Audit Trails. As discussed earlier, logical access controls can be difficult to implement correctly. Also, it is sometimes *not possible* to make logical access control as precise, or fine-grained, as would be ideal for an organization. In such situations, users may either deliberately or inadvertently abuse their access. For example, access controls cannot prevent a user from modifying data the user is authorized to modify, even if the modification is incorrect. Auditing provides a way to identify abuse of access permissions. It also provides a means to review the actions of system or security administrators.

Identification and Authentication. In most logical access control scenarios, the identity of the user must be established before an access control decision can be made. The access control process then associates the permissible forms of accesses with that identity. This means that access control can only be as effective as the I&A process employed for the system.

Physical Access Control. Most systems can be compromised if someone can physically access the machine (i.e., CPU or other major components) by, for example, restarting the system with different software. Logical access controls are, therefore, dependent on physical access controls (with the exception of encryption, which can depend solely on the strength of the algorithm and the secrecy of the key).

17.7 Cost Considerations

Incorporating logical access controls into a computer system involves the purchase or use of access control mechanisms, their implementation, and changes in user behavior.

Direct Costs. Among the direct costs associated with the use of logical access controls are the purchase and support of hardware, operating systems, and applications that provide the controls, and any add-on security packages. The most significant personnel cost in relation to logical access control is usually for administration (e.g., initially determining, assigning, and keeping access rights up to date). Label-based access control is available in a limited number of commercial products, but at greater cost and with less variety of selection. Role-based systems are becoming more available, but there are significant costs involved in customizing these systems for a particular organization. Training users to understand and use an access control system is another necessary cost.

Indirect Costs. The primary indirect cost associated with introducing logical access controls into a computer system is the effect on user productivity. There may be additional overhead involved in having individual users properly determine (when under their control) the protection attributes of information. Another indirect cost that may arise results from users not being able to immediately access information necessary to accomplish their jobs because the permissions were incorrectly assigned (or have changed). This situation is familiar to most organizations that put strong emphasis on logical access controls.

References

Abrams, M.D., et al. *A Generalized Framework for Access Control: An Informal Description*. McLean, VA: Mitre Corporation, 1990.

Baldwin, R.W. "Naming and Grouping Privileges to Simplify Security Management in Large Databases." *1990 IEEE Symposium on Security and Privacy Proceedings*. Oakland, CA: IEEE Computer Society Press, May 1990. pp. 116-132.

Caelli, William, Dennis Longley, and Michael Shain. *Information Security Handbook*. New York, NY: Stockton Press, 1991.

Cheswick, William, and Steven Bellovin. *Firewalls and Internet Security*. Reading, MA: Addison-Wesley Publishing Company, 1994.

Curry, D. *Improving the Security of Your UNIX System, ITSTD-721-FR-90-21*. Menlo Park, CA: SRI International, 1990.

Dinkel, Charles. *Secure Data Network System Access Control Documents*. NISTIR 90-4259. Gaithersburg, MD: National Institute of Standards and Technology, 1990.

Fites, P., and M. Kratz. *Information Systems Security: A Practitioner's Reference*. New York, NY: Van Nostrand Reinhold, 1993. Especially Chapters 1, 9, and 12.

Garfinkel, S., and Spafford, G. "UNIX Security Checklist." *Practical UNIX Security*. Sebastopol, CA: O'Riley & Associates. Inc., 1991. pp. 401-413.

Gasser, Morrie. *Building a Secure Computer System*. New York, NY: Van Nostrand Reinhold, 1988.

Haykin, M., and R. Warner. *Smart Card Technology: New Methods for Computer Access Control*. Spec Pub 500-157. Gaithersburg, MD: National Institute of Standards and Technology, 1988.

Landwehr, C., C. Heitmeyer, and J. McLean. "A Security Model for Military Message Systems." *ACM Transactions on Computer Systems*, Vol. 2, No. 3, August 1984.

National Bureau of Standards. *Guidelines for Security of Computer Applications*. Federal Information Processing Standard Publication 73. June 1980.

Pfleeger, Charles. *Security in Computing*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1989.

President's Council on Integrity and Efficiency. *Review of General Controls in Federal Computer Systems.* Washington, DC: President's Council on Integrity and Efficiency, October 1988.

S. Salamone, "Internetwork Security: Unsafe at Any Node?" *Data Communications.* 22(12), 1993. pp. 61-68.

Sandhu, R. "Transaction Control Expressions for Separation of Duty." *Fourth Annual Computer Security Applications Conference Proceedings.* Orlando, FL, December 1988, pp. 282-286.

Thomsen, D.J. "Role-based Application Design and Enforcement." *Fourth IFIP Workshop on Database Security Proceedings.* International Federation for Information Processing, Halifax, England, September 1990.

T. Whiting. "Understanding VAX/VMS Security." *Computers and Security.* 11(8), 1992. pp. 695-698.

Chapter 18

AUDIT TRAILS

Audit trails maintain a record of system activity both by system and application processes and by user activity of systems and applications.[127] In conjunction with appropriate tools and procedures, audit trails can assist in detecting security violations, performance problems, and flaws in applications.[128]

Audit trails may be used as either a support for regular system operations or a kind of insurance policy or as both of these. As insurance, audit trails are maintained but are not used unless needed, such as after a system outage. As a support for operations, audit trails are used to help system administrators ensure that the system or resources have not been harmed by hackers, insiders, or technical problems.

The Difference Between Audit Trails and Auditing

An *audit trail* is a series of records of computer events, about an operating system, an application, or user activities. A computer system may have several audit trails, each devoted to a particular type of activity.

Auditing is the review and analysis of management, operational, and technical controls. The auditor can obtain valuable information about activity on a computer system from the audit trail. Audit trails improve the *auditability* of the computer system. Auditing is discussed in the assurance chapter.

This chapter focuses on audit trails as a technical control, rather than the process of security auditing, which is a review and analysis of the security of a system as discussed in Chapter 9. This chapter discusses the benefits and objectives of audit trails, the types of audit trails, and some common implementation issues.

18.1 Benefits and Objectives

An *event* is any action that happens on a computer system. Examples include logging into a system, executing a program, and opening a file.

Audit trails can provide a means to help accomplish *several* security-related objectives, including individual accountability, reconstruction of events,

---

[127] Some security experts make a distinction between an *audit trail* and an *audit log* as follows: a *log* is a record of events made by a particular software package, and an *audit trail* is an entire history of an event, possibly using several logs. However, common usage within the security community does not make use of this definition. Therefore, this document does not distinguish between trails and logs.

[128] The type and amount of detail recorded by audit trails vary by both the technical capability of the logging application and the managerial decisions. Therefore, when we state that "audit trails can...," the reader should be aware that capabilities vary widely.

intrusion detection, and problem analysis.

18.1.1 Individual Accountability

Audit trails are a technical mechanism that help managers maintain individual accountability. By advising users that they are personally accountable for their actions, which are tracked by an audit trail that logs user activities, managers can help promote proper user behavior.[129] Users are less likely to attempt to circumvent security policy if they know that their actions will be recorded in an audit log.

For example, audit trails can be used in concert with access controls to identify and provide information about users suspected of improper modification of data (e.g., introducing errors into a database). An audit trail may record "before" and "after" versions of records. (Depending upon the size of the file and the capabilities of the audit logging tools, this may be very resource-intensive.) Comparisons can then be made between the actual changes made to records and what was expected. This can help management determine if errors were made by the user, by the system or application software, or by some other source.

Audit trails work in concert with logical access controls, which restrict use of system resources. Granting users access to particular resources usually means that they need that access to accomplish their job. Authorized access, of course, can be misused, which is where audit trail analysis is useful. While users cannot be prevented from using resources to which they have legitimate access authorization, audit trail analysis is used to examine their actions. For example, consider a personnel office in which users have access to those personnel records for which they are responsible. Audit trails can reveal that an individual is printing far more records than the average user, which could indicate the selling of personal data. Another example may be an engineer who is using a computer for the design of a new product. Audit trail analysis could reveal that an outgoing modem was used extensively by the engineer the week before quitting. This could be used to investigate whether proprietary data files were sent to an unauthorized party.

18.1.2 Reconstruction of Events

Audit trails can also be used to reconstruct events after a problem has occurred. Damage can be more easily assessed by reviewing audit trails of system activity to pinpoint how, when, and why normal operations ceased. Audit trail analysis can often distinguish between operator-induced errors (during which the system may have performed exactly as instructed) or system-created errors (e.g., arising from a poorly tested piece of replacement code). If, for example, a system fails or the integrity of a file (either program or data) is questioned, an analysis of the audit trail can reconstruct the series of steps taken by the system, the users, and the application.

---

[129] For a fuller discussion of changing employee behavior, see Chapter 13.

Knowledge of the conditions that existed at the time of, for example, a system crash, can be useful in avoiding future outages. Additionally, if a technical problem occurs (e.g., the corruption of a data file) audit trails can aid in the recovery process (e.g., by using the record of changes made to reconstruct the file).

18.1.3 Intrusion Detection

Intrusion detection refers to the process of identifying attempts to penetrate a system and gain unauthorized access.

If audit trails have been designed and implemented to record appropriate information, they can assist in intrusion detection. Although normally thought of as a real-time effort, intrusions can be detected *in real time*, by examining audit records as they are created (or through the use of other kinds of warning flags/notices), or *after the fact* (e.g., by examining audit records in a batch process).

Real-time intrusion detection is primarily aimed at outsiders attempting to gain unauthorized access to the system. It may also be used to detect changes in the system's performance indicative of, for example, a virus or worm attack.[130] There may be difficulties in implementing real-time auditing, including unacceptable system performance.

After-the-fact identification may indicate that unauthorized access was attempted (or was successful). Attention can then be given to damage assessment or reviewing controls that were attacked.

18.1.4 Problem Analysis

Audit trails may also be used as on-line tools to help identify problems other than intrusions as they occur. This is often referred to as *real-time auditing* or monitoring. If a system or application is deemed to be critical to an organization's business or mission, real-time auditing may be implemented to monitor the status of these processes (although, as noted above, there can be difficulties with real-time analysis). An analysis of the audit trails may be able to verify that the *system* operated normally (i.e., that an error may have resulted from operator error, as opposed to a system-originated error). Such use of audit trails may be complemented by system performance logs. For example, a significant increase in the use of system resources (e.g., disk file space or outgoing modem use) *could* indicate a security problem.

18.2 Audit Trails and Logs

A system can maintain several different audit trails concurrently. There are typically two kinds

---

[130] Viruses and worms of forms of malicious code. A virus is a code segment that replicates by attaching copies of itself to existing executables. A worm is a self-replicating program.

of audit records, (1) an event-oriented log and (2) a record of every keystroke, often called keystroke monitoring. Event-based logs usually contain records describing *system* events, *application* events, or *user* events.

An audit trail should include sufficient information to establish what events occurred and who (or what) caused them. In general, an event record should specify when the event occurred, the user ID associated with the event, the program or command used to initiate the event, and the result. Date and time can help determine if the user was a masquerader or the actual person specified.

18.2.1 Keystroke Monitoring[131]

Keystroke monitoring is the process used to view or record both the keystrokes entered by a computer user and the computer's response during an interactive session. Keystroke monitoring is usually considered a special case of audit trails. Examples of keystroke monitoring would include viewing characters as they are typed by users, reading users' electronic mail, and viewing other recorded information typed by users.

Some forms of routine system maintenance may record user keystrokes. This could constitute keystroke monitoring if the keystrokes are preserved along with the user identification so that an administrator could determine the keystrokes entered by specific users. Keystroke monitoring is conducted in an effort to protect systems and data from intruders who access the systems without authority or in excess of their assigned authority. Monitoring keystrokes typed by intruders can help administrators assess and repair damage caused by intruders.

18.2.2 Audit Events

System audit records are generally used to monitor and fine-tune system performance. *Application audit trails* may be used to discern flaws in applications, or violations of security policy committed within an application. *User audits records* are generally used to hold individuals accountable for their actions. An analysis of user audit records may expose a variety of security violations, which might range from simple browsing to attempts to plant Trojan horses or gain unauthorized privileges.

---

[131] The Department of Justice has advised that an ambiguity in U.S. law makes it unclear whether keystroke monitoring is considered equivalent to an unauthorized telephone wiretap. The ambiguity results from the fact that current laws were written years before such concerns as keystroke monitoring or system intruders became prevalent. Additionally, no legal precedent has been set to determine whether keystroke monitoring is legal or illegal. System administrators conducting such monitoring might be subject to criminal and civil liabilities. The Department of Justice advises system administrators to protect themselves by giving notice to system users if keystroke monitoring is being conducted. Notice should include agency/organization policy statements, training on the subject, and a banner notice on each system being monitored. [NIST, *CSL Bulletin*, March 1993]

Sample System Log File Showing Authentication Messages

```
Jan 27 17:14:04 host1 login: ROOT LOGIN console
Jan 27 17:15:04 host1 shutdown: reboot by root
Jan 27 17:18:38 host1 login: ROOT LOGIN console
Jan 27 17:19:37 host1 reboot: rebooted by root
Jan 28 09:46:53 host1 su: 'su root' succeeded for user1 on /dev/ttyp0
Jan 28 09:47:35 host1 shutdown: reboot by user1
Jan 28 09:53:24 host1 su: 'su root' succeeded for user1 on /dev/ttyp1
Feb 12 08:53:22 host1 su: 'su root' succeeded for user1 on /dev/ttyp1
Feb 17 08:57:50 host1 date: set by user1
Feb 17 13:22:52 host1 su: 'su root' succeeded for user1 on /dev/ttyp0
```

The system itself enforces certain aspects of policy (particularly *system-specific* policy) such as access to files and access to the system itself. Monitoring the alteration of systems configuration files that implement the policy is important. If special accesses (e.g., security administrator access) have to be used to alter configuration files, the system should generate audit records whenever these accesses are used.

Application-Level Audit Record for a Mail Delivery System

```
Apr  9 11:20:22 host1 AA06370: from=<user2@host2>, size=3355, class=0
Apr  9 11:20:23 host1 AA06370: to=<user1@host1>, delay=00:00:02,
stat=Sent
Apr  9 11:59:51 host1 AA06436: from=<user4@host3>, size=1424, class=0
Apr  9 11:59:52 host1 AA06436: to=<user1@host1>, delay=00:00:02,
stat=Sent
Apr  9 12:43:52 host1 AA06441: from=<user2@host2>, size=2077, class=0
Apr  9 12:43:53 host1 AA06441: to=<user1@host1>, delay=00:00:01,
stat=Sent
```

Sometimes a finer level of detail than system audit trails is required. *Application audit trails* can provide this greater level of recorded detail. If an application is critical, it can be desirable to record not only who invoked the application, but certain details specific to each use. For example, consider an e-mail application. It may be desirable to record who sent mail, as well as to whom they sent mail and the length of messages. Another example would be that of a database application. It may be useful to record who accessed what database as well as the individual rows or columns of a table that were read (or changed or deleted), instead of just recording the execution of the database program.

User Log Showing a Chronological List of Commands Executed by Users

```
rcp         user1    ttyp0    0.02 secs Fri Apr  8 16:02
ls          user1    ttyp0    0.14 secs Fri Apr  8 16:01
clear       user1    ttyp0    0.05 secs Fri Apr  8 16:01
rpcinfo     user1    ttyp0    0.20 secs Fri Apr  8 16:01
nroff       user2    ttyp2    0.75 secs Fri Apr  8 16:00
sh          user2    ttyp2    0.02 secs Fri Apr  8 16:00
mv          user2    ttyp2    0.02 secs Fri Apr  8 16:00
sh          user2    ttyp2    0.03 secs Fri Apr  8 16:00
col         user2    ttyp2    0.09 secs Fri Apr  8 16:00
man         user2    ttyp2    0.14 secs Fri Apr  8 15:57
```

A *user audit trail* monitors and logs user activity in a system or application by recording events initiated by the user (e.g., access of a file, record or field, use of a modem).

Flexibility is a critical feature of audit trails. Ideally (from a security point of view), a system administrator would have the ability to monitor all system and user activity, but could choose to log only certain functions at the system level, and within certain applications. The decision of how much to log and how much to review should be a function of application/data sensitivity and should be decided by each functional manager/application owner with guidance from the system administrator and the computer security manager/officer, weighing the costs and benefits of the logging.[132]

18.2.2.1 System-Level Audit Trails

If a system-level audit capability exists, the audit trail should capture, at a minimum, any attempt to log on (successful or unsuccessful), the log-on ID, date and time of each log-on attempt, date and time of each log-off, the devices used, and the function(s) performed once logged on (e.g., the applications that the user tried, successfully or unsuccessfully, to invoke). System-level logging also typically includes information that is not specifically security-related, such as system operations, cost-accounting charges, and network performance.

A system audit trail should be able to identify failed log-on attempts, especially if the system does not limit the number of failed log-on attempts. Unfortunately, some system-level audit trails cannot detect attempted log-ons, and therefore, cannot log them for later review. These audit trails can only monitor and log successful log-ons and subsequent activity. To effectively detect intrusion, a record of failed log-on attempts is required.

---

[132] In general, audit logging can have privacy implications. Users should be aware of applicable privacy laws, regulations, and policies that may apply in such situations.

18.2.2.2 Application-Level Audit Trails

System-level audit trails may not be able to track and log events *within* applications, or may not be able to provide the level of detail needed by application or data owners, the system administrator, or the computer security manager. In general, application-level audit trails monitor and log user activities, including data files opened and closed, specific actions, such as reading, editing, and deleting records or fields, and printing reports. Some applications may be sensitive enough from a data availability, confidentiality, and/or integrity perspective that a "before" and "after" picture of each modified record (or the data element(s) changed within a record) should be captured by the audit trail.

18.2.2.3 User Audit Trails

User audit trails can usually log:

- all commands directly initiated by the user;
- all identification and authentication attempts; and
- files and resources accessed.

It is most useful if options and parameters are also recorded from commands. It is much more useful to know that a user tried to delete a log file (e.g., to hide unauthorized actions) than to know the user merely issued the delete command, possibly for a personal data file.

18.3 Implementation Issues

Audit trail data requires protection, since the data should be available for use when needed and is not useful if it is not accurate. *Also, the best planned and implemented audit trail is of limited value without timely review of the logged data.* Audit trails may be reviewed periodically, as needed (often triggered by occurrence of a security event), automatically in realtime, or in some combination of these. System managers and administrators, with guidance from computer

Audit Logs for Physical Access

Physical access control systems (e.g., a card/key entry system or an alarm system) use software and audit trails similar to general-purpose computers. The following are *examples* of criteria that may be used in selecting which events to log:

The date and time the access was attempted or made should be logged, as should the gate or door through which the access was attempted or made, and the individual (or user ID) making the attempt to access the gate or door.

Invalid attempts should be monitored and logged by noncomputer audit trails just as they are for computer-system audit trails. Management should be made aware if someone attempts to gain access during unauthorized hours.

Logged information should also include attempts to add, modify, or delete physical access privileges (e.g., granting a new employee access to the building or granting transferred employees access to their new office [and, of course, deleting their old access, as applicable]).

As with system and application audit trails, auditing of noncomputer functions can be implemented to send messages to security personnel indicating valid or invalid attempts to gain access to controlled spaces. In order not to desensitize a guard or monitor, all access should not result in messages being sent to a screen. Only exceptions, such as failed access attempts, should be highlighted to those monitoring access.

security personnel, should determine how long audit trail data will be maintained – either on the system or in archive files.

Following are examples of implementation issues that may have to be addressed when using audit trails.

18.3.1 Protecting Audit Trail Data

Access to on-line audit logs should be strictly controlled. Computer security managers and system administrators or managers should have access for review purposes; however, security and/or administration personnel who maintain logical access functions may have no need for access to audit logs.

It is particularly important to ensure the *integrity* of audit trail data against modification. One way to do this is to use digital signatures. (See Chapter 19.) Another way is to use write-once devices. The audit trail files needs to be protected since, for example, intruders may try to "cover their tracks" by modifying audit trail records. Audit trail records should be protected by strong access controls to help prevent unauthorized access. The integrity of audit trail information may be particularly important when legal issues arise, such as when audit trails are used as legal evidence. (This may, for example, require daily printing and signing of the logs.) Questions of such legal issues should be directed to the cognizant legal counsel.

The confidentiality of audit trail information may also be protected, for example, if the audit trail is recording information about users that may be disclosure-sensitive such as transaction data containing personal information (e.g., "before" and "after" records of modification to income tax data). Strong access controls and encryption can be particularly effective in preserving confidentiality.

18.3.2 Review of Audit Trails

Audit trails can be used to review what occurred after an event, for periodic reviews, and for real-time analysis. Reviewers should know what to look for to be effective in spotting unusual activity. They need to understand what normal activity looks like. Audit trail review can be easier if the audit trail function can be queried by user ID, terminal ID, application name, date and time, or some other set of parameters to run reports of selected information.

Audit Trail Review After an Event. Following a known system or application software problem, a known violation of existing requirements by a user, or some unexplained system or user problem, the appropriate system-level or application-level administrator should review the audit trails. Review by the application/data owner would normally involve a separate report, based upon audit trail data, to determine if their resources are being misused.

Periodic Review of Audit Trail Data. Application owners, data owners, system administrators, data processing function managers, and computer security managers should determine how much review of audit trail records is necessary, based on the importance of identifying unauthorized activities. This determination should have a direct correlation to the frequency of periodic reviews of audit trail data.

Real-Time Audit Analysis. Traditionally, audit trails are analyzed in a batch mode at regular intervals (e.g., daily). Audit records are archived during that interval for later analysis. Audit analysis tools can also be used in a real-time, or near real-time fashion. Such intrusion detection tools are based on audit reduction, attack signature, and variance techniques. Manual review of audit records in real time is almost never feasible on large multiuser systems due to the volume of records generated. However, it might be possible to view all records associated with a particular user or application, and view them in real time.[133]

18.3.3 Tools for Audit Trail Analysis

Many types of tools have been developed to help to reduce the amount of information contained in audit records, as well as to distill useful information from the raw data. Especially on larger systems, audit trail software can create very large files, which can be extremely difficult to analyze manually. The use of automated tools is likely to be the difference between unused audit trail data and a robust program. Some of the types of tools include:

Audit reduction tools are preprocessors designed to reduce the volume of audit records to facilitate manual review. Before a security review, these tools can remove many audit records known to have little security significance. (This alone may cut in half the number of records in the audit trail.) These tools generally remove records generated by specified classes of events, such as records generated by nightly backups might be removed.

Trends/variance-detection tools look for anomalies in user or system behavior. It is possible to construct more sophisticated processors that monitor usage trends and detect major variations. For example, if a user typically logs in at 9 a.m., but appears at 4:30 a.m. one morning, this may indicate a security problem that may need to be investigated.

Attack signature-detection tools look for an *attack signature*, which is a specific sequence of events indicative of an unauthorized access attempt. A simple example would be repeated failed log-in attempts.

18.4 Interdependencies

The ability to audit supports many of the controls presented in this handbook. The following

---

[133] This is similar to keystroke monitoring, though, and may be legally restricted.

paragraphs describe some of the most important interdependencies.

Policy. The most fundamental interdependency of audit trails is with policy. Policy dictates who is authorized access to what system resources. Therefore it specifies, directly or indirectly, what violations of policy should be identified through audit trails.

Assurance. System auditing is an important aspect of operational assurance. The data recorded into an audit trail is used to support a system audit. The analysis of audit trail data and the process of auditing systems are closely linked; in some cases, they may even be the same thing. In most cases, the analysis of audit trail data is a critical part of maintaining operational assurance.

Identification and Authentication. Audit trails are tools often used to help hold users accountable for their actions. To be held accountable, the users must be known to the system (usually accomplished through the identification and authentication process). However, as mentioned earlier, audit trails record events and associate them with the *perceived* user (i.e., the user ID). If a user is impersonated, the audit trail will establish events but *not* the identity of the user.

Logical Access Control. Logical access controls restrict the use of system resources to authorized users. Audit trails complement this activity in two ways. First, they may be used to identify breakdowns in logical access controls or to verify that access control restrictions are behaving as expected, for example, if a particular user is erroneously included in a group permitted access to a file. Second, audit trails are used *to audit use of resources by those who have legitimate access*. Additionally, to protect audit trail files, access controls are used to ensure that audit trails are not modified.

Contingency Planning. Audit trails assist in contingency planning by leaving a record of activities performed on the system or within a specific application. In the event of a technical malfunction, this log can be used to help reconstruct the state of the system (or specific files).

Incident Response. If a security incident occurs, such as hacking, audit records and other intrusion detection methods can be used to help determine the extent of the incident. For example, was just one file browsed, or was a Trojan horse planted to collect passwords?

Cryptography. Digital signatures can be used to protect audit trails from undetected modification. (This does not prevent deletion or modification of the audit trail, but will provide an alert that the audit trail has been altered.) Digital signatures can also be used in conjunction with adding secure time stamps to audit records. Encryption can be used if confidentiality of audit trail information is important.

18.5 Cost Considerations

Audit trails involve many costs. First, some system overhead is incurred recording the audit trail. Additional system overhead will be incurred storing and processing the records. The more detailed the records, the more overhead is required. Another cost involves human and machine time required to do the analysis. This can be minimized by using tools to perform most of the analysis. Many simple analyzers can be constructed quickly (and cheaply) from system utilities, but they are limited to audit reduction and identifying particularly sensitive events. More complex tools that identify trends or sequences of events are slowly becoming available as off-the-shelf software. (If complex tools are not available for a system, development may be prohibitively expensive. Some intrusion detection systems, for example, have taken years to develop.)

The final cost of audit trails is the cost of investigating anomalous events. If the system is identifying too many events as suspicious, administrators may spend undue time reconstructing events and questioning personnel.

References

Fites, P., and M. Kratz. *Information Systems Security: A Practitioner's Reference*. New York: Van Nostrand Reinhold, 1993, (especially Chapter 12, pp. 331 - 350).

Kim, G., and E. Spafford, "Monitoring File System Integrity on UNIX Platforms." *Infosecurity News*. 4(4), 1993. pp. 21-22.

Lunt, T. "Automated Audit Trail Analysis for Intrusion Detection," *Computer Audit Update*, April 1992. pp. 2-8.

National Computer Security Center. *A Guide to Understanding Audit in Trusted Systems*. NCSC-TG-001, Version-2. Ft. Meade, MD, 1988.

National Institute of Standards and Technology. "Guidance on the Legality of Keystroke Monitoring." *CSL Bulletin*. March 1993.

Phillips, P. W. "New Approach Identifies Malicious System Activity." *Signal*. 46(7), 1992. pp. 65-66.

Ruthberg, Z., et al. *Guide to Auditing for Controls and Security: A System Development Life Cycle Approach*. Special Publication 500-153. Gaithersburg, MD: National Bureau of Standards, 1988.

Stoll, Clifford. *The Cuckoo's Egg*. New York, NY: Doubleday, 1989.

Chapter 19

CRYPTOGRAPHY

Cryptography is a branch of mathematics based on the transformation of data. It provides an important tool for protecting information and is used in many aspects of computer security. For example, cryptography can help provide data confidentiality, integrity, electronic signatures, and advanced user authentication. Although modern cryptography relies upon advanced mathematics, users can reap its benefits without understanding its mathematical underpinnings.

This chapter describes cryptography as a tool for satisfying a wide spectrum of computer security needs and requirements. It describes fundamental aspects of the basic cryptographic technologies and some specific ways cryptography can be applied to improve security. The chapter also explores some of the important issues that should be considered when incorporating cryptography into computer systems.

> Cryptography is traditionally associated only with keeping data secret. However, modern cryptography can be used to provide many security services, such as electronic signatures and ensuring that data has not been modified.

19.1 Basic Cryptographic Technologies

Cryptography relies upon two basic components: an *algorithm* (or cryptographic methodology) and a *key*. In modern cryptographic systems, algorithms are complex mathematical formulae and keys are strings of bits. For two parties to communicate, they must use the same algorithm (or algorithms that are designed to work together). In some cases, they must also use the same key. Many cryptographic keys must be kept secret; sometimes algorithms are also kept secret.

There are two basic types of cryptography: *secret key systems* (also called symmetric systems) and *public key systems* (also called asymmetric systems). Table 19.1 compares some of the distinct features of secret and public key systems. Both types of systems offer advantages and disadvantages. Often, the two are combined to form a *hybrid system* to exploit the strengths of each type. To determine which type of cryptography best meets its needs, an organization first has to identify its security requirements and operating environment.

> There are two basic types of cryptography: "secret key" and "public key."

DISTINCT FEATURES	SECRET KEY CRYPTOGRAPHY	PUBLIC KEY CRYPTOGRAPHY
NUMBER OF KEYS	Single key.	Pair of keys.
TYPES OF KEYS	Key is secret.	One key is private, and one key is public.
PROTECTION OF KEYS	Disclosure and modification.	Disclosure and modification for private keys and modification for public keys.
RELATIVE SPEEDS	Faster.	Slower.

Table 19.1

19.1.1 Secret Key Cryptography

In secret key cryptography, two (or more) parties share the same key, and that key is used to encrypt and decrypt data. As the name implies, secret key cryptography relies on keeping the key secret. If the key is compromised, the security offered by cryptography is severely reduced or eliminated. Secret key cryptography assumes that the parties who share a key rely upon each other not to disclose the key and protect it against modification.

Secret key cryptography has been in use for centuries. Early forms merely transposed the written characters to hide the message.

The best known secret key system is the *Data Encryption Standard* (DES), published by NIST as Federal Information Processing Standard (FIPS) 46-2. Although the adequacy of DES has at times been questioned, these claims remain unsubstantiated, and DES remains strong. It is the most widely accepted, publicly available cryptographic system today. The American National Standards Institute (ANSI) has adopted DES as the basis for encryption, integrity, access control, and key management standards.

The *Escrowed Encryption Standard*, published as FIPS 185, also makes use of a secret key system. (See the discussion of Key Escrow Encryption in this chapter.)

19.1.2 Public Key Cryptography

Public key cryptography is a modern invention and requires the use of advanced mathematics.

Whereas secret key cryptography uses a

single key shared by two (or more) parties, public key cryptography uses a pair of keys for *each* party. One of the keys of the pair is "public" and the other is "private." The public key can be made known to other parties; the private key must be kept confidential and must be known only to its owner. Both keys, however, need to be protected against modification.

Public key cryptography is particularly useful when the parties wishing to communicate cannot rely upon each other or do not share a common key. There are several public key cryptographic systems. One of the first public key systems is RSA, which can provide many different security services. The Digital Signature Standard (DSS), described later in the chapter, is another example of a public key system.

19.1.3 Hybrid Cryptographic Systems

Public and secret key cryptography have relative advantages and disadvantages. Although public key cryptography does not require users to share a common key, secret key cryptography is much faster: equivalent implementations of secret key cryptography can run 1,000 to 10,000 times faster than public key cryptography.

Secret key systems are often used for bulk data encryption and public key systems for automated key distribution.

To maximize the advantages and minimize the disadvantages of both secret and public key cryptography, a computer system can use both types in a complementary manner, with each performing different functions. Typically, the speed advantage of secret key cryptography means that it is used for encrypting data. Public key cryptography is used for applications that are less demanding to a computer system's resources, such as encrypting the keys used by secret key cryptography (for distribution) or to sign messages.

19.1.4 Key Escrow

Because cryptography can provide extremely strong encryption, it can thwart the government's efforts to lawfully perform electronic surveillance. For example, if strong cryptography is used to encrypt a phone conversation, a court-authorized wiretap will not be effective. To meet the needs of the government *and* to provide privacy, the federal government has adopted voluntary key escrow cryptography. This technology allows the use of strong encryption, but also allows the government when legally authorized to obtain decryption keys held by escrow agents. NIST has published the *Escrowed Encryption Standard* as FIPS 185. Under the Federal Government's voluntary key escrow initiative, the decryption keys are split into parts and given to separate escrow authorities. Access to one part of the key does *not* help decrypt the data; both keys must be obtained.

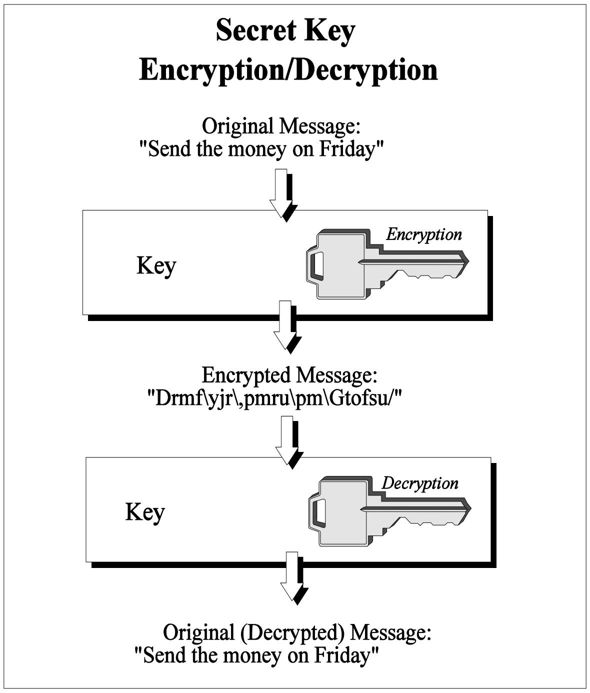
19.2 Uses of Cryptography

Cryptography is used to protect data *both* inside and outside the boundaries of a computer system. Outside the computer system, cryptography is sometimes the *only* way to protect data. While in a computer system, data is normally protected with logical and physical access controls (perhaps supplemented by cryptography). However, when in transit across communications lines or resident on someone else's computer, data cannot be protected by the originator's[134] logical or physical access controls. Cryptography provides a solution by protecting data even when the data is no longer in the control of the originator.

19.2.1 Data Encryption

One of the best ways to obtain cost-effective data confidentiality is through the use of encryption. Encryption transforms intelligible data, called *plaintext,*[135] into an unintelligible form, called *ciphertext*. This process is reversed through the process of decryption. Once data is encrypted, the ciphertext does not have to be protected against disclosure. However, if ciphertext is modified, it will not decrypt correctly.

Both secret key and public key cryptography can be used for data encryption although not all public key algorithms provide for data encryption.



Secret Key Encryption/Decryption

To use a secret key algorithm, data is encrypted using a key. The same key must be used to decrypt the data.

When public key cryptography is used for encryption, any party may use any other party's public key to encrypt a message; however, only the party with the corresponding private key can

---

[134] The originator does not have to be the original creator of the data. It can also be a guardian or custodian of the data.

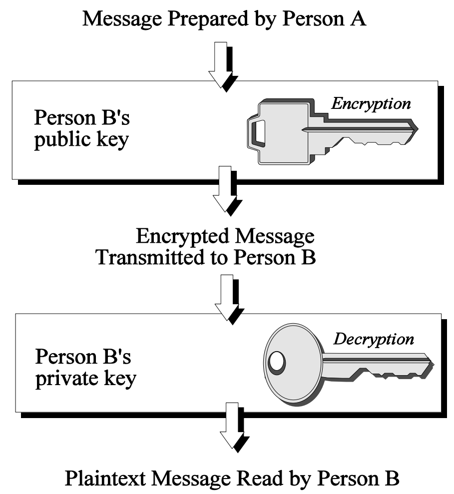
[135] Plaintext can be intelligible to a human (e.g., a novel) or to a machine (e.g., executable code).

decrypt, and thus read, the message.

Since secret key encryption is typically much faster, it is normally used for encrypting larger amounts of data.

19.2.2 Integrity

Use of Public Key Cryptography for Encryption/Decryption



Person A knows that only Person B can read the message.

In computer systems, it is not always possible for humans to scan information to determine if data has been erased, added, or modified. Even if scanning were possible, the individual may have no way of knowing what the correct data should be. For example, "do" may be changed to "do not," or $1,000 may be changed to $10,000. It is therefore desirable to have an automated means of detecting *both* intentional and unintentional modifications of data.

While error detecting codes have long been used in communications protocols (e.g., parity bits), these are more effective in detecting (and correcting) unintentional modifications. They can be defeated by adversaries. Cryptography can effectively detect both intentional and unintentional modification; however, cryptography does not protect files from being modified. Both secret key and public key cryptography can be used to ensure integrity. Although newer public key methods may offer more flexibility than the older secret key method, secret key integrity verification systems have been successfully integrated into many applications.

When secret key cryptography is used, a message authentication code (MAC) is calculated from and appended to the data. To verify that the data has not been modified at a later time, any party with access to the correct secret key can recalculate the MAC. The new MAC is compared with the original MAC, and if they are identical, the verifier has confidence that the data has not been modified by an unauthorized party. FIPS 113, *Computer Data Authentication*, specifies a standard technique for calculating a MAC for integrity verification.

Public key cryptography verifies integrity by using of public key signatures and secure hashes. A secure hash algorithm is used to create a message digest. The message digest, called a hash, is a short form of the message that changes if the message is modified. The hash is then signed with a private key. Anyone can recalculate the hash and use the corresponding public key to verify the integrity of the message.[136]

19.2.3 Electronic Signatures

What Is an Electronic Signature?

An electronic signature is a cryptographic mechanism that performs a similar function to a written signature. It is used to verify the origin and contents of a message. For example, a recipient of data (e.g., an e-mail message) can verify who signed the data and that the data was not modified after being signed. This also means that the originator (e.g., sender of an e-mail message) cannot falsely deny having signed the data.

Today's computer systems store and process increasing numbers of paper-based documents in electronic form. Having documents in electronic form permits rapid processing and transmission and improves overall efficiency. However, approval of a paper document has traditionally been indicated by a written signature. What is needed, therefore, is the electronic equivalent of a written signature that can be recognized as having the same legal status as a written signature. In addition to the integrity protections, discussed above, cryptography can provide a means of linking a document with a particular person, as is done with a written signature. Electronic signatures can use either secret key or public key cryptography; however, public key methods are generally easier to use.

Cryptographic signatures provide extremely strong proof that a message has not been altered and was signed by a specific key.[137] However, there are other mechanisms besides cryptographic-based electronic signatures that perform a *similar* function. These mechanisms provide some assurance of the origin of a message, some verification of the message's integrity, or both.[138]

- Examination of the transmission path of a message. When messages are sent across a network, such as the Internet, the message source and the physical path of the message are recorded as a part of the message. These can be examined electronically or manually

---

[136] Sometimes a secure hash is used for integrity verification. However, this can be defeated if the hash is not stored in a secure location, since it may be possible for someone to change the message and then replace the old hash with a new one based on the modified message.

[137] Electronic signatures rely on the secrecy of the keys and the link or binding between the owner of the key and the key itself. If a key is compromised (by theft, coercion, or trickery), then the electronic originator of a message may not be the same as the owner of the key. Although the binding of cryptographic keys to actual people is a significant problem, it does not necessarily make electronic signatures less secure than written signatures. Trickery and coercion are problems for written signatures as well. In addition, written signatures are easily forged.

[138] The strength of these mechanisms relative to electronic signatures varies depending on the specific implementation; however, in general, electronic signatures are stronger and more flexible. These mechanisms may be used in conjunction with electronic signatures or separately, depending upon the system's specific needs and limitations.

to help ascertain the origin of a message.

- Use of a value-added network provider. If two or more parties are communicating via a third party network, the network provider may be able to provide assurance that messages originate from a given source and have not been modified.
- Acknowledgment statements. The recipient of an electronic message may confirm the message's origin and contents by sending back an acknowledgement statement.
- Use of audit trails. Audit trails can track the sending of messages and their contents for later reference.

Simply taking a digital picture of a written signature does not provide adequate security. Such a *digitized* written signature could easily be copied from one electronic document to another with no way to determine whether it is legitimate. Electronic signatures, on the other hand, are unique to the message being signed and will not verify if they are copied to another document.

19.2.3.1 Secret Key Electronic Signatures

> Systems incorporating message authentication technology have been approved for use by the federal government as a replacement for written signatures on electronic documents.

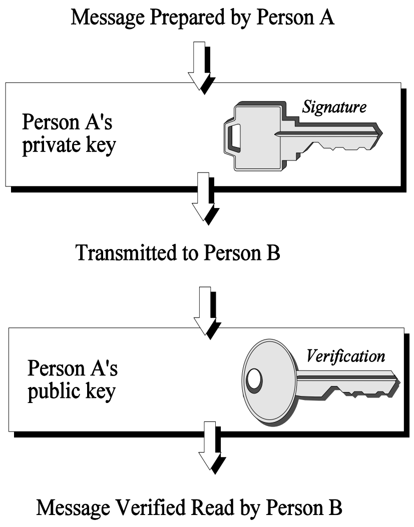
An electronic signature can be implemented using secret key message authentication codes (MACs). For example, if two parties share a secret key, and one party receives data with a MAC that is correctly verified using the shared key, that party may assume that the other party signed the data. This assumes, however, that the two parties trust each other. Thus, through the use of a MAC, in addition to data integrity, a form of electronic signature is obtained. Using additional controls, such as key notarization and key attributes, it is possible to provide an electronic signature even if the two parties do not trust each other.

19.2.3.2 Public Key Electronic Signatures

Another type of electronic signature called a *digital signature* is implemented using public key cryptography. Data is electronically signed by applying the originator's private key to the data. (The exact mathematical process for doing this is not important for this discussion.) To increase the speed of the process, the private key is applied to a shorter form of the data, called a "hash" or "message digest," rather than to the entire set of data. The resulting digital signature can be stored or transmitted along with the data. The signature can be verified by any party using the public key of the signer. This feature is very useful, for example, when distributing signed copies of virus-free software. Any recipient can verify that the program remains virus-free. If the signature verifies properly, then the verifier has confidence that the data was not modified after being signed and that the owner of the public key was the signer.

NIST has published standards for a digital signature and a secure hash for use by the federal government in FIPS 186, *Digital Signature Standard* and FIPS 180, *Secure Hash Standard.*

Use of Public Key Cryptography for Digital Signature



Person B knows that only Person A could have sent the message.

19.2.4 User Authentication

Cryptography can increase security in user authentication techniques. As discussed in Chapter 16, cryptography is the basis for several advanced authentication methods. Instead of communicating passwords over an open network, authentication can be performed by demonstrating knowledge of a cryptographic key. Using these methods, a one-time password, which is not susceptible to eavesdropping, can be used. User authentication can use either secret or public key cryptography.

19.3 Implementation Issues

This section explores several important issues that should be considered when using (e.g., designing, implementing, integrating) cryptography in a computer system.

19.3.1 Selecting Design and Implementation Standards

NIST and other organizations have developed numerous standards for designing, implementing, and using cryptography and for integrating it into automated systems. By using these standards, organizations can reduce costs and protect their investments in technology. Standards provide solutions that have been accepted by a wide community and that have been reviewed by experts in relevant areas. Standards help ensure interoperability among different vendors' equipment, thus allowing an organization to select from among various products in order to find cost-effective equipment.

Applicable security standards provide a common level of security and interoperability among users.

Managers and users of computer systems will have to select among various standards when deciding to use cryptography. Their selection should be based on cost-effectiveness analysis, trends in the standard's acceptance, and interoperability requirements. In addition, each standard should be carefully analyzed to determine if it is applicable to the organization and the desired application. For example, the Data Encryption Standard and the Escrowed Encryption Standard are both applicable to certain applications involving communications of data over commercial modems. Some federal standards are mandatory for federal computer systems, including DES (FIPS 46-2) and the DSS (FIPS 181).

19.3.2 Deciding on Hardware vs. Software Implementations

The trade-offs among security, cost, simplicity, efficiency, and ease of implementation need to be studied by managers acquiring various security products meeting a standard. Cryptography can be implemented in either hardware or software. Each has its related costs and benefits.

In general, software is less expensive and slower than hardware, although for large applications, hardware may be less expensive. In addition, software may be less secure, since it is more easily modified or bypassed than equivalent hardware products. Tamper resistance is usually considered better in hardware.

In many cases, cryptography is implemented in a hardware device (e.g., electronic chip, ROM-protected processor) but is controlled by software. This software requires integrity protection to ensure that the hardware device is provided with correct information (i.e., controls, data) and is not bypassed. Thus, a hybrid solution is generally provided, even when the basic cryptography is implemented in hardware. Effective security requires the correct management of the entire hybrid solution.

19.3.3 Managing Keys

The proper management of cryptographic keys is essential to the effective use of cryptography for security. Ultimately, the security of information protected by cryptography directly depends upon the protection afforded to keys.

All keys need to be protected against modification, and secret keys and private keys need protection against unauthorized disclosure. Key management involves the procedures and protocols, both manual and automated, used throughout the entire life cycle of the keys. This includes the generation, distribution, storage, entry, use, destruction, and archiving of cryptographic keys.

With secret key cryptography, the secret key(s) should be securely distributed (i.e., safeguarded against unauthorized replacement, modification, and disclosure) to the parties wishing to communicate. Depending upon the number and location of users, this task may not be trivial. Automated techniques for generating and distributing cryptographic keys can ease overhead costs of key management, but some resources have to be devoted to this task. FIPS 171, *Key Management Using ANSI X9.17*, provides key management solutions for a variety of operational environments.

Public key cryptography users also have to satisfy certain key management requirements. For example, since a private-public key pair is associated with (i.e., generated or held by) a specific user, it is necessary to *bind* the public part of the key pair to the user.[139]

In a small community of users, public keys and their "owners" can be strongly bound by simply exchanging public keys (e.g., putting them on a CD-ROM or other media). However, conducting electronic business on a larger scale, potentially involving geographically and organizationally distributed users, necessitates a means for obtaining public keys electronically with a high degree of confidence in their integrity and binding to individuals. The support for the binding between a key and its owner is generally referred to as a *public key infrastructure*.

Users also need to be able enter the community of key holders, generate keys (or have them generated on their behalf), disseminate public keys, revoke keys (in case, for example, of compromise of the private key), and change keys. In addition, it may be necessary to build in time/date stamping and to archive keys for verification of old signatures.

19.3.4 Security of Cryptographic Modules

Cryptography is typically implemented in a *module* of software, firmware, hardware, or some combination thereof. This module contains the cryptographic algorithm(s), certain control parameters, and temporary storage facilities for the key(s) being used by the algorithm(s). The proper functioning of the cryptography requires the secure design, implementation, and use of the cryptographic module. This includes protecting the module against tampering.

FIPS 140-1, *Security Requirements for Cryptographic Modules*, specifies the physical and logical security requirements for cryptographic modules. The standard defines four security levels for cryptographic modules, with each level providing a significant increase in security over the preceding level. The four levels allow for cost-effective solutions that are appropriate for different degrees of data sensitivity and different application environments. The user can select the best module for any given application or system, avoiding the cost of unnecessary security features.

19.3.5 Applying Cryptography to Networks

The use of cryptography within networking applications often requires special considerations. In these applications, the suitability of a cryptographic module may depend on its capability for handling special requirements imposed by locally attached communications equipment or by the network protocols and software.

---

[139] In some cases, the key may be bound to a position or an organization, rather than to an individual user.

Encrypted information, MACs, or digital signatures may require transparent communications protocols or equipment to avoid being misinterpreted by the communications equipment or software as control information. It may be necessary to format the encrypted information, MAC, or digital signature to ensure that it does not confuse the communications equipment or software. It is essential that cryptography satisfy the requirements imposed by the communications equipment and does not interfere with the proper and efficient operation of the network.

Data is encrypted on a network using either link or end-to-end encryption. In general, *link encryption* is performed by service providers, such as a data communications provider. Link encryption encrypts all of the data along a communications path (e.g., a satellite link, telephone circuit, or T1 line). Since link encryption also encrypts routing data, communications nodes need to decrypt the data to continue routing. *End-to-end encryption* is generally performed by the end-user organization. Although data remains encrypted when being passed through a network, routing information remains visible. It is possible to combine both types of encryption.

19.3.6 Complying with Export Rules

The U.S. Government controls the export of cryptographic implementations. The rules governing export can be quite complex, since they consider multiple factors. In addition, cryptography is a rapidly changing field, and rules may change from time to time. Questions concerning the export of a particular implementation should be addressed to appropriate legal counsel.

19.4 Interdependencies

There are many interdependencies among cryptography and other security controls highlighted in this handbook. Cryptography both depends on other security safeguards and assists in providing them.

Physical Security. Physical protection of a cryptographic module is required to prevent – or at least detect – physical replacement or modification of the cryptographic system and the keys within it. In many environments (e.g., open offices, portable computers), the cryptographic module itself has to provide the desired levels of physical security. In other environments (e.g., closed communications facilities, steel-encased Cash-Issuing Terminals), a cryptographic module may be safely employed within a secured facility.

User Authentication. Cryptography can be used both to protect passwords that are stored in computer systems and to protect passwords that are communicated between computers. Furthermore, cryptographic-based authentication techniques may be used in conjunction with, or in place of, password-based techniques to provide stronger authentication of users.

Logical Access Control. In many cases, cryptographic software may be embedded within a host system, and it may not be feasible to provide extensive physical protection to the host system. In these cases, logical access control may provide a means of isolating the cryptographic software from other parts of the host system and for protecting the cryptographic software from tampering and the keys from replacement or disclosure. The use of such controls should provide the equivalent of physical protection.

Audit Trails. Cryptography may play a useful role in audit trails. For example, audit records may need to be signed. Cryptography may also be needed to protect audit records stored on computer systems from disclosure or modification. Audit trails are also used to help support electronic signatures.

Assurance. Assurance that a cryptographic module is properly and securely implemented is essential to the effective use of cryptography. NIST maintains validation programs for several of its standards for cryptography. Vendors can have their products validated for conformance to the standard through a rigorous set of tests. Such testing provides increased assurance that a module meets stated standards, and system designers, integrators, and users can have greater confidence that validated products conform to accepted standards.

NIST maintains validation programs for several of its cryptographic standards.

A cryptographic system should be monitored and periodically audited to ensure that it is satisfying its security objectives. All parameters associated with correct operation of the cryptographic system should be reviewed, and operation of the system itself should be periodically tested and the results audited. Certain information, such as secret keys or private keys in public key systems, should not be subject to audit. However, nonsecret or nonprivate keys could be used in a simulated audit procedure.

19.5 Cost Considerations

Using cryptography to protect information has both direct and indirect costs. Cost is determined in part by product availability; a wide variety of products exist for implementing cryptography in integrated circuits, add-on boards or adapters, and stand-alone units.

19.5.1 Direct Costs

The direct costs of cryptography include:

- Acquiring or implementing the cryptographic module and integrating it into the computer system. The medium (i.e., hardware, software, firmware, or combination) and various other issues such as level of security, logical and physical configuration, and special processing requirements will have an impact on cost.

- Managing the cryptography and, in particular, managing the cryptographic keys, which includes key generation, distribution, archiving, and disposition, as well as security measures to protect the keys, as appropriate.

19.5.2 Indirect Costs

The indirect costs of cryptography include:

- A decrease in system or network performance, resulting from the additional overhead of applying cryptographic protection to stored or communicated data.

- Changes in the way users interact with the system, resulting from more stringent security enforcement. However, cryptography can be made nearly transparent to the users so that the impact is minimal.

References

Alexander, M., ed. "Protecting Data With Secret Codes," *Infosecurity News*. 4(6), 1993. pp. 72-78.

American Bankers Association. *American National Standard for Financial Institution Key Management (Wholesale)*. ANSI X9.17-1985. Washington, DC., 1985.

Denning, P., and D. Denning, "The Clipper and Capstone Encryption Systems." *American Scientist*. 81(4), 1993. pp. 319-323.

Diffie, W., and M. Hellman. "New Directions in Cryptography." *IEEE Transactions on Information Theory*. Vol. IT-22, No. 6, November 1976. pp. 644-654.

Duncan, R. "Encryption ABCs." *Infosecurity News*. 5(2), 1994. pp. 36-41.

International Organization for Standardization. *Information Processing Systems - Open Systems Interconnection Reference Model - Part 2: Security Architecture*. ISO 7498/2. 1988.

Meyer, C.H., and S. M. Matyas. *Cryptography: A New Dimension in Computer Data Security*. New York, NY: John Wiley & Sons, 1982.

Nechvatal, James. *Public-Key Cryptography*. Special Publication 800-2. Gaithersburg, MD: National Institute of Standards and Technology, April 1991.

National Bureau of Standards. *Computer Data Authentication*. Federal Information Processing Standard Publication 113. May 30, 1985.

National Institute of Standards and Technology. "Advanced Authentication Technology." *Computer Systems Laboratory Bulletin*. November 1991.

National Institute of Standards and Technology. *Data Encryption Standard*. Federal Information Processing Standard Publication 46-2. December 30, 1993.

National Institute of Standards and Technology. "Digital Signature Standard." *Computer Systems Laboratory Bulletin*. January 1993.

National Institute of Standards and Technology. *Digital Signature Standard*. Federal Information Processing Standard Publication 186. May 1994.

National Institute of Standards and Technology. *Escrowed Encryption Standard*. Federal Information Processing Standard Publication 185. 1994.

National Institute of Standards and Technology. *Key Management Using ANSI X9.17*. Federal Information Processing Standard Publication 171. April 27, 1992.

National Institute of Standards and Technology. *Secure Hash Standard*. Federal Information Processing Standard Publication 180. May 11, 1993.

National Institute of Standards and Technology. *Security Requirements for Cryptographic Modules*. Federal Information Processing Standard Publication 140-1. January 11, 1994.

Rivest, R., A. Shamir, and L. Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems." *Communications of the ACM.*, Vol. 21, No. 2, 1978. pp. 120-126.

Saltman, Roy G., ed. *Good Security Practices for Electronic Commerce, Including Electronic Data interchange*. Special Publication 800-9. Gaithersburg, MD: National Institute of Standards and Technology. December 1993.

Schneier, B. "A Taxonomy of Encryption Algorithms." *Computer Security Journal*. 9(1), 1193. pp. 39-60.

Schneier, B. "Four Crypto Standards." *Infosecurity News*. 4(2), 1993. pp. 38-39.

Schneier, B. *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. New York, NY: John Wiley & Sons, Inc., 1994.

U.S. Congress, Office of Technology Assessment. "Security Safeguards and Practices." *Defending Secrets, Sharing Data: New Locks and Keys for Electronic Information*. Washington, DC: 1987, pp. 54-72.

V. EXAMPLE

Chapter 20

ASSESSING AND MITIGATING THE RISKS TO A HYPOTHETICAL COMPUTER SYSTEM

This chapter illustrates how a hypothetical government agency (HGA) deals with computer security issues in its operating environment.[140] It follows the evolution of HGA's initiation of an assessment of the threats to its computer security system all the way through to HGA's recommendations for mitigating those risks. In the real world, many solutions exist for computer security problems. No single solution can solve similar security problems in all environments. Likewise, the solutions presented in this example may not be appropriate for all environments.

This example can be used to help understand how security issues are examined, how some potential solutions are analyzed, how their cost and benefits are weighed, and ultimately how management accepts responsibility for risks.

This case study is provided for illustrative purposes only, and should not be construed as guidance or specific recommendations to solving specific security issues. Because a comprehensive example attempting to illustrate all handbook topics would be inordinately long, this example necessarily simplifies the issues presented and omits many details. For instance, to highlight the similarities and differences among controls in the different processing environments, it addresses some of the major types of processing platforms linked together in a distributed system: personal computers, local-area networks, wide-area networks, and mainframes; it does not show how to secure these platforms.

This section also highlights the importance of management's acceptance of a particular level of risk—this will, of course, vary from organization to organization. It is management's prerogative to decide what level of risk is appropriate, given operating and budget environments and other applicable factors.

20.1 Initiating the Risk Assessment

HGA has information systems that comprise and are intertwined with several different kinds of assets valuable enough to merit protection. HGA's systems play a key role in transferring U.S. Government funds to individuals in the form of paychecks; hence, financial resources are among the assets associated with HGA's systems. The system components owned and operated by HGA are also assets, as are personnel information, contracting and procurement documents, draft

---

[140] While this chapter draws upon many actual systems, details and characteristics were changed and merged. Although the chapter is arranged around an agency, the case study could also apply to a large division or office within an agency.

regulations, internal correspondence, and a variety of other day-to-day business documents, memos, and reports. HGA's assets include intangible elements as well, such as reputation of the agency and the confidence of its employees that personal information will be handled properly and that the wages will be paid on time.

A recent change in the directorship of HGA has brought in a new management team. Among the new Chief Information Officer's first actions was appointing a Computer Security Program Manager who immediately initiated a comprehensive risk analysis to assess the soundness of HGA's computer security program in protecting the agency's assets and its compliance with federal directives. This analysis drew upon prior risk assessments, threat studies, and applicable internal control reports. The Computer Security Program Manager also established a timetable for periodic reassessments.

Since the wide-area network and mainframe used by HGA are owned and operated by other organizations, they were not treated in the risk assessment as HGA's assets. And although HGA's personnel, buildings, and facilities are essential assets, the Computer Security Program Manager considered them to be outside the scope of the risk analysis.

After examining HGA's computer system, the risk assessment team identified specific threats to HGA's assets, reviewed HGA's and national safeguards against those threats, identified the vulnerabilities of those policies, and recommended specific actions for mitigating the remaining risks to HGA's computer security. The following sections provide highlights from the risk assessment. The assessment addressed many other issues at the programmatic and system levels. However, this chapter focuses on security issues related to the time and attendance application. (Other issues are discussed in Chapter 6.)

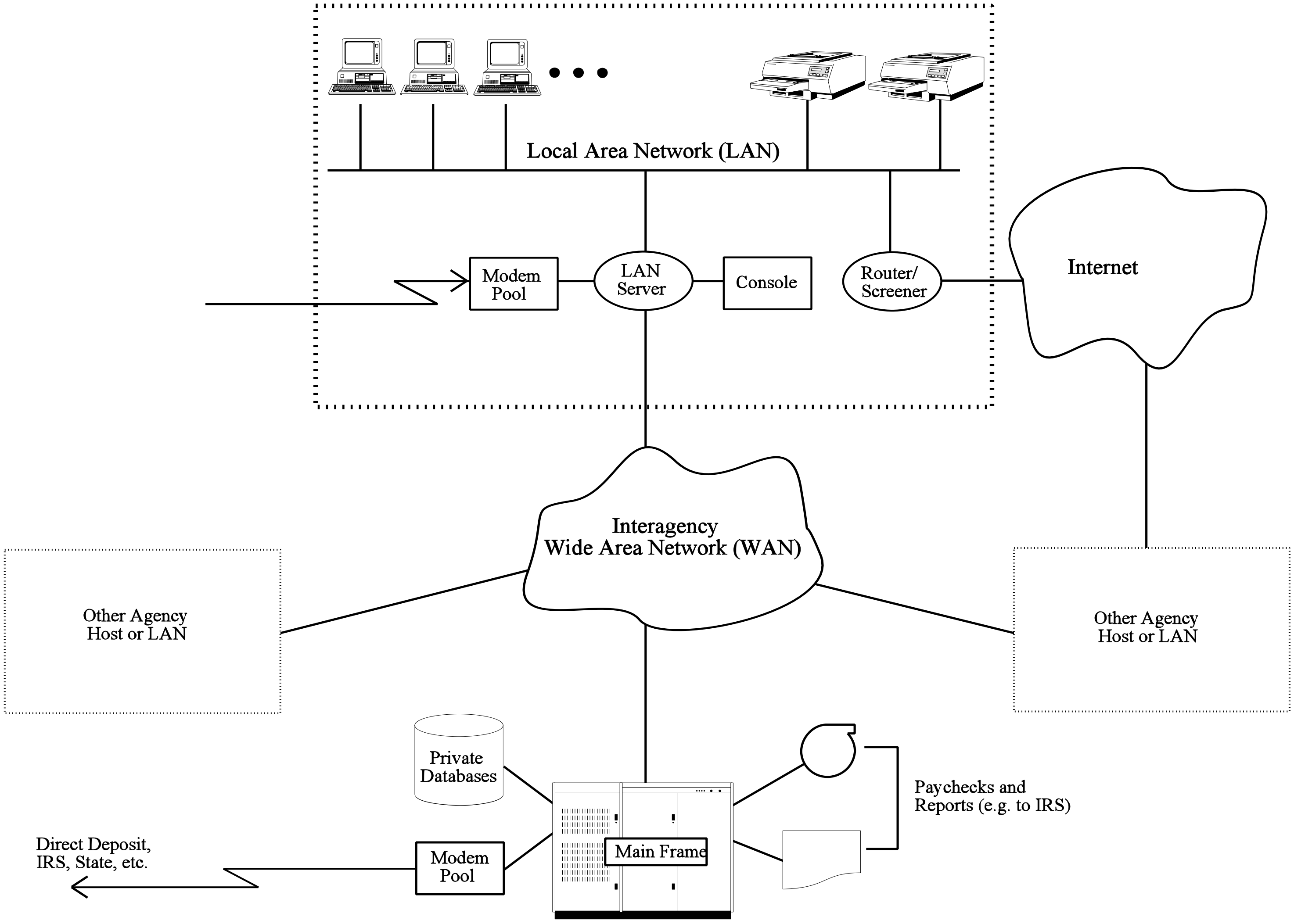
20.2 HGA's Computer System

HGA relies on the distributed computer systems and networks shown in Figure 20.1. They consist of a collection of components, some of which are systems in their own right. Some belong to HGA, but others are owned and operated by other organizations. This section describes these components, their role in the overall distributed system architecture, and how they are used by HGA.

20.2.1 System Architecture

Most of HGA's staff (a mix of clerical, technical, and managerial staff) are provided with personal computers (PCs) located in their offices. Each PC includes hard-disk and floppy-disk drives.

The PCs are connected to a local area network (LAN) so that users can exchange and share


Local Area Network (LAN)
Modem Pool
LAN Server
Console
Router/ Screener
Internet
Interagency Wide Area Network (WAN)
Other Agency Host or LAN
Other Agency Host or LAN
Private Databases
Main Frame
Modem Pool
Direct Deposit, IRS, State, etc.
Paychecks and Reports (e.g. to IRS)

information. The central component of the LAN is a *LAN server*, a more powerful computer that acts as an intermediary between PCs on the network and provides a large volume of disk storage for shared information, including shared application programs. The server provides logical access controls on potentially sharable information via elementary access control lists. These access controls can be used to limit user access to various files and programs stored on the server. Some programs stored on the server can be retrieved via the LAN and executed on a PC; others can only be executed on the server.

To initiate a session on the network or execute programs on the server, users at a PC must log into the server and provide a user identifier and password known to the server. Then they may use files to which they have access.

One of the applications supported by the server is *electronic mail* (e-mail), which can be used by all PC users. Other programs that run on the server can only be executed by a limited set of PC users.

Several printers, distributed throughout HGA's building complex, are connected to the LAN. Users at PCs may direct printouts to whichever printer is most convenient for their use.

Since HGA must frequently communicate with industry, the LAN also provides a connection to the Internet via a *router*. The router is a network interface device that translates between the protocols and addresses associated with the LAN and the Internet. The router also performs *network packet filtering*, a form of network access control, and has recently been configured to disallow non–e-mail (e.g., file transfer, remote log-in) between LAN and Internet computers.

The LAN server also has connections to several other devices.

- A *modem pool* is provided so that HGA's employees on travel can "dial up" via the public switched (telephone) network and read or send e-mail. To initiate a dial-up session, a user must successfully log in. During dial-up sessions, the LAN server provides access only to e-mail facilities; no other functions can be invoked.

- A *special console* is provided for the server administrators who configure the server, establish and delete user accounts, and have other special privileges needed for administrative and maintenance functions. These functions can only be invoked from the *administrator console*; that is, they cannot be invoked from a PC on the network or from a dial-up session.

- A *connection to a government agency X.25-based wide-area network* (WAN) is provided so that information can be transferred to or from other agency systems. One of the other hosts on the WAN is a large multiagency mainframe system. This mainframe is used to collect and process information from a large number of agencies while providing a range of access controls.

20.2.2 System Operational Authority/Ownership

The system components contained within the large dashed rectangle shown in Figure 20.1 are managed and operated by an organization within HGA known as the Computer Operations Group (COG). This group includes the PCs, LAN, server, console, printers, modem pool, and router. The WAN is owned and operated by a large commercial telecommunications company that provides WAN services under a government contract. The mainframe is owned and operated by a federal agency that acts as a service provider for HGA and other agencies connected to the WAN.

20.2.3 System Applications

PCs on HGA's LAN are used for word processing, data manipulation, and other common applications, including spreadsheet and project management tools. Many of these tasks are concerned with data that are sensitive with respect to confidentiality or integrity. Some of these documents and data also need to be available in a timely manner.

The mainframe also provides storage and retrieval services for other databases belonging to individual agencies. For example, several agencies, including HGA, store their personnel databases on the mainframe; these databases contain dates of service, leave balances, salary and W-2 information, and so forth.

In addition to their time and attendance application, HGA's PCs and the LAN server are used to manipulate other kinds of information that may be sensitive with respect to confidentiality or integrity, including personnel-related correspondence and draft contracting documents.

20.3 Threats to HGA's Assets

Different assets of HGA are subject to different kinds of threats. Some threats are considered less likely than others, and the potential impact of different threats may vary greatly. The likelihood of threats is generally difficult to estimate accurately. Both HGA and the risk assessment's authors have attempted to the extent possible to base these estimates on historical data, but have also tried to anticipate new trends stimulated by emerging technologies (e.g., external networks).

20.3.1 Payroll Fraud

As for most large organizations that control financial assets, attempts at fraud and embezzlement are likely to occur. Historically, attempts at payroll fraud have almost always come from within HGA or the other agencies that operate systems on which HGA depends. Although HGA has thwarted many of these attempts, and some have involved relatively small sums of money, it considers preventing financial fraud to be a *critical* computer security priority, particularly in light of the potential financial losses and the risks of damage to its reputation with Congress, the public, and other federal agencies.

Attempts to defraud HGA have included the following:

- Submitting fraudulent time sheets for hours or days not worked, or for pay periods following termination or transfer of employment. The former may take the form of overreporting compensatory or overtime hours worked, or underreporting vacation or sick leave taken. Alternatively, attempts have been made to modify time sheet data after being entered and approved for submission to payroll.

- Falsifying or modifying dates or data on which one's "years of service" computations are based, thereby becoming eligible for retirement earlier than allowed, or increasing one's pension amount.

- Creating employee records and time sheets for fictitious personnel, and attempting to obtain their paychecks, particularly after arranging for direct deposit.

20.3.2 Payroll Errors

Of greater likelihood, but of perhaps lesser potential impact on HGA, are errors in the entry of time and attendance data; failure to enter information describing new employees, terminations, and transfers in a timely manner; accidental corruption or loss of time and attendance data; or errors in interagency coordination and processing of personnel transfers.

Errors of these kinds can cause financial difficulties for employees and accounting problems for HGA. If an employee's vacation or sick leave balance became negative erroneously during the last pay period of the year, the employee's last paycheck would be automatically reduced. An individual who transfers between HGA and another agency may risk receiving duplicate paychecks or no paychecks for the pay periods immediately following the transfer. Errors of this sort that occur near the end of the year can lead to errors in W-2 forms and subsequent difficulties with the tax collection agencies.

20.3.3 Interruption of Operations

HGA's building facilities and physical plant are several decades old and are frequently under repair or renovation. As a result, power, air conditioning, and LAN or WAN connectivity for the server are typically interrupted several times a year for periods of up to one work day. For example, on several occasions, construction workers have inadvertently severed power or network cables. Fires, floods, storms, and other natural disasters can also interrupt computer operations, as can equipment malfunctions.

Another threat of small likelihood, but significant potential impact, is that of a malicious or disgruntled employee or outsider seeking to disrupt time-critical processing (e.g., payroll) by deleting necessary inputs or system accounts, misconfiguring access controls, planting computer viruses, or stealing or sabotaging computers or related equipment. Such interruptions, depending upon when they occur, can prevent time and attendance data from getting processed and transferred to the mainframe before the payroll processing deadline.

20.3.4 Disclosure or Brokerage of Information

Other kinds of threats may be stimulated by the growing market for information about an organization's employees or internal activities. Individuals who have legitimate work-related reasons for access to the master employee database may attempt to disclose such information to other employees or contractors or to sell it to private investigators, employment recruiters, the press, or other organizations. HGA considers such threats to be moderately likely and of low to high potential impact, depending on the type of information involved.

20.3.5 Network-Related Threats

Most of the human threats of concern to HGA originate from insiders. Nevertheless, HGA also recognizes the need to protect its assets from outsiders. Such attacks may serve many different purposes and pose a broad spectrum of risks, including unauthorized disclosure or modification of information, unauthorized use of services and assets, or unauthorized denial of services.

As shown in Figure 20.1, HGA's systems are connected to the three external networks: (1) the Internet, (2) the Interagency WAN, and (3) the public-switched (telephone) network. Although these networks are a source of security risks, connectivity with them is essential to HGA's mission and to the productivity of its employees; connectivity cannot be terminated simply because of security risks.

In each of the past few years before establishing its current set of network safeguards, HGA had detected several attempts by outsiders to penetrate its systems. Most, but not all of these, have come from the Internet, and those that succeeded did so by learning or guessing user account passwords. In two cases, the attacker deleted or corrupted significant amounts of data, most of which were later restored from backup files. In most cases, HGA could detect no ill effects of the attack, but concluded that the attacker may have browsed through some files. HGA also conceded that its systems did not have audit logging capabilities sufficient to track an attacker's activities. Hence, for most of these attacks, HGA could not accurately gauge the extent of penetration.

In one case, an attacker made use of a bug in an e-mail utility and succeeded in acquiring System Administrator privileges on the server—a significant breach. HGA found no evidence that the attacker attempted to exploit these privileges before being discovered two days later. When the attack was detected, COG immediately contacted the HGA's Incident Handling Team, and was told that a bug fix had been distributed by the server vendor several months earlier. To its embarrassment, COG discovered that it had already received the fix, which it then promptly installed. It now believes that no subsequent attacks of the same nature have succeeded.

Although HGA has no evidence that it has been significantly harmed to date by attacks via external networks, it believes that these attacks have great potential to inflict damage. HGA's management considers itself lucky that such attacks have not harmed HGA's reputation and the confidence of the citizens its serves. It also believes the likelihood of such attacks via external networks will increase in the future.

20.3.6 Other Threats

HGA's systems also are exposed to several other threats that, for reasons of space, cannot be fully enumerated here. Examples of threats and HGA's assessment of their probabilities and impacts include those listed in Table 20.1.

20.4 Current Security Measures

HGA has numerous policies and procedures for protecting its assets against the above threats. These are articulated in HGA's *Computer Security Manual*, which implements and synthesizes the requirements of many federal directives, such as Appendix III to OMB Circular A-130, the Computer Security Act of 1987, and the Privacy Act. The manual also includes policies for automated financial systems, such as those based on OMB Circulars A-123 and A-127, as well as the Federal Managers' Financial Integrity Act.

Several examples of those policies follow, as they apply generally to the use and administration of HGA's computer system and specifically to security issues related to time and attendance, payroll, and continuity of operations.

20.4.1 General Use and Administration of HGA's Computer System

HGA's Computer Operations Group (COG) is responsible for controlling, administering, and maintaining the computer resources owned and operated by HGA. These functions are depicted in Figure 20.1 enclosed in the large, dashed rectangle. Only individuals holding the job title System Administrator are authorized to establish log-in IDs and passwords on multiuser HGA systems (e.g., the LAN server). Only HGA's employees and contract personnel may use the system, and only after receiving written authorization from the department supervisor (or, in the case of contractors, the contracting officer) to whom these individuals report.

COG issues copies of all relevant security policies and procedures to new users. Before activating a system account for a new users, COG requires that they (1) attend a security awareness and training course or complete an interactive computer-aided-instruction training session and (2) sign an acknowledgment form indicating that they understand their security responsibilities.

Authorized users are assigned a secret log-in ID and password, which they must not share with anyone else. They are expected to comply with all of HGA's password selection and security procedures (e.g., periodically changing passwords). Users who fail to do so are subject to a range of penalties.

Examples of Threats to HGA Systems

Potential Threat	**Probability**	**Impact**
Accidental Loss/Release of Disclosure-Sensitive Information	Medium	Low/Medium
Accidental Destruction of Information	High	Medium
Loss of Information due to Virus Contamination	Medium	Medium
Misuse of System Resources	Low	Low
Theft	High	Medium
Unauthorized Access to Telecommunications Resources[*]	Medium	Medium
Natural Disaster	Low	High

[*] HGA operates a PBX system, which may be vulnerable to (1) hacker disruptions of PBX availability and, consequently, agency operations, (2) unauthorized access to outgoing phone lines for long-distance services, (3) unauthorized access to stored voice-mail messages, and (4) surreptitious access to otherwise private conversations/data transmissions.

Table 20.1

Users creating data that are sensitive with respect to disclosure or modification are expected to make effective use of the automated access control mechanisms available on HGA computers to reduce the risk of exposure to unauthorized individuals. (Appropriate training and education are in place to help users do this.) In general, access to disclosure-sensitive information is to be granted only to individuals whose jobs require it.

20.4.2 Protection Against Payroll Fraud and Errors: Time and Attendance Application

The time and attendance application plays a major role in protecting against payroll fraud and errors. Since the time and attendance application is a component of a larger automated payroll process, many of its functional and security requirements have been derived from both governmentwide and HGA-specific policies related to payroll and leave. For example, HGA must protect personal information in accordance with the Privacy Act. Depending on the specific type of information, it should normally be viewable only by the individual concerned, the individual's supervisors, and personnel and payroll department employees. Such information should also be timely and accurate.

Each week, employees must sign and submit a time sheet that identifies the number of hours they have worked and the amount of leave they have taken. The Time and Attendance Clerk enters the data for a given group of employees and runs an application on the LAN server to verify the data's validity and to ensure that only authorized users with access to the Time and Attendance Clerk's functions can enter time and attendance data. The application performs these security checks by using the LAN server's access control and identification and authentication (I&A) mechanisms. The application compares the data with a limited database of employee information to detect incorrect employee identifiers, implausible numbers of hours worked, and so forth. After correcting any detected errors, the clerk runs another application that formats the time and attendance data into a report, flagging exception/out-of-bound conditions (e.g., negative leave balances).

Department supervisors are responsible for reviewing the correctness of the time sheets of the employees under their supervision and indicating their approval by initialing the time sheets. If they detect significant irregularities and indications of fraud in such data, they must report their findings to the Payroll Office before submitting the time sheets for processing. In keeping with the principle of separation of duty, all data on time sheets and corrections on the sheets that may affect pay, leave, retirement, or other benefits of an individual must be reviewed for validity by at least two authorized individuals (other than the affected individual).

Protection Against Unauthorized Execution

Only users with access to Time and Attendance Supervisor functions may approve and submit time and attendance data — or subsequent corrections thereof — to the mainframe. Supervisors may not approve their own time and attendance data.

Only the System Administrator has been granted access to assign a special access control privilege to server programs. As a result, the server's operating system is designed to prevent a bogus time and attendance application created by any other user from communicating with the WAN and, hence, with the mainframe.

The time and attendance application is supposed to be configured so that the clerk and supervisor functions can only be carried out from specific PCs attached to the LAN and only during normal working hours. Administrators are not authorized to exercise functions of the time and attendance application apart from those concerned with configuring the accounts, passwords, and access permissions for clerks and supervisors. Administrators are expressly prohibited by policy from entering, modifying, or submitting time and attendance data via the time and attendance application or other mechanisms.[141]

Protection against unauthorized execution of the time and attendance application depends on I&A and access controls. While the time and attendance application is accessible from any PC, unlike most programs run by PC users, it does not execute directly on the PC's processor. Instead, it executes on the server, while the PC behaves as a terminal, relaying the user's keystrokes to the server and displaying text and graphics sent from the server. The reason for this approach is that common PC systems do not provide I&A and access controls and, therefore, cannot protect against unauthorized time and attendance program execution. *Any* individual who has access to the PC could run any program stored there.

Another possible approach is for the time and attendance program to perform I&A and access control on its own by requesting and validating a password before beginning each time and attendance session. This approach, however, can be defeated easily by a moderately skilled programming attack, and was judged inadequate by HGA during the application's early design phase.

Recall that the server is a more powerful computer equipped with a multiuser operating system that includes password-based I&A and access controls. Designing the time and attendance application program so that it executes on the server under the control of the server's operating system provides a more effective safeguard against unauthorized execution than executing it on the user's PC.

Protection Against Payroll Errors

The frequency of data entry errors is reduced by having Time and Attendance clerks enter each time sheet into the time and attendance application twice. If the two copies are identical, both are considered error free, and the record is accepted for subsequent review and approval by a supervisor. If the copies are not identical, the discrepancies are displayed, and for each discrepancy, the clerk determines which copy is correct. The clerk then incorporates the corrections into one of the copies, which is then accepted for further processing. If the clerk makes the same data-entry error twice, then the two copies will match, and one will be accepted

---

[141] Technically, Systems Administrators may still have the ability to do so. This highlights the importance of adequate managerial reviews, auditing, and personnel background checks.

as correct, even though it is erroneous. To reduce this risk, the time and attendance application could be configured to require that the two copies be entered by different clerks.

In addition, each department has one or more Time and Attendance Supervisors who are authorized to review these reports for accuracy and to approve them by running another server program that is part of the time and attendance application. The data are then subjected to a collection of "sanity checks" to detect entries whose values are outside expected ranges. Potential anomalies are displayed to the supervisor prior to allowing approval; if errors are identified, the data are returned to a clerk for additional examination and corrections.

When a supervisor approves the time and attendance data, this application logs into the interagency mainframe via the WAN and transfers the data to a payroll database on the mainframe. The mainframe later prints paychecks or, using a pool of modems that can send data over phone lines, it may transfer the funds electronically into employee-designated bank accounts. Withheld taxes and contributions are also transferred electronically in this manner.

The Director of Personnel is responsible for ensuring that forms describing significant payroll-related personnel actions are provided to the Payroll Office at least one week before the payroll processing date for the first affected pay period. These actions include hiring, terminations, transfers, leaves of absences and returns from such, and pay raises.

The Manager of the Payroll Office is responsible for establishing and maintaining controls adequate to ensure that the amounts of pay, leave, and other benefits reported on pay stubs and recorded in permanent records and those distributed electronically are accurate and consistent with time and attendance data and with other information provided by the Personnel Department. In particular, paychecks must never be provided to anyone who is not a bona fide, active-status employee of HGA. Moreover, the pay of any employee who terminates employment, who transfers, or who goes on leave without pay must be suspended as of the effective date of such action; that is, extra paychecks or excess pay must not be dispersed.

Protection Against Accidental Corruption or Loss of Payroll Data

The same mechanisms used to protect against fraudulent modification are used to protect against accidental corruption of time and attendance data — namely, the access-control features of the server and mainframe operating systems.

COG's nightly backups of the server's disks protect against loss of time and attendance data. To a limited extent, HGA also relies on mainframe administrative personnel to back up time and attendance data stored on the mainframe, even though HGA has no direct control over these individuals. As additional protection against loss of data at the mainframe, HGA retains copies of all time and attendance data on line on the server for at least one year, at which time the data are archived and kept for three years. The server's access controls for the on-line files are automatically set to read-only access by the time and attendance application at the time of submission to the mainframe. The integrity of time and attendance data will be protected by digital signatures as they are implemented.

The WAN's communications protocols also protect against loss of data during transmission from the server to the mainframe (e.g., error checking). In addition, the mainframe payroll application includes a program that is automatically run 24 hours before paychecks and pay stubs are printed. This program produces a report identifying agencies from whom time and attendance data for the current pay period were expected but not received. Payroll department staff are responsible for reviewing the reports and immediately notifying agencies that need to submit or resubmit time and attendance data. If time and attendance input or other related information is not available on a timely basis, pay, leave, and other benefits are temporarily calculated based on information estimated from prior pay periods.

20.4.3 Protection Against Interruption of Operations

HGA's policies regarding continuity of operations are derived from requirements stated in OMB Circular A-130. HGA requires various organizations within it to develop contingency plans, test them annually, and establish appropriate administrative and operational procedures for supporting them. The plans must identify the facilities, equipment, supplies, procedures, and personnel needed to ensure reasonable continuity of operations under a broad range of adverse circumstances.

COG Contingency Planning

COG is responsible for developing and maintaining a contingency plan that sets forth the procedures and facilities to be used when physical plant failures, natural disasters, or major equipment malfunctions occur sufficient to disrupt the normal use of HGA's PCs, LAN, server, router, printers, and other associated equipment.

The plan prioritizes applications that rely on these resources, indicating those that should be suspended if available automated functions or capacities are temporarily degraded. COG personnel have identified system software and hardware components that are compatible with those used by two nearby agencies. HGA has signed an agreement with those agencies, whereby they have committed to reserving spare computational and storage capacities sufficient to support HGA's system-based operations for a few days during an emergency.

No communication devices or network interfaces may be connected to HGA's systems without written approval of the COG Manager. The COG staff is responsible for installing all known security-related software patches in a timely manner and for maintaining spare or redundant PCs, servers, storage devices, and LAN interfaces to ensure that at least 100 people can simultaneously perform word processing tasks at all times.

To protect against accidental corruption or loss of data, COG personnel back up the LAN server's disks onto magnetic tape every night and transport the tapes weekly to a sister agency for storage. HGA's policies also stipulate that all PC users are responsible for backing up weekly any significant data stored on their PC's local hard disks. For the past several years, COG has issued a yearly memorandum reminding PC users of this responsibility. COG also strongly encourages them to store significant data on the LAN server instead of on their PC's hard disk so that such data will be backed up automatically during COG's LAN server backups.

To prevent more limited computer equipment malfunctions from interrupting routine business operations, COG maintains an inventory of approximately ten fully equipped spare PC's, a spare LAN server, and several spare disk drives for the server. COG also keeps thousands of feet of LAN cable on hand. If a segment of the LAN cable that runs through the ceilings and walls of HGA's buildings fails or is accidentally severed, COG technicians will run temporary LAN cabling along the floors of hallways and offices, typically restoring service within a few hours for as long as needed until the cable failure is located and repaired.

To protect against PC virus contamination, HGA authorizes only System Administrators approved by the COG Manager to install licensed, copyrighted PC software packages that appear on the COG-approved list. PC software applications are generally installed only on the server. (These stipulations are part of an HGA assurance strategy that relies on the quality of the engineering practices of vendors to provide software that is adequately robust and trustworthy.) Only the COG Manager is authorized to add packages to the approved list. COG procedures also stipulate that every month System Administrators should run virus-detection and other security-configuration validation utilities on the server and, on a spot-check basis, on a number of PCs. If they find a virus, they must immediately notify the agency team that handles computer security incidents.

COG is also responsible for reviewing audit logs generated by the server, identifying audit records indicative of security violations, and reporting such indications to the Incident-Handling Team. The COG Manager assigns these duties to specific members of the staff and ensures that they are implemented as intended.

The COG Manager is responsible for assessing adverse circumstances and for providing recommendations to HGA's Director. Based on these and other sources of input, the Director will determine whether the circumstances are dire enough to merit activating various sets of procedures called for in the contingency plan.

Division Contingency Planning

HGA's divisions also must develop and maintain their own contingency plans. The plans must identify critical business functions, the system resources and applications on which they depend, and the maximum acceptable periods of interruption that these functions can tolerate without significant reduction in HGA's ability to fulfill its mission. The head of each division is responsible for ensuring that the division's contingency plan and associated support activities are adequate.

For each major application used by multiple divisions, a chief of a single division must be designated as the *application owner*. The designated official (supported by his or her staff) is responsible for addressing that application in the contingency plan and for coordinating with other divisions that use the application.

If a division relies exclusively on computer resources maintained by COG (e.g., the LAN), it need not duplicate COG's contingency plan, but is responsible for reviewing the adequacy of that plan. If COG's plan does not adequately address the division's needs, the division must communicate its concerns to the COG Director. In either situation, the division must make known the criticality of its applications to the COG. If the division relies on computer resources or services that are *not* provided by COG, the division is responsible for (1) developing its own contingency plan or (2) ensuring that the contingency plans of other organizations (e.g., the WAN service provider) provide adequate protection against service disruptions.

20.4.4 Protection Against Disclosure or Brokerage of Information

HGA's protection against information disclosure is based on a need-to-know policy and on personnel hiring and screening practices. The need-to-know policy states that time and attendance information should be made accessible only to HGA employees and contractors whose assigned professional responsibilities require it. Such information must be protected against access from all other individuals, including other HGA employees. Appropriate hiring and screening practices can lessen the risk that an untrustworthy individual will be assigned such responsibilities.

The need-to-know policy is supported by a collection of physical, procedural, and automated safeguards, including the following:

- Time and attendance paper documents are must be stored securely when not in use, particularly during evenings and on weekends. Approved storage containers include locked file cabinets and desk drawers—to which only the owner has the keys. While storage in a container is preferable, it is also permissible to leave time and attendance documents on top of a desk or other exposed surface in a locked office (with the realization that the guard force has keys to the office). (This is a judgment left to local discretion.) Similar rules apply to disclosure-sensitive information stored on floppy disks and other removable magnetic media.
- Every HGA PC is equipped with a key lock that, when locked, disables the PC.

When information is stored on a PC's local hard disk, the user to whom that PC was assigned is expected to (1) lock the PC at the conclusion of each work day and (2) lock the office in which the PC is located.

- The LAN server operating system's access controls provide extensive features for controlling access to files. These include group-oriented controls that allow teams of users to be assigned to named groups by the System Administrator. Group members are then allowed access to sensitive files not accessible to nonmembers. Each user can be assigned to several groups according to need to know. (The reliable functioning of these controls is assumed, perhaps incorrectly, by HGA.)

- All PC users undergo security awareness training when first provided accounts on the LAN server. Among other things, the training stresses the necessity of protecting passwords. It also instructs users to log off the server before going home at night or before leaving the PC unattended for periods exceeding an hour.

20.4.5 Protection Against Network-Related Threats

HGA's current set of external network safeguards has only been in place for a few months. The basic approach is to tightly restrict the kinds of external network interactions that can occur by funneling all traffic to and from external networks through two interfaces that filter out unauthorized kinds of interactions. As indicated in Figure 20.1, the two interfaces are the network router and the LAN server. The only kinds of interactions that these interfaces allow are (1) e-mail and (2) data transfers from the server to the mainframe controlled by a few special applications (e.g., the time and attendance application).

Figure 20.1 shows that the network router is the only direct interface between the LAN and the Internet. The router is a dedicated special-purpose computer that translates between the protocols and addresses associated with the LAN and the Internet. Internet protocols, unlike those used on the WAN, specify that packets of information coming from or going to the Internet must carry an indicator of the kind of service that is being requested or used to process the information. This makes it possible for the router to distinguish e-mail packets from other kinds of packets—for example, those associated with a remote log-in request.[142] The router has been configured by COG to discard all packets coming from or going to the Internet, except those associated with e-mail. COG personnel believe that the router effectively eliminates Internet-based attacks on HGA user accounts because it disallows all remote log-in sessions, even those accompanied by a legitimate password.

---

[142] Although not discussed in this example, recognize that technical "spoofing" can occur.

The LAN server enforces a similar type of restriction for dial-in access via the public-switched network. The access controls provided by the server's operating system have been configured so that during dial-in sessions, only the e-mail utility can be executed. (HGA policy, enforced by periodic checks, prohibits installation of modems on PCs, so that access must be through the LAN server.) In addition, the server's access controls have been configured so that its WAN interface device is accessible only to programs that possess a special access-control privilege. Only the System Administrator can assign this privilege to server programs, and only a handful of special-purpose applications, like the time and attendance application, have been assigned this privilege.

20.4.6 Protection Against Risks from Non–HGA Computer Systems

HGA relies on systems and components that it cannot control directly because they are owned by other organizations. HGA has developed a policy to avoid undue risk in such situations. The policy states that system components controlled and operated by organizations other than HGA may not be used to process, store, or transmit HGA information without obtaining explicit permission from the application owner and the COG Manager. Permission to use such system components may not be granted without written commitment from the controlling organization that HGA's information will be safeguarded commensurate with its value, as designated by HGA. This policy is somewhat mitigated by the fact that HGA has developed an issue-specific policy on the use of the Internet, which allows for its use for e-mail with outside organizations and access to other resources (but not for transmission of HGA's proprietary data).

20.5 Vulnerabilities Reported by the Risk Assessment Team

The risk assessment team found that many of the risks to which HGA is exposed stem from (1) the failure of individuals to comply with established policies and procedures or (2) the use of automated mechanisms whose assurance is questionable because of the ways they have been developed, tested, implemented, used, or maintained. The team also identified specific vulnerabilities in HGA's policies and procedures for protecting against payroll fraud and errors, interruption of operations, disclosure and brokering of confidential information, and unauthorized access to data by outsiders.

20.5.1 Vulnerabilities Related to Payroll Fraud

Falsified Time Sheets

The primary safeguards against falsified time sheets are review and approval by supervisory personnel, who are not permitted to approve their own time and attendance data. The risk assessment has concluded that, while imperfect, these safeguards are adequate. The related requirement that a clerk and a supervisor must cooperate closely in creating time and attendance data and submitting the data to the mainframe also safeguards against other kinds of illicit manipulation of time and attendance data by clerks or supervisors acting independently.

Unauthorized Access

When a PC user enters a password to the server during I&A, the password is sent to the server by broadcasting it over the LAN "in the clear." This allows the password to be intercepted easily by any other PC connected to the LAN. In fact, so-called "password sniffer" programs that capture passwords in this way are widely available. Similarly, a malicious program planted on a PC could also intercept passwords before transmitting them to the server. An unauthorized individual who obtained the captured passwords could then run the time and attendance application in place of a clerk or supervisor. Users might also store passwords in a log-on script file.

Bogus Time and Attendance Applications

The server's access controls are probably adequate for protection against bogus time and attendance applications that run on the server. However, the server's operating system and access controls have only been in widespread use for a few years and contain a number of security-related bugs. And the server's access controls are ineffective if not properly configured, and the administration of the server's security features in the past has been notably lax.

Unauthorized Modification of Time and Attendance Data

Protection against unauthorized modification of time and attendance data requires a variety of safeguards because each system component on which the data are stored or transmitted is a potential source of vulnerabilities.

First, the time and attendance data are entered on the server by a clerk. On occasion, the clerk may begin data entry late in the afternoon, and complete it the following morning, storing it in a temporary file between the two sessions. One way to avoid unauthorized modification is to store the data on a diskette and lock it up overnight. After being entered, the data will be stored in another temporary file until reviewed and approved by a supervisor. These files, now stored on the system, must be protected against tampering. As before, the server's access controls, if reliable and properly configured, can provide such protection (as can digital signatures, as discussed later) in conjunction with proper auditing.

Second, when the Supervisor approves a batch of time and attendance data, the time and attendance application sends the data over the WAN to the mainframe. The WAN is a collection of communications equipment and special-purpose computers called "switches" that act as relays, routing information through the network from source to destination. Each switch is a potential site at which the time and attendance data may be fraudulently modified. For example, an HGA PC user might be able to intercept time and attendance data and modify the data enroute to the payroll application on the mainframe. Opportunities include tampering with incomplete time and attendance input files while stored on the server, interception and tampering during WAN transit, or tampering on arrival to the mainframe prior to processing by the payroll application.

Third, on arrival at the mainframe, the time and attendance data are held in a temporary file on the mainframe until the payroll application is run. Consequently, the mainframe's I&A and access controls must provide a critical element of protection against unauthorized modification of the data.

According to the risk assessment, the server's access controls, with prior caveats, probably provide acceptable protection against unauthorized modification of data stored on the server. The assessment concluded that a WAN-based attack involving collusion between an employee of HGA and an employee of the WAN service provider, although unlikely, should not be dismissed entirely, especially since HGA has only cursory information about the service provider's personnel security practices and no contractual authority over how it operates the WAN.

The greatest source of vulnerabilities, however, is the mainframe. Although its operating system's access controls are mature and powerful, it uses password-based I&A. This is of particular concern, because it serves a large number of federal agencies via WAN connections. A number of these agencies are known to have poor security programs. As a result, one such agency's systems could be penetrated (e.g., from the Internet) and then used in attacks on the mainframe via the WAN. In fact, time and attendance data awaiting processing on the mainframe would probably not be as attractive a target to an attacker as other kinds of data or, indeed, disabling the system, rendering it unavailable. For example, an attacker might be able to modify the employee data base so that it disbursed paychecks or pensions checks to fictitious employees. Disclosure-sensitive law enforcement databases might also be attractive targets.

The access control on the mainframe is strong and provides good protection against intruders breaking into a second application after they have broken into a first. However, previous audits have shown that the difficulties of system administration may present some opportunities for intruders to defeat access controls.

20.5.2 Vulnerabilities Related to Payroll Errors

HGA's management has established procedures for ensuring the timely submission and interagency coordination of paperwork associated with personnel status changes. However, an unacceptably large number of troublesome payroll errors during the past several years has been traced to the late submission of personnel paperwork. The risk assessment documented the adequacy of HGA's safeguards, but criticized the managers for not providing sufficient incentives for compliance.

20.5.3 Vulnerabilities Related to Continuity of Operations

COG Contingency Planning

The risk assessment commended HGA for many aspects of COG's contingency plan, but pointed out that many COG personnel were completely unaware of the responsibilities the plan assigned to them. The assessment also noted that although HGA's policies require annual testing of contingency plans, the capability to resume HGA's computer-processing activities at another cooperating agency has never been verified and may turn out to be illusory.

Division Contingency Planning

The risk assessment reviewed a number of the application-oriented contingency plans developed by HGA's divisions (including plans related to time and attendance). Most of the plans were cursory and attempted to delegate nearly all contingency planning responsibility to COG. The assessment criticized several of these plans for failing to address potential disruptions caused by lack of access to (1) computer resources not managed by COG and (2) nonsystem resources, such as buildings, phones, and other facilities. In particular, the contingency plan encompassing the time and attendance application was criticized for not addressing disruptions caused by WAN and mainframe outages.

Virus Prevention

The risk assessment found HGA's virus-prevention policy and procedures to be sound, but noted that there was little evidence that they were being followed. In particular, no COG personnel interviewed had ever run a virus scanner on a PC on a routine basis, though several had run them during publicized virus scares. The assessment cited this as a significant risk item.

Accidental Corruption and Loss of Data

The risk assessment concluded that HGA's safeguards against accidental corruption and loss of time and attendance data were adequate, but that safeguards for some other kinds of data were not. The assessment included an informal audit of a dozen randomly chosen PCs and PC users in the agency. It concluded that many PC users store significant data on their PC's hard disks, but do not back them up. Based on anecdotes, the assessment's authors stated that there appear to have been many past incidents of loss of information stored on PC hard disks and predicted that such losses would continue.

20.5.4 Vulnerabilities Related to Information Disclosure/Brokerage

HGA takes a conservative approach toward protecting information about its employees. Since information brokerage is more likely to be a threat to large collections of data, HGA risk assessment focused primarily, but not exclusively, on protecting the mainframe.

The risk assessment concluded that significant, avoidable information brokering vulnerabilities were present—particularly due to HGA's lack of compliance with its own policies and procedures. Time and attendance documents were typically not stored securely after hours, and few PCs containing time and attendance information were routinely locked. Worse yet, few were routinely powered down, and many were left logged into the LAN server overnight. These practices make it easy for an HGA employee wandering the halls after hours to browse or copy time and attendance information on another employee's desk, PC hard disk, or LAN server directories.

The risk assessment pointed out that information sent to or retrieved from the server is subject to eavesdropping by other PCs on the LAN. The LAN hardware transmits information by broadcasting it to all connection points on the LAN cable. Moreover, information sent to or retrieved from the server is transmitted in the clear—that is, without encryption. Given the widespread availability of LAN "sniffer" programs, LAN eavesdropping is trivial for a prospective information broker and, hence, is likely to occur.

Last, the assessment noted that HGA's employee master database is stored on the mainframe, where it might be a target for information brokering by employees of the agency that owns the mainframe. It might also be a target for information brokering, fraudulent modification, or other illicit acts by any outsider who penetrates the mainframe via another host on the WAN.

20.5.5 Network-Related Vulnerabilities

The risk assessment concurred with the general approach taken by HGA, but identified several vulnerabilities. It reiterated previous concerns about the lack of assurance associated with the server's access controls and pointed out that these play a critical role in HGA's approach. The assessment noted that the e-mail utility allows a user to include a copy of *any* otherwise accessible file in an outgoing mail message. If an attacker dialed in to the server and succeeded in logging in as an HGA employee, the attacker could use the mail utility to export copies of all the files accessible to that employee. In fact, copies could be mailed to any host on the Internet.

The assessment also noted that the WAN service provider may rely on microwave stations or satellites as relay points, thereby exposing HGA's information to eavesdropping. Similarly, any information, including passwords and mail messages, transmitted during a dial-in session is subject to eavesdropping.

20.6 Recommendations for Mitigating the Identified Vulnerabilities

The discussions in the following subsections were chosen to illustrate a *broad sampling*[143] of handbook topics. Risk management and security program management themes are integral throughout, with particular emphasis given to the selection of risk-driven safeguards.

20.6.1 Mitigating Payroll Fraud Vulnerabilities

To remove the vulnerabilities related to payroll fraud, the risk assessment team recommended[144] the use of stronger authentication mechanisms based on smart tokens to generate one-time passwords that cannot be used by an interloper for subsequent sessions. Such mechanisms would make it very difficult for outsiders (e.g., from the Internet) who penetrate systems on the WAN to use them to attack the mainframe. The authors noted, however, that the mainframe serves many different agencies, and HGA has no authority over the way the mainframe is configured and operated. Thus, the costs and procedural difficulties of implementing such controls would be substantial. The assessment team also recommended improving the server's administrative procedures and the speed with which security-related bug fixes distributed by the vendor are installed on the server.

After input from COG security specialists and application owners, HGA's managers accepted most of the risk assessment team's recommendations. They decided that since the residual risks from the falsification of time sheets were acceptably low, no changes in procedures were necessary. However, they judged the risks of payroll fraud due to the interceptability of LAN server passwords to be unacceptably high, and thus directed COG to investigate the costs and procedures associated with using one-time passwords for Time and Attendance Clerks and supervisor sessions on the server. Other users performing less sensitive tasks on the LAN would continue to use password-based authentication.

While the immaturity of the LAN server's access controls was judged a significant source of risk, COG was only able to identify one other PC LAN product that would be significantly better in this respect. Unfortunately, this product was considerably less friendly to users and application developers, and incompatible with other applications used by HGA. The negative impact of changing PC LAN products was judged too high for the potential incremental gain in security benefits. Consequently, HGA decided to accept the risks accompanying use of the current product, but directed COG to improve its monitoring of the server's access control configuration and its responsiveness to vendor security reports and bug fixes.

---

[143] Some of the controls, such as auditing and access controls, play an important role in many areas. The limited nature of this example, however, prevents a broader discussion.

[144] Note that, for the sake of brevity, the process of evaluating the cost-effectiveness of various security controls is not specifically discussed.

HGA concurred that risks of fraud due to unauthorized modification of time and attendance data at or in transit to the mainframe should not be accepted unless no practical solutions could be identified. After discussions with the mainframe's owning agency, HGA concluded that the owning agency was unlikely to adopt the advanced authentication techniques advocated in the risk assessment. COG, however, proposed an alternative approach that did not require a major resource commitment on the part of the mainframe owner.

The alternative approach would employ digital signatures based on public key cryptographic techniques to detect unauthorized modification of time and attendance data. The data would be *digitally signed* by the supervisor using a private key prior to transmission to the mainframe. When the payroll application program was run on the mainframe, it would use the corresponding public key to validate the correspondence between the time and attendance data and the signature. Any modification of the data during transmission over the WAN or while in temporary storage at the mainframe would result in a mismatch between the signature and the data. If the payroll application detected a mismatch, it would reject the data; HGA personnel would then be notified and asked to review, sign, and send the data again. If the data and signature matched, the payroll application would process the time and attendance data normally.

HGA's decision to use advanced authentication for time and attendance Clerks and Supervisors can be combined with digital signatures by using smart tokens. Smart tokens are programmable devices, so they can be loaded with private keys and instructions for computing digital signatures without burdening the user. When supervisors approve a batch of time and attendance data, the time and attendance application on the server would instruct the supervisor to insert their token in the token reader/writer device attached to the supervisors' PC. The application would then send a special "hash" (summary) of the time and attendance data to the token via the PC. The token would generate a digital signature using its embedded secret key, and then transfer the signature back to the server, again via the PC. The time and attendance application running on the server would append the signature to the data before sending the data to the mainframe and, ultimately, the payroll application.

Although this approach did not address the broader problems posed by the mainframe's I&A vulnerabilities, it does provide a reliable means of detecting time and attendance data tampering. In addition, it protects against bogus time and attendance submissions from systems connected to the WAN because individuals who lack a time and attendance supervisor's smart token will be unable to generate valid signatures. (Note, however, that the use of digital signatures does require increased administration, particularly in the area of key management.) In summary, digital signatures mitigate risks from a number of different kinds of threats.

HGA's management concluded that digitally signing time and attendance data was a practical, cost-effective way of mitigating risks, and directed COG to pursue its implementation. (They also noted that it would be useful as the agency moved to use of digital signatures in other applications.) This is an example of developing and providing a solution in an environment over which no single entity has overall authority.

20.6.2 Mitigating Payroll Error Vulnerabilities

After reviewing the risk assessment, HGA's management concluded that the agency's current safeguards against payroll errors and against accidental corruption and loss of time and attendance data were adequate. However, the managers also concurred with the risk assessment's conclusions about the necessity for establishing incentives for complying (and penalties for not complying) with these safeguards. They thus tasked the Director of Personnel to ensure greater compliance with paperwork-handling procedures and to provide quarterly compliance audit reports. They noted that the digital signature mechanism HGA plans to use for fraud protection can also provide protection against payroll errors due to accidental corruption.

20.6.3 Mitigating Vulnerabilities Related to the Continuity of Operations

The assessment recommended that COG institute a program of periodic internal training and awareness sessions for COG personnel having contingency plan responsibilities.The assessment urged that COG undertake a rehearsal during the next three months in which selected parts of the plan would be exercised. The rehearsal should include attempting to initiate some aspect of processing activities at one of the designated alternative sites. HGA's management agreed that additional contingency plan training was needed for COG personnel and committed itself to its first plan rehearsal within three months.

After a short investigation, HGA divisions owning applications that depend on the WAN concluded that WAN outages, although inconvenient, would not have a major impact on HGA. This is because the few time-sensitive applications that required WAN-based communication with the mainframe were originally designed to work with magnetic tape instead of the WAN, and could still operate in that mode; hence courier-delivered magnetic tapes could be used as an alternative input medium in case of a WAN outage. The divisions responsible for contingency planning for these applications agreed to incorporate into their contingency plans both descriptions of these procedures and other improvements.

With respect to mainframe outages, HGA determined that it could not easily make arrangements for a suitable alternative site. HGA also obtained and examined a copy of the mainframe facility's own contingency plan. After detailed study, including review by an outside consultant, HGA concluded that the plan had major deficiencies and posed significant risks because of HGA's reliance on it for payroll and other services. This was brought to the attention of the Director of HGA, who, in a formal memorandum to the head of the mainframe's owning agency, called for (1) a high-level interagency review of the plan by all agencies that rely on the mainframe, and (2) corrective action to remedy any deficiencies found.

HGA's management agreed to improve adherence to its virus-prevention procedures. It agreed (from the point of view of the entire agency) that information stored on PC hard disks is frequently lost. It estimated, however, that the labor hours lost as a result would amount to less than a person year—which HGA management does *not* consider to be unacceptable. After reviewing options for reducing this risk, HGA concluded that it would be cheaper to accept the associated loss than to commit significant resources in an attempt to avoid it. COG volunteered, however, to set up an automated program on the LAN server that e-mails backup reminders to all PC users once each quarter. In addition, COG agreed to provide regular backup services for about 5 percent of HGA's PCs; these will be chosen by HGA's management based on the information stored on their hard disks.

20.6.4 Mitigating Threats of Information Disclosure/Brokering

HGA concurred with the risk assessment's conclusions about its exposure to information-brokering risks, and adopted most of the associated recommendations.

The assessment recommended that HGA improve its security awareness training (e.g., via mandatory refresher courses) and that it institute some form of compliance audits. The training should be sure to stress the penalties for noncompliance. It also suggested installing "screen lock" software on PCs that automatically lock a PC after a specified period of idle time in which no keystrokes have been entered; unlocking the screen requires that the user enter a password or reboot the system.

The assessment recommended that HGA modify its information-handling policies so that employees would be required to store some kinds of disclosure-sensitive information only on PC local hard disks (or floppies), but not on the server. This would eliminate or reduce risks of LAN eavesdropping. It was also recommended that an activity log be installed on the server (and regularly reviewed). Moreover, it would avoid unnecessary reliance on the server's access-control features, which are of uncertain assurance. The assessment noted, however, that this strategy conflicts with the desire to store most information on the server's disks so that it is backed up routinely by COG personnel. (This could be offset by assigning responsibility for someone other than the PC owner to make backup copies.) Since the security habits of HGA's PC users have generally been poor, the assessment also recommended use of hard-disk encryption utilities to protect disclosure-sensitive information on unattended PCs from browsing by unauthorized individuals. Also, ways to encrypt information on the server's disks would be studied.

The assessment recommended that HGA conduct a thorough review of the mainframe's safeguards in these respects, and that it regularly review the mainframe audit log, using a query package, with particular attention to records that describe user accesses to HGA's employee master database.

20.6.5 Mitigating Network-Related Threats

The assessment recommended that HGA:

- require stronger I&A for dial-in access or, alternatively, that a restricted version of the mail utility be provided for dial-in, which would prevent a user from including files in outgoing mail messages;
- replace its current modem pool with encrypting modems, and provide each dial-in user with such a modem; and
- work with the mainframe agency to install a similar encryption capability for server-to-mainframe communications over the WAN.

As with previous risk assessment recommendations, HGA's management tasked COG to analyze the costs, benefits, and impacts of addressing the vulnerabilities identified in the risk assessment. HGA eventually adopted some of the risk assessment's recommendations, while declining others. In addition, HGA decided that its policy on handling time and attendance information needed to be clarified, strengthened, and elaborated, with the belief that implementing such a policy would help reduce risks of Internet and dial-in eavesdropping. Thus, HGA developed and issued a revised policy, stating that users are individually responsible for ensuring that they do not transmit disclosure-sensitive information outside of HGA's facilities via e-mail or other means. It also prohibited them from examining or transmitting e-mail containing such information during dial-in sessions and developed and promulgated penalties for noncompliance.

20.7 Summary

This chapter has illustrated how many of the concepts described in previous chapters might be applied in a federal agency. An integrated example concerning a Hypothetical Government Agency (HGA) has been discussed and used as the basis for examining a number of these concepts. HGA's distributed system architecture and its uses were described. The time and attendance application was considered in some detail.

For context, some national and agency-level policies were referenced. Detailed operational policies and procedures for computer systems were discussed and related to these high-level policies. HGA assets and threats were identified, and a detailed survey of selected safeguards, vulnerabilities, and risk mitigation actions were presented. The safeguards included a wide variety of procedural and automated techniques, and were used to illustrate issues of assurance, compliance, security program oversight, and inter-agency coordination.

As illustrated, effective computer security requires clear direction from upper management. Upper management must assign security responsibilities to organizational elements and individuals and must formulate or elaborate the security policies that become the foundation for the organization's security program. These policies must be based on an understanding of the organization's mission priorities and the assets and business operations necessary to fulfill them. They must also be based on a pragmatic assessment of the threats against these assets and operations. A critical element is assessment of threat likelihoods. These are most accurate when derived from historical data, but must also anticipate trends stimulated by emerging technologies.

A good security program relies on an integrated, cost-effective collection of physical, procedural, and automated controls. Cost-effectiveness requires targeting these controls at the threats that pose the highest risks while accepting other residual risks. The difficulty of applying controls properly and in a consistent manner over time has been the downfall of many security programs. This chapter has provided numerous examples in which major security vulnerabilities arose from a lack of assurance or compliance. Hence, periodic compliance audits, examinations of the effectiveness of controls, and reassessments of threats are essential to the success of any organization's security program.

Cross Reference and Index

Interdependencies Cross Reference

The following is a cross reference of the interdependencies sections. Note that the references only include specific controls. Some controls were referenced in groups, such as technical controls and occasionally interdependencies were noted for all controls.

Control	Chapters Where It Is Cited
Policy	Program Management Life Cycle Personnel/User Contingency Awareness and Training Logical Access Audit
Program Management	Policy Awareness and Training
Risk Management	Life Cycle Contingency Incident
Life Cycle	Program Management Assurance
Assurance	Life Cycle Support and Operations Audit Cryptography
Personnel	Training and Awareness Support and Operations Access
Training and Awareness	Personnel/User Incident Support and Operations
Contingency	Incident Support and Operations Physical and Environmental Audit

Incident	Contingency Support and Operations Audit
Physical and Environment	Contingency Support and Operations Logical Access Cryptography
Support and Operations	Contingency Incident
Identification and Authentication	Personnel/User Physical and Environmental Logical Access Audit Cryptography
Access Controls	Policy Personnel/User Physical and Environmental Identification and Authentication Audit Cryptography
Audit	Identification and Authentication Logical Access Cryptography
Cryptography	Identification and Authentication

General Index