Essay 13

Supporting Policies and Functions

Marshall D. Abrams and Harold J. Podell

The major policy objective, to protect information assets against specific harm, usually requires additional policies and functions for support and implementation. This essay discusses supporting policies and functions drawn from the TCSEC, the supporting "rainbow series," and the ITSEC.

The major security policies and functions are discussed in most of the other essays in this collection. This essay is concerned with the support necessary to enable the major security objectives to be achieved. The TCSEC and its "rainbow series" identify the supporting functions and often establish requirements for the presence of such functions in systems designated to protect specified levels and ranges of classified information. The ITSEC identifies similar functions but avoids establishing policy.

In this essay we first present the TCSEC and ITSEC approaches, and then provide detail on each individual policy or function. These supporting policies are sometimes considered part of the infrastructure. Sterne [STER91b], in his discussion of nontraditional security policy, observes that these supporting policies are likely to be applicable independent of policy objectives.

The TCSEC approach

The TCSEC identifies and sets requirements for six supporting policies:

- identification and authentication,
- accountability,
- assurance,
- continuous protection,
- object reuse, and
- covert channels.

The ITSEC approach

The ITSEC does not set policy; in fact, it very explicitly avoids doing so. It relies on external statements of policy, requirements, or security objectives. These external statements specify the security target. Once a security target is established, the ITSEC comes into play as the governing rules by which to determine whether the system or product meets the security target.

The ITSEC provides eight generic headings for groupings of security-enforcing functions. It recommends use of these generic headings to facilitate comparison of security targets and to simplify the work of the evaluators. The recommended headings are:

- identification and authentication,
- access control,
- accountability,
- audit,
- object reuse,
- accuracy,
- reliability of service, and
- data exchange.

The ITSEC recommended headings are a superset of the TCSEC supporting policies. The additions come from consideration of networks and database management systems, plus integrity objectives. While not establishing policy, the ITSEC headings are a good list of security-enforcing functions that should be considered for inclusion in any system or product.

Specific functions and policies

Identification and authentication. In the ITSEC, identification and authentication determine and control the users who are permitted access to controlled resources. This involves verifying the user's claimed identity. The user provides some information that is known by the target of evaluation (TOE) to be associated with the user in question. This heading also includes:

1. any functions to enable new user identities and the associated authentication information to be added, and old user identities to be removed or invalidated;
2. functions to generate, change, or allow authorized users to inspect the authentication information;

3. functions to assure the integrity of, or prevent the unauthorized use of, authentication information; and
4. functions to limit the opportunity for repeated attempts to establish a false identity.

The TCSEC identification and authentication policy requires that individual users be identified, and this claimed identification must be authenticated. Users must identify themselves before beginning to perform any other actions mediated by the TCB. For many users this login interaction is the only time they must explicitly deal with the TCB, so they mistakenly think that identification and authentication constitutes access control. It is somewhat ironic that the less obtrusive the TCB is, the higher the probability of this misunderstanding.

The TCB maintains authentication information for verifying the claimed identity. Authentication information is commonly divided into three groups:

1. something the user knows,
2. something the user has, and
3. something the user is.

Mechanisms and procedures are established to authenticate claimed identity by checking one or more of these groups. Using more than one piece of information, especially from different groups, increases the probability of valid authentication.

The most common form of "something the user knows" is a password. Although there are password guidelines published by both the Defense Department [CSCD85] and the National Institute for Standards and Technology [GUID85a], password protection is often the Achilles' heel of computer security.

For example, the Department of Defense Password Management Guideline [CSCD85] provides a set of good practices related to the use of password-based user authentication, including:

1. security management responsibilities for initial password assignment, password change authorization, and user ID revalidation;
2. user responsibilities for security awareness, and changing and remembering passwords; and
3. technical guidance on internal storage of passwords, transmission, login attempt rate, auditing, password distribution, password length, and the probability of guessing a password.

But well-known weaknesses [KLEI90] in the implementation and use of passwords have caused concerned security administrators to add password filters that reject weak passwords such as:

- passwords based on the user's identity;
- passwords that exactly match a word in a dictionary (with some or all letters capitalized);
- passwords that match a reversed word in a dictionary (with some or all letters capitalized);
- passwords that are simple modifications of a dictionary word (for example, words with added plural endings or added "-ing" or "-ed");
- passwords based on the user's initials or given name;
- passwords that match a dictionary word with the numerals 0 (zero) and 1 (one) substituted for the letters o (oh) and l (el);
- passwords that are patterns from the keyboard (for example, "aaaaa" or "qwerty");
- passwords that are shorter than a specified length (for example, six characters);
- passwords that do not contain a mixture of or at least two of the following: uppercase characters, lowercase characters, numerals, and punctuation; and
- passwords that look like a state-issued license plate.

Another form of information the user knows is maintained in a database of personal information, such as mother's maiden name, favorite flavor of ice cream, and so on. An inquiry about one or more of these items as part of the login sequence serves to authenticate identity.

"Something the user has" relies on a physical possession. Unlike "something the user knows," it may be difficult for two people to possess the physical item simultaneously. In contrast, "something the user knows" can be shared with other users without the first person giving up the information. A key to a locked terminal or workstation is a form of "something the user has," but technology has provided an electronic alternative. These electronic possessions can be incorporated into small calculators, or can be self-contained smart cards. Their general operating mode is to provide a number for the user to type into the computer as an authenticator. This number may be generated completely internally by the device, or may be derived from some input which the user enters. One form of the latter is a challenge-response system wherein the computer provides a multidigit number as a challenge; the user then enters this number into the calculator and receives a response to type into the computer. The variable user input is often coupled with a secret number the user knows to bind the device to the user.

"Something the user is" relies on a biometric characteristic, such as signature, hand geometry, fingerprint, or retina pattern. There has been considerable research into biometric input devices that would convert these physical characteristics into unique authentication data, but cost, speed, and reliability problems still remain.

Whatever the source of the authentication data, it is used by the TCB to authenticate the user's identity. The TCB must protect authentication data so that it cannot be accessed by any unauthorized user. At one time it was thought sufficient to protect authentication data by encryption, but this has proven inadequate.

Once identity has been authenticated, the user identity itself is the basis for identity-based access controls such as discretionary access control (DAC). Other security attributes can be associated with the user to be used by access control policies. Security clearance is used by mandatory access control (MAC). Other attributes may be used by other policies. For example, the employer attribute is used by proprietary and no contractor policies.

Authentication is also applicable to networks. The TNI specifies that the network should ensure that a data exchange is established with the addressed peer entity (and not with an entity attempting a masquerade or a replay of a previous establishment). The network should assure that the data source is the one claimed. When this service is provided in support of a connection-oriented association, it is known as *peer entity authentication*; when it supports a connectionless association, it is known as *data origin authentication*.

Attempts to create a session under a false identity or to play back a previous legitimate session initiation sequence are typical threats for which peer entity authentication is an appropriate countermeasure.

Authentication generally follows identification, establishing the validity of the claimed identity and providing protection against fraudulent transactions. Identification, authentication, and authorization information (for example, passwords) should be protected by the network.

In addition to the authentication methods used by people, network entities may be authenticated by cryptographic means and use of the characteristics and/or possessions of the entity. These mechanisms may be incorporated into the (N)-layer peer-to-peer protocol to provide peer entity authentication.

To tie data to a specific origin, implicit or explicit identification information must be derived and associated with data. Ad hoc methods for authentication may include verification through an alternate communications channel or a user-unique cryptographic authentication.

Encryption is discussed in Essay 15. To understand encryption used for authentication, it is sufficient to know that encryption acts as a function on a string of readable text to produce a string of unreadable symbols. These strings can be treated as numeric values for various purposes, such as authentication. A cryptographic checksum is similar to other checksums used for message authentication, with the difference that the algorithm that produces the checksum takes a secret quantity, the cryptographic key, as an input. As long as the key is kept secret, it is very difficult to create a valid checksum for an altered message.

When encryption is used for authentication service, it can be provided by encipherment or signature mechanisms. In conventional secret-key (symmetric) cryptosystems, the cryptographic checksum of a message that is produced with a secret key automatically implies data origin authenticity, because only the holder of that key can produce a cryptographic checksum form of a message. The kind of authentication provided by the conventional secret-key cryptosystem can protect both sender and receiver against third-party enemies, but it *cannot* protect one against fraud committed by the other. The reason is that the receiver, knowing the encryption key, could generate the cryptographic checksum of a message and forge messages appearing to come from the sender. In a case where disputes may arise from the dishonesty of either sender or receiver, a digital signature scheme is required.

In public-key (asymmetric) cryptosystems, message secrecy and message/sender authenticity are functionally independent. To achieve authenticity, the message digest (fixed-length representation of the message) is "decrypted" with the public key of the sender to provide proof of its origin, but that does not conceal the message. If both secrecy and authenticity are required, a public-key digital signature scheme must be used.

Access control. As used in the ITSEC, access control ensures that users and processes acting on their behalf are prevented from gaining access to information or resources that they are not authorized to access or have no need to access. There are similar restrictions concerning unauthorized creation, modification, or deletion of information. This heading includes functions intended to control the flow of information between, and the use of resources by, users, processes, and objects. This includes the administration (that is, the granting and revocation) of access rights and their verification. This heading also includes:

1. any function to set up and maintain any lists or rules governing the rights to perform different types of access;
2. functions concerned with temporarily restricting access to objects that are simultaneously accessible to several users or processes and are needed to maintain the consistency and accuracy of such objects;
3. functions to ensure that upon creation, default access lists or access rules apply to objects;
4. functions to control the propagation of access rights to objects; and
5. functions to control the inference of information by the aggregation of data obtained from otherwise legitimate accesses.

Accountability. In the ITSEC, accountability ensures that relevant information is recorded about actions performed by users or processes acting on their behalf so that the consequences of these actions can later be linked to the user in question, and the user held accountable for his actions. This heading also includes functions intended to record the exercising of rights that are relevant to security, and functions related to the collection, protection, and analysis of such information. Certain functions may satisfy requirements for both accountability and auditability; such functions may be included under either heading, but should be cross-referenced.

The TCSEC accountability policy requires that actions affecting security must be traced to the responsible party. The TCB must provide the capability of associating the user's identity with all auditable actions taken by that individual. Audit information must be selectively kept and protected. A trusted system must be able to record occurrences of security-relevant events in an audit log. Audit data must be protected from modification and unauthorized destruction to permit detection and after-the-fact investigations of security violations. The TCSEC requires that audit data be protected by the TCB so that read access to it is limited to those who are authorized for audit data.

Audit. In the ITSEC, audit ensures that sufficient information is recorded about both routine and exceptional events so that later investigations can determine if security violations have actually occurred, and if so what information or other resources were compromised. This heading covers any functions intended to detect and investigate events that might represent a threat to security. This heading also includes functions related to the collection, protection, and analysis of such information. Such analysis may also include trend analysis used to attempt to detect potential violations of the security target before a violation occurs.

A TCSEC audit system should be able to record the following types of events:

1. use of identification and authentication mechanisms,
2. introduction of objects into a user's address space (for example, file open, program initiation),
3. deletion of objects, and
4. actions taken by computer operators and system administrators and/or system security officers, and other security-relevant events.

Audit should also include any override of human-readable output markings. The TCSEC requires B2 and better systems to audit identified events that may be used in the exploitation of covert storage channels.

Audit records typically include date and time of the event, user identification, type of event, and success or failure of the event. For identification

and authentication events, the origin of the request (for example, terminal ID) is also included in the audit record. For events that introduce an object into a user's address space and for object deletion events, the audit record includes the name of the object and the object's security level.

The capability to select audit events to be recorded is necessary to minimize the overhead of auditing and to allow efficient analysis. Collecting too much audit data can cause the important information to be hidden like the needle in the haystack. It can also cause the system to crash if there is no suitable place to store the audit data. Therefore, the security administrator must be able to selectively audit the actions of any one or more users based on individual identity and/or object security level. Flexibility in selecting audited events can be very helpful.

The TCSEC assigns the audit system of a B3 or better system the responsibility to monitor the occurrence or accumulation of security auditable events that may indicate an imminent violation of security policy and to immediately notify the security administrator when thresholds are exceeded. If the occurrence or accumulation of these security-relevant events continues, the system is to take the least disruptive action to terminate the event.

The TDI notes that the emphasis of the audit criterion is to provide individual accountability for actions by users. This goal is not the same as that for a backup and recovery log. There is no requirement in the TDI to integrate the audit log with the backup and recovery log, although such an integrated log is not prohibited. At the designer's discretion, there may be a selectable capability to reduce the number of audit records generated in response to queries that involve many access control decisions.

Object reuse. In the ITSEC, object reuse ensures that resources such as main memory and disk storage can be reused while observing security. This heading also covers any functions intended to control the reuse of data objects. This heading also includes functions to initialize or clear unallocated or reallocated data objects, to initialize or clear reusable media such as magnetic disks and tapes, or to clear output devices such as display screens when not in use.

According to the TCSEC object reuse requirement, introduced at C2 and unchanged thereafter:

> authorizations to the information contained within a storage object shall be revoked prior to initial assignment, allocation or reallocation to a subject from the TCB's pool of unused storage objects. No information, including encrypted representations of information, produced by a prior subject's actions is to be available to any subject that obtains access to an object that has been released back to the system.

The reference to encrypted information was added when the TCSEC was reissued as a DoD standard in 1985; it was not present in the 1983 version.

Accuracy. In the ITSEC, accuracy ensures that specific relationships between different pieces of data are maintained correctly and that data is passed between processes without alteration. This heading covers any function intended to ensure that data has not been modified in an unauthorized manner. This heading also includes functions to determine, establish, and maintain the accuracy of the relationships between related data; and functions to ensure that when data is passed between processes, users, and objects, it is possible to detect or prevent loss, addition, or alteration, and that it is not possible to change the claimed or actual source and destination of the data transfer.

Reliability of service. In the ITSEC, reliability of service ensures that time-critical tasks are performed when they are necessary, and not earlier or later; that non-time-critical tasks cannot be made time-critical; that access to resources is possible when it is needed; and that resources are not requested or retained unnecessarily. This heading covers any function intended to ensure that resources are accessible and usable on demand by an authorized user or process and to prevent or limit interference with time-critical operations. This heading also includes error detection and error recovery functions intended to restrict the impact of errors on operation and so minimize disruption or loss of service, and any scheduling functions that ensure response to external events and produce outputs within specified deadlines.

Data exchange. In the ITSEC, data exchange ensures the security of data during transmission over communications channels. The ITSEC recommends that the following subheadings from the OSI Security Architecture [ISO89] (also used in the TNI [NCSC87a]) be used to group functions:

- authentication,
- access control,
- data confidentiality
- data integrity, and
- nonrepudiation.

Sterne et al. [STER91b] note that source authentication or nonrepudiation requirements appear to be widespread. Certain functions may satisfy requirements for both computer and communications security and so be relevant to other headings; such functions should be cross-referenced.

Assurance. In the TCSEC, assurance is concerned with guaranteeing or providing confidence that the security policy has been implemented correctly and that the protection-relevant elements of the system do indeed accurately enforce the intent of that policy. By extension, assurance must include a guarantee that the trusted portion of the system works only as intended. To accomplish these objectives, the TCSEC identifies two types of assurance: life-cycle assurance and operational assurance.

Life-cycle assurance refers to steps taken by an organization to ensure that the system is designed, developed, and maintained using appropriate controls and standards. Trusted computer systems depend on the hardware, firmware, and software to protect the information with which they are entrusted. It follows that the hardware, firmware, and software must be protected against unauthorized changes that could cause protection mechanisms to malfunction or be bypassed completely. For this reason, trusted computer systems must be carefully evaluated and tested during the design and development phases and reevaluated whenever changes are made that could affect the integrity of the protection mechanisms. Only in this way can confidence be provided that the hardware, firmware, and software interpretation of the security policy is maintained accurately and without distortion.

While life-cycle assurance is concerned with procedures for managing system design, development, and maintenance, operational assurance focuses on features and system architecture used to ensure that the security policy is uncircumventably enforced during system operation.

The TCSEC assurance policy requires the computer system to contain hardware, firmware, and software mechanisms that can be independently evaluated to provide sufficient assurance that the system enforces the major policy objectives and the supporting policies. Assurance refers to the basis for believing that the functionality will be achieved, including tamper resistance, verifiability, and resistance to circumvention or bypass. Assurance is generally based on analysis involving theory, testing, software engineering, and validation and verification. In the TCSEC, the assurance requirements parallel the evaluation classes, progressing from informal to formal, where formal implies mathematical techniques, such as those discussed in Essay 8. The basis for trusting system mechanisms in their operational setting must be clearly documented such that it is possible to independently examine evidence to evaluate the mechanisms' sufficiency.

The TCSEC provides requirements for operational assurance, including system architecture, system integrity (correct operation), covert channel analysis (starting at B2), trusted facility management, and trusted recovery. Security testing and design specification and verification are included under life-cycle assurance.

The ITSEC requirements for assurance of effectiveness are based on the proposed use of the TOE as described in its security target. The ITSEC

requirements for assurance of correctness are expressed in seven hierarchical levels (see Essay 12).

Continuous protection. The TCSEC continuous protection policy requires that the trusted mechanisms that enforce these basic requirements must be continuously protected against tampering and/or unauthorized changes. No computer system can be considered truly secure if basic hardware, firmware, and software mechanisms that enforce security policy are themselves subject to unauthorized modification or subversion. The continuous protection requirement has direct implications throughout the computer system's life cycle.

The TCSEC introduces configuration management requirements at B2 and trusted distribution at A1. We suggest that both of these requirements should be effective at all levels, as they are in the ITSEC.

Covert channels. The TCSEC defines a covert channel as any communication channel that can be exploited by a process to transfer information in a manner that violates the system's security policy. The TCSEC also distinguishes two types of covert channels: storage channels and timing channels.

Current thinking refers to a covert channel as the use of some artifact not designed for communication to transfer information in a manner that violates the system's security policy. The misuse of a communications channel is simply a security violation.

Covert storage channels include all vehicles that would allow the direct or indirect writing of a storage location by one process and the direct or indirect reading of it by another. Covert timing channels include all vehicles that would allow one process to signal information to another process by modulating its own use of system resources in such a way that the change in response time observed by the second process would provide information.

Current thinking is that every covert channel involves elements of storage and timing. Information or state must exist long enough to be sensed in a storage channel. Timing involves the frequency of one event or state relative to another event or state. It is not necessary for one of these states to be related to the passage of time.

The TCSEC asserts that covert channels with low bandwidths represent a lower threat than those with high bandwidths. However, for many types of covert channels, techniques used to reduce the bandwidth below a certain rate (which depends on the specific channel mechanism and the system architecture) also have the effect of degrading the performance provided to legitimate system users. Hence, a trade-off between system performance and covert channel bandwidth must be made. Because of the threat of compromise that would be present in any multileve computer

system containing classified or sensitive information, such systems should not contain covert channels with high bandwidths.

The TCSEC covert channel guideline is intended to provide system developers with an idea of just how high a "high" covert channel bandwidth is. It considers a covert channel bandwidth that exceeds a rate of 100 bits per second to be "high." It also considers maximum bandwidths of less than 1 bit per second acceptable in most application environments. Finally, covert channels with bandwidths that exceed a rate of 1 bit in 10 seconds are to be audited. The rationale for these figures does not withstand close examination.

The TCSEC covert channel requirement begins at the B2 level, requiring a thorough search for covert storage channels and a determination (either by actual measurement or by engineering estimation) of the maximum bandwidth of each identified channel. At B3, timing channels are added, and at A1 it is required that formal methods be used in the analysis.