at different security classes would be to utilize existing server software that permits flexible underlying information storage while presenting a unified view to users. Software able to support multiple "root" directories would be particularly well suited to this approach.

In our approach [6], both the mail spool and the mail boxes of individual users will be stored at the high assurance server. We take advantage of the existing file system provided by the XTS-300. In a series of experiments using the free-ware *elm* user-agent software, we explored several issues related to the management of mail on the high assurance platform.

Movement of mail to the server mail spools

We anticipate two methods to move incoming mail to mail spools. First, if the external connection is single level, then all incoming mail at that connection will be implicitly labeled with the access class of the connection. An untrusted subject executing at the access class of the connection can move mail into the spool.

If a multilevel connection is made for incoming mail, then a trusted subject will be required. It must be able to read and write at a range of security levels. Its task is to examine the security label on incoming mail and place it in a mail spool at the correct security level. To minimize the complexity of the trusted subject, mail spool daemons at each security level instantiated in the

system can read mail from buffers used by the multilevel subject. The use of a trusted subject to sort mail by access class is justified in that it is not complex and satisfies the definition of a "classic" security guard [4].

Mail spool to individual *in-boxes* movement

The movement of mail from the spool to an individual's in-box is usually invoked by a *get-mail* command issued by the client or user agent. If the client is running at a high security level, then the user should be able to see new mail at all access classes dominated by the session level.

How can mail be moved to the *in-boxes* without involving a complex trusted subject? We have chosen to implement mail daemons that move mail from the mail spools at each access class to user mailboxes at each access class. Our approach differs from typical mail movement systems, in which users invoke some *get-mail* command, but we believe that we are justified since all mail remains stored at the high assurance server.

Mail daemons that move messages from the mail spool to mailboxes at each access class eliminate the requirement for a *get-mail* trusted subject. Because mail will be stored in the mailboxes at the high assurance server, mail will not be moved to an inappropriate client, so there is no requirement to leave the mail on the spool. For users who desire more control, *get-mail* can be invoked at individual session levels. The possibility of per user configurations of these services will be explored.

Marking and deleting mail within mail folders

Typically, as a user reads mail, the following things can occur:

- The mail is marked as read,
- The mail is moved to an alternate mail folder, e.g., if a user is storing mail from a correspondent named Bill, then it may be moved from the general *in-box* to a folder called *mail-from-bill*.
- Unwanted mail is deleted.

How will this take place in a multilevel environment?

First, we have made a high-level decision that mail will be moved to other mail folders only when the user's session level matches that of the folders.

What about marking or deleting mail? If the session level is HIGH, then marking or deleting mail at lower access classes requires a disallowed write down. To avoid this, there are two alternatives. First, users can be forced to read and mark mail at each security level. This is a choice that will minimize the compromise of sensitive information. It is also a choice that could result in significant user frustration as mail would have to be reprocessed at each sensitivity level.

An alternative is to have the server create lists intended for downgrading to lower security levels. These will contain identifiers for the mail to be marked as read and or deleted. Each list can be downgraded by a trusted subject invoked by the user. Untrusted mail subjects can interpret the lists at each access class, and automatically mark and delete mail appropriately.

Substantial effort is still required to support a complete mail system including: continued exploration of user-agent issues and the implementation of transport-agent functions.

3.3. Client Workstations

COTS workstations are being fitted with a TCB extension that is intended to provide three security-relevant functions: establish a trusted path between the user and the TCB, enforce the object reuse provision of the extended TCB, and enforce data movement and access controls based on a system of labels indicating the user's authorization and data sensitivity.

To insure that object reuse requirements [12] are met, workstations will be "diskless," with sufficient RAM-disk capability to support a wide variety of user applications. The workstation TCB extension will satisfy object reuse requirements by ensuring that RAM and other volatile primary and secondary storage at the workstation are purged with each change of session level or new user login at the workstation.

Each user will be able to manipulate data from the server in COTS applications (e.g. Word Perfect, Lotus 1-2-3, etc.) at his/her authorized security level. Any data dominated by a user's session level can be read from the server. Writes to the server will be at the current session level.

As constrained by the security policy enforced by the mandatory component of the high assurance

TCB, any data dominated by a user's session level can be read from the server. Writes to the server will be at the current session level. The number of levels supported will be a configurable function of the underlying TCB supporting the server.

The use of "diskless" workstations logically leads to a recent commercial development: the network computer.

3.3.1. Network Computers: The Ultimate Diskless Clients

The use of network computers is an attractive advanced option for our architecture. These devices, proposed in 1996 by Larry Ellison, of Oracle, Inc., take advantage of the World Wide Web or a corporate Intranet. Java is envisioned as the initial language to support network computer applications. Although these devices have yet to achieve widespread popularity, they offer several advantages for many DoD environments:

Reduced cost

It has been estimated that although the hardware cost for a typical PC for office automation is on the order of $2000, the cost of installing and maintaining software on these devices is on the order of $7000 annually. The network computer is intended to reduce those costs dramatically by allowing users to download executable content from a centralized server.

Centralized control of software updates

This will permit much more effective configuration management in a distributed computing environment. The complexity of making the transition between software versions can be simplified. Centralization will allow new applications to be introduced more easily.

Reduced Employee Distraction

Centralized services can be used to limit the number and types of applications available to employees during working hours. Game playing and other distractions can be reduced. Employees will be better able to focus on their jobs.

Smartcard Interfaces

Smartcard technology is being incorporated into network computers to personalize them for the user and to provide security functionality such as authentication support for communication with the server. It is interesting to note that industrial alliances are already being forged to incorporate smartcard technology into network computers [13]. Without sufficient security engineering, the smart cards will be a bypassable mechanism [6] which would render them inadequate for use with sensitive information. The incorporation of the smartcard interface into the TCB extension could insure a high assurance trusted path between the client and server.

These innovations are likely to be pursued by DoD, however without sufficient security design and engineering, there is a significant risk that sensitive information will be compromised or that untrustworthy applications will result in malicious executions at clients.

3.3.2. TCB Extension

What requirements must be imposed on a workstation in order to use it in our architecture? For a monolithic system, the device employed for user interaction with high assurance systems can be a dumb terminal. All processing of information is isolated within the high assurance system. In our architecture, information will be passed from the high assurance server to a fully functional PC. Unless the PC is constrained, there are many opportunities for the exfiltration of sensitive information.

We intend to treat PC-based workstations as if they were "diskless" network computers. Each COTS workstation will be adapted to contain a TCB extension providing the following security-relevant functions:

Secure boot of workstation

The TCB extension must insure that it has initial control of the client and its resources at the time of boot and that control over security critical actions is maintained throughout the client session.

User-to-TCB trusted path

The TCB extension must provide a trusted path between the high assurance workstation and the client for the purposes of identification and authentication as well as session level negotiation. It must be established as part of start-up and invocation using a secure attention key must be available at all times. To achieve the trusted path, there must be a protocol between the workstation TCB-extension and the trusted server TCB so that

the server TCB can unambiguously establish that it is communicating with the workstation TCB. Ultimately, use of cryptographic techniques to embed a "secure attention key" in the data stream, which is scanned by the server TCB that identifies it as a request for the trusted path provides TCB-to-TCB communications with high confidence. Limited resources may require that our prototype demonstration use an interim alternative for the trusted path: enforcement of a low layer protocol to uniquely mark TCB-to-TCB communication.

Enforce object reuse at the "extended" TCB

The TCB extension must insure that requirements for object reuse are met. This will be achieved by purging all volatile primary and secondary storage when users negotiate a new session level.

Session-based cryptography on the LAN

Although control over security critical actions have been identified, control over I/O on the LAN, because of its importance, is specifically identified as a responsibility of the TCB extension.

The TCB extension is based on a programmable controller card. The architecture of the workstation TCB extension is intended to maximize compatibility with COTS products. We are building on work already completed on products for workstation control by using the MESA/MEMS products [15]. These are being adapted to provide the TCB services required for this effort. Our project currently has two MESA/MEMS controller boards operating in PCs running DOS Windows 3.1. This permits us to build upon an existing engineering effort rather than pursuing, at greater risk, the development of a simple TCB extension especially for this effort. The development of a intermediate software-based design prototype on a Wintel-based PC will permit initial examination of design choices. Use of an add-on card that is self-contained and requires no modification to the workstation in order to function materially reduces both effort and risk.

4. Comparison with Other Approaches

In this section we will describe other approaches to achieving high assurance multilevel security for practical network architectures. Recall that our objective is to permit any authorized user to be able to login onto a workstation with a selected session level and be able to access all information dominated by that session level.

4.1. High Assurance at the Workstation

An architecture we rejected was one in which the user negotiates a session level at the workstation and then connects to a host at the negotiated level. Such a system has several drawbacks.

First, the user will have to toggle back and forth between security levels in order to access information at the different levels. The problem here is that the toggling must be invoked not only for write access to information at differing security levels, but for read access as well.

Second, the user will not be able to view information at multiple security levels simultaneously unless information at multiple security levels is stored at the workstation. Clearly, an architecture of this type will be vulnerable to Trojan Horse attacks unless the workstation is itself trusted. Let us explore for a moment why this is so: Suppose that the user sets the session level to SECRET and reads information into workstation-local memory. If the user then resets the session level to UNCLASSIFIED, then it is possible that SECRET information could be illicitly moved from the workstation to the UNCLASSIFIED host. There are two ways to prevent such Trojan Horse attacks. First, the change in session level can force the workstation to be purged of all stored information or, second, the workstation must enforce the mandatory security policy by partitioning the information by security level. The former does not permit the user to view information at multiple security levels simultaneously. This would severely limit the usefulness of the workstation as a component in a multilevel architecture. On the other hand, the latter choice results in a requirement for a high assurance multilevel secure workstation. The likelihood that such a high assurance MLS workstation would keep pace with rapid commercial developments in the office productivity arena is low. Thus users would find such a solution unacceptable since it would bar them from both the most up-to-date hardware and rapidly evolving software products.

4.2. Static Workstation Security Level

Hinke suggested the notion of a high assurance server to provide a locus of multilevel control to single level clients [7]. In that design sketch clients were relegated to a single level and were connected to the multilevel server via single level network links.

Although this architecture may be useful in certain static situations, it does not provide the flexibility inherent in that described in this paper. By restricting the client to a single level throughout its lifetime, users will be required to access multiple clients in order to manipulate information at several security levels. Thus, the problem we are trying to solve, viz. multiple clients on a single desktop, is not addressed. Our requirement for a choice of security level at the client forces us to address several challenges. These include: multilevel connections between clients and the server, with the resulting requirement for protection of network communications; high assurance, server-based identification and authentication for client session level negotiation; and provision of a trusted path between each client and the server.

4.3. Chip-Level Filtering

The use of a chip-level filtering mechanism in a PC to enforce aspects of security policy re-introduces to the workstation many of the challenges associated with the development of high assurance policy enforcement mechanism. In order for the filter to work effectively, information within the PC must be labeled so that the filter can process it appropriately. Anything less will result in an approach to policy enforcement based on heuristics.

4.4. Replication Architectures

Replication architectures [8] provide a simple technique to achieve near-term multilevel security by copying all information at low security levels to all dominating levels. On a small scale one can expect them to work rather well; on a large scale, their usefulness is rather problematic. The preponderance of information used in DoD today is either unclassified or designated sensitive but unclassified (SBU). Multilevel requirements have always been to provide decision makers with simultaneous access to information at all security levels. In a replication architecture, this implies massive amounts of replication. In the commercial sector, the ratio of proprietary to less sensitive information is similar.

5. Continuing and Future Work

Several interesting challenges lie before us in this effort. They include:

- Adaptation of a transport agent to the multilevel secure environment.
- Protection of data in transit between workstations and servers. Software techniques will be used initially. Possible protection mechanisms include cryptography using PCMCIA-based cryptographic peripherals, such as Fortezza cards, or other hardware approaches.
- Covert channel considerations in a distributed environment and possible relationships to the use of encryption.
- Examination of evaluation and assurance issues associated with the combination of trusted components into a coherent network architecture.
- Design of a multilevel subject for encryption of outgoing information .
- Design of a reliable communications subject to execute on the high assurance platform. The purpose of this subject will be to demultiplex labeled requests for server content to single level subjects at the level of the request.
- Performance issues associated with multilevel multiplexing at the trusted server.

5.1. Communications Services

There may be a need for external users to connect to the server across a WAN. These users need to be able to transmit protocols and data over a network with trust throughout the transmission path and, for accountability purposes, an adequate level of audit as part of the transmission process.

Once the TCB has been extended from the mail service to the workstation or network computer so that a trusted path is available at each desktop, the technical foundation to enforce a mandatory security policy and the command/control over the

movement of data outside of the work unit will have been established. It will be possible to provide a crypto-server to encrypt data using cryptographic functions selected on the basis of the data's security label. The system will provide high assurance that the information is encrypted using algorithms and keys commensurate with its sensitivity. Thus, it will be possible to create an E-mail server, for example a Defense Message System (DMS) server that employs label-based controls to select the cryptography to apply to messages.

Although current hardware-based encryption technology, with a processing throughput of less than 1 Mbit/second, appears to be inadequate for handling hundreds of workstations on a LAN, future developments, projected to process 10 Mbyte/sec (80 Mbit/sec) will provide the necessary through-put. Using a crypto-concentrator on the server, these new developments may be able to handle all of the messages from the LAN. Several possible cryptographic configurations are possible. For example, the crypto concentrator could direct mail messages to particular hardware encryption chips based on the keys to be used for the message, or, if the keysets are small and all hardware encryption chips have the same keyset, the concentrator could perform "load balancing" across what would effectively be a crypto-multiprocessor. The use of high speed, effective cryptography would eliminate the need for a trusted LAN since cryptography would be provided both at the workstation and on the server. It is predicted that this new generation of crypto-concentrators will be available in the near term. With the recent cancellation of work on Fortezza 2.0, which was to provide Type I encryption for classified information, careful analysis of available government and commercial options will be required as part of this investigation. In the interim, current technology can be employed for test purposes on LANs with a relatively small number of Wintel work stations.

We believe that the use of a centralized TCB server will provide sufficient performance to support the message traffic for the initial prototype LAN. Future research may be required to the determine how the mandatory TCB might be distributed for performance enhancement. (Note that the use of a multiprocessing TCB presents significant performance advantages for the initial system by allowing the mail server and crypto-server functions to be allocated to different processors.) Among the future challenges would be security administration in a highly distributed architecture.

6. Summary

We have described our preliminary design for a COTS-based LAN providing high assurance multilevel security for user information. Huge investments in COTS software and user-built applications based on commercial APIs can be amortized within the context of our architecture. No requirements for special source licenses are required since no modification of COTS software is required.

The capability to provide users with the ability to use unmodified commercial PC operating systems and office productivity software is a key aspect of our architecture. This approach has several benefits:

- Existing commercial applications will be immediately usable within the context of the system
- Users will be able to maintain applications using standard upgrades and patches
- Standard PC-based operating systems will be used without modification
- The use of diskless "Wintel" workstations provides users with an attractive platform.
- An evaluated COTS high assurance platform simplifies the effort and builds on earlier DoD investment in trusted systems.
- A TCB-extension card eliminates the need for multilevel secure workstations.

In summary, the marriage of high assurance COTS products and current technology can create a multilevel system insuring the security of information while providing end users with the continued use of their favorite COTS workstation applications.

Acknowledgements

The authors would like to thank LCDR James P. Downey, USN for his many hours of creative effort devoted to this research while a student at

the Naval Postgraduate School. He made an invaluable contribution to the project.

References

1. Anderson, J. P., *A Summary of TFS Requirements*, Tech. Report, James P. Anderson Co. Apr, 1989.

2. Bell, D. E., and LaPadula, L., Secure Computer Systems: Mathematical Foundations and Model, M74-244, MITRE Corp. Bedford, MA, 1973.

3. Biba, K, J., Integrity Considerations for Secure Computer Systems, ESD-TR-76-372, MITRE Corp., 1977.

4. D. L. Brinkley and Schell, R. R., Concepts and Terminology for Computer Security, in Information Security: An Integrated Collection of Essays, ed. Abrams and Jajodia and Podell, IEEE Computer Society Press, Los Alamitos, CA, 1995, pp. 40-97.

5. Denning, D., A Lattice Model of Secure Information Flow, *Comm. A.C.M.*, Vol. 19, No. 5, 1976, pp. 236-343.

6. Downey, J.P., and Robb, D.A., Design of a High Assurance Multilevel Mail Server (HAMMS), M.S. thesis, Naval Postgraduate School, Monterey, CA, 1997.

7. Hinke, T., The Trusted Server Approach to Multilevel Security, in *Proc. Computer Security Applications Conference*, Tucson, AZ, 1990, pp. 335-341.

8. Froscher, J.N, Kang, M, McDermott, J., Costich, O. and Landwehr, C.E.,.A Practical Approach to High Assurance Multilevel Secure Computing Service, *in Proc. 10th Comp. Sec. Appln. Conf*, Orlando, FL, 1994, pp. 2-11.

9. Irvine, C.E., A Multilevel File System for High Assurance, in *Proc. IEEE Symp. on Sec. and Privacy*, Oakland, CA, May, 1995, pp. 78-87.

10. Lipner, S.B., Non-Discretionary Controls for Commercial Applications, in *1982 Proc. IEEE Symp. on Sec. and Privacy*, Oakland, CA, 1982, pp. 2-20.

11. Lunt, Teresa F., Schell, Roger R., Shockley, W.R., Heckman, M., and Warren, D.F., A Near-Term Design for the SeaView Multilevel Database System, in *1988 Proc. IEEE Symp. on Sec. and Privacy*, Oakland, CA, 1988, pp. 234-244.

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

13. Network Computer, Inc., Oracle's NCI and Schlumberger to Provide Smartcard Technology for Network Computers, URL http://www.nc.com/pr_schlum.html, June, 1997.

14. Shirley, L.J., and Schell, R. R., Mechanism Sufficiency Validation by Assignment, in *1981 Proc. IEEE Symp. on Sec. and Privacy*, Oakland, CA, 1981, pp. 26-32.

15. Shockley, W.R., and Schell, R.R., TCB Subsets for Incremental Evaluation, in Proc. 3rd. *AIAA Conf. on Computer Security*, December, 1987, pp. 131-139.

16. Spyrus, System Architecture Document (for Media Encryption Management System), prepared for Office of Research and Development, Spyrus, 2813 Junction Avenue, Suite 110, San Jose, CA, August 1996.