7.0 TRANSFER SYSTEM

This section discusses the basic goal of the transfer system security architecture and then the means to achieve that goal. Section 7.1 discusses the basic notion of distributed security contexts and the primary function that supports them, the security association. Section 7.2 describes several supporting functions and tools needed to implement distributed security contexts and security associations. Section 7.3 discusses the relationship of the transfer system security architecture to some specific security-related topics.

In section 4, the transfer system was identified as the LCSs, CNs, and the communications protocols in end systems and relay systems. Security services allocated to the transfer system provide the basis for the protection of information in transfer. Availability is the only security service allocated to CNs and LCSs. Additional security services may be provided by LCSs, but they are only applicable to local communications.

The portion of the transfer system in end systems and relay systems consists of open system networking applications and communication protocols (including some security protocols). These applications and protocols are executed in the same security context as other user applications for a user operating in a particular information domain. Except for transfer system functions that are among the security-critical functions (e.g., network interface device drivers, cryptographic functions), transfer system software does not need to be trusted. The transfer system must be managed, so the SMAP and SMIB of section 6 are extended to account for transfer system functions.

The primary goal of the transfer system security architecture is to provide protection of information in transfer to support information sharing and distributed processing within the security architectures of the other DGSA elements and the fundamental concepts. The basic approach to achieving this goal is to enable security contexts in different end systems or relay systems (that support the same information domain) to communicate as if they were in the same end system or relay system. The transfer system security architecture must fit within the end system and relay system architecture of section 5 and the security management architecture of section 6, and it must extend the support of fundamental DGSA concepts to communications, especially information domains, strict isolation, multidomain information objects, and absolute protection. The remainder of section 7 addresses various concepts and functions needed to achieving the transfer system goal.

7.1 DISTRIBUTED SECURITY CONTEXTS

The generic transfer system security architecture seeks to create structures in which applications in security contexts in different end systems or relay systems (that support the same information domain) communicate with the same assurance as if they were in the same end system or relay system. Such structures are referred to as *distributed security contexts*. There are two basic classes of communication that must be considered, *interactive* and *staged delivery*. Staged delivery refers to communications in which the information being transferred is sent from the originating end system application to a relay system application, in its entirety, and then is sent from the relay system application to the destination end system application. (There may be several relay system applications involved before the information is finally delivered to the destination end system application.) The most common example of staged delivery is electronic mail. Interactive communications include all non-staged delivery applications. The means used to create distributed security contexts are different for interactive and staged delivery communications and will be discussed separately.

7.1.1 Distributed Security Contexts for Interactive Communications

An *interactive distributed security context* is formed when two security contexts in different end systems are joined securely using a set of mechanisms that is referred to as a security association. A *security association* is the totality of communication and security mechanisms and functions (e.g., communications protocols, security protocols, doctrinal mechanisms, security-critical mechanisms and functions) that securely binds together two security contexts in different end systems or relay systems supporting the same information domain.[1] A security association extends the protections required by an information domain security policy within an end system to information in transfer between two end systems and it maintains strict isolation from other information domains. A security association can be considered an extension or expansion of an OSI application association. OSI application layer entities in different end systems employ application associations to communicate. An application association is composed of appropriate application layer functions and protocols plus all of the underlying communications functions and protocols at other layers. A security association is an application association that includes additional support from security functions and mechanisms.

[17-1] Note that the DGSA meanings of security association, agreed set of security rules, security association identifier, and security association management protocol are different and more general than their meanings in existing protocol specifications.

The security management information for a security association is contained in a SMIB data structure called the *agreed set of security rule*s (ASSR). The ASSR includes all the security-relevant attributes required to establish and maintain a security association, such as the information domain label and secure communication attributes (e.g., cryptographic algorithm identifiers and keys). Each end system supporting a security association chooses a local *security association identifier* (SAID). The pair of SAIDs uniquely identifies the particular security association and links it to the ASSR for that security association. Thus, any security function supporting the security association obtains necessary information from the ASSR.

7.1.1.1 Security Association Establishment

A security association is established using a SAMP. The originating end system SAMP implementation, invoked by a SMAP, creates an OSI application association with the destination end system peer SAMP implementation.[2] The SAMP implementations (hereafter referred to as SAMP machines) cooperate to establish a security association through a set of SAMP exchanges. The SAMP exchanges include three basic functions. Initially, the originating SAMP machine makes known its secure communications capabilities to its peer SAMP machine in the form of one or more object identifiers (i.e., an Abstract Syntax Notation.1 *Sequence of Object Identifiers* (ISO, 1988d)). An entire set of capabilities may be referenced by a single object identifier that corresponds to a specific registered ASSR. The destination SAMP machine subsequently generates the appropriate response (i.e., a positive acknowledgment to continue the SAMP exchanges or an error response). This first paired SAMP exchange is always conducted in the "clear" (i.e., it is not cryptographically protected). A positive response does not indicate that the security association has been established nor that the destination intends to accept the security association, but only that the subsequent exchanges can proceed. (Some of the considerations involved in deciding whether to allow a security association to be established are considered in the next subsection.)

The primary purpose of the second SAMP exchange is to establish the keys needed for cryptographic security mechanisms. Generally, security associations will rely upon cryptographic mechanisms so that sufficient strength and assurance for the security services is provided. One or more

---

[27-2] Like the end system security context that is dedicated to user login activities, there will be an end system security context dedicated to processing incoming SAMP exchanges since the information domain to be supported will not be known reliably until the security association has been established.

mechanisms may be chosen to support a particular information domain security policy. Either asymmetric or symmetric key generation, coordination, and exchange techniques may be employed. The SAMP must be general enough to support any standard key management technique. Depending upon the key management technique selected, various processes or key management protocols may be executed to form a traffic key. Examples include the Secure Data Network System (SDNS) Key Management Protocol for exchange of certificates and associated user keying material, and the X9.17 (ANSI, 1985) and Rivest-Shamir-Adelman (RSA, 1978) key management protocols and associated techniques developed to support commercial cryptography. This second SAMP exchange may or may not be conducted in the clear, depending on the key management technique employed. Other security functions may be performed in conjunction with the second SAMP exchange, such as access control checks based on information conveyed in a certificate delivered by the SAMP. If multiple cryptographic algorithms or keys are required to support a security association, it may be possible to convey the required information in one exchange or it might be necessary to repeat the second SAMP exchange several times.

The third SAMP exchange employs the encryption algorithms and keys established in the second exchange to test the state (liveness) of the security association. This exchange also provides peer entity authentication between the cooperating SAMP machines, reliably sends any remaining security attributes needed to operate the security association which are not already in the ASSR (e.g., the specific security services to be supported by the security protocols selected in the second exchange), and may validate the information used in the earlier SAMP exchanges. When high assurance is required that a security association has been established between two end systems that are accredited for support of a particular information domain, the peer entity authentication must be based on cryptographic techniques. The peer entity authentication is not assured until the algorithm and keys have been tested between the two cooperating SAMP machines.

When the third SAMP exchange has been successfully completed, the security association is established. The destination end system then creates a security context for the appropriate information domain and initiates the execution of the applications necessary to communicate with the originating end system applications, including the communications protocols and the SMAP for the information domain. The security contexts in the originating and destination end systems are now joined by the security association to form the interactive distributed security context.

7.1.1.2 Additional Aspects of Interactive Distributed Security Contexts

The decision to allow establishment of a security association may require several related functions to be performed such as the exchange and processing of security attributes of the user (e.g., authenticated identity, access privileges). These attributes might be contained in a security certificate such as that defined in the X.509 Directory Services Authentication Framework (CCITT, 1992). The information contained in an X.509 certificate may be signed by any number of hierarchically related certificate-issuing authorities, down to an information domain-specific certificate-issuing authority if that level of granularity is required. This signature verification adds greater assurance to the credibility of the information contained in the certificate.

Multiple security protocols may be included in a single security association to provide a combination of security services. For example, a network layer protocol might provide continuous end system origin authentication and data integrity, while a presentation layer protocol might provide selective field data confidentiality. Some lower layer security protocols can multiplex several security associations between the same end systems. The security associations share the same cryptographic algorithm and keys. This arrangement may be appropriate for interactive distributed security contexts that support the same information domain, but it is unlikely to be acceptable for different information domains because of strict isolation requirements.

In some instances, an interactive distributed security context will be formed between end systems that employ no security protocols and may not even require an authenticated user identity. Such instances include access to public information utilities (e.g., a news wire service feed) or completely unprotected end systems. In these instances, an end system that supports other information domains, then the end system strict isolation mechanisms will be entirely responsible for maintaining the isolation of unprotected information domains from other information domains.

Some communications between end systems involve information that is not ordinarily stored in an end system, for example, real-time voice and video applications. In these cases, users must monitor and enforce the accuracy of the security context and association established for the distributed security context. That is, humans must ensure that information exchanged belongs to the information domain represented by the distributed security

context as is currently done when using Secure Telephone Unit-IIIs for secure voice or data communications.

Figure 7-1 illustrates the relationships among the primary components that create a security association.

Figure 7-1. Component Relationships for Creating a Security Association

Appendix B provides a detailed example of the establishment of an interactive distributed security context.

7.1.2 Staged Delivery Distributed Security Contexts

A staged delivery distributed security context is transferred from the originating end system to the destination end system. This is accomplished by an application in the originating end system cryptographically wrapping the information to be transferred in a form that allows the destination end system to reconstitute the security context in which the information was wrapped. The wrapped information is transferred (in stages) from the originating end system to the destination end system. Ideally, the wrapping process should provide all security protection of the information while in transfer. No security services (other than availability) should be expected of the application relay systems involved in the staged delivery because they might be provided by common carrier providers, as is the case for CNs. If the wrapping process cannot provide all the necessary security protection, the application relay systems will have to be implemented to support the DGSA and interactive distributed security contexts between end systems and relay systems will have to be used to ensure the secure staged transfer of information.

There is an existing specification for a secure electronic mail service that satisfies the requirements for staged relay distributed security contexts. This document is the SDNS Message Security Protocol (MSP) specification (NSA, 1992). MSP can provide authentication, access control, message confidentiality and integrity, and non-repudiation security services. MSP allows delivery of the same message to multiple recipients supported by several end systems without creating multiple copies of the message in the originating end system. Multiple messages created in different security contexts can be combined in a single MSP transfer. The wrapping of the

messages takes place in the originating end system in an MSP user agent. The wrapped message is submitted to the message transfer system, which consists of a group of untrusted cooperating message transfer agents. The message is delivered to one or more destination MSP user agents, which unwrap the message. For details of how secure staged delivery can be achieved, the MSP specification should be examined. MSP will be the basis for secure messaging in DoD as Phase II of the Defense Message System is implemented and deployed.

7.1.3 Other Aspects of Distributed Security Contexts

7.1.3.1 Multidomain Object Transfer

Section 4.3.4 defined and discussed multidomain objects and noted that their purpose is to display or print related information objects from several information domains in an ordered format. Section 5.2.2 discussed some high-level implementation aspects of multidomain objects. The transfer of a multidomain object between end systems requires that both the component information objects and the description of their relationships be transferred. Since a distributed security context supports transfer of information within a single information domain, one distributed security context is used for each of the component information domains. If the description of the component relationships is contained in an information object in a separate information domain, another distributed security context is required for its transfer. An application similar to those used to display or print multidomain objects is needed to coordinate the transfer of the component information objects.

7.1.3.2 Distributed Security Context Single Information Domain Restriction

The definition of a distributed security context restricts it to joining end system or relay system security contexts that support the same information domain. In principle, this restriction could be removed, however, there are practical reasons for retaining it. One of the principle functions of a distributed security context is to maintain strict isolation of information in transfer. Within an end system, the separation kernel (or other strict isolation mechanism) controls all interactions between security contexts. As noted earlier, it is expected that cryptographic mechanisms will be the usual means to maintain strict isolation for information in transfer. The use of such cryptographic mechanisms requires shared use of keys and other supporting information between security contexts in the communicating end systems. If those security contexts support different information domains,

sharing of the keying information is difficult. There will also be additional complexity introduced into many communications and security protocols that will result in trusted implementation of additional functions. The restriction that distributed security contexts support transfers within a single information domain is intended to simplify implementations that support the DGSA concepts.

7.2 TRANSFER SYSTEM SUPPORT

This section describes several elements needed to support the basic transfer system activities.

7.2.1 Security Management Application Process

In addition to the SMAP functions described in section 6, it also controls the establishment and termination of all security associations and distributed security contexts, and all transfer system security services and mechanisms. Additional transfer system-related SMAP functions and interfaces support the following activities:

- End system communications applications requests (through the extended GSS-API)
- Additional SMIB information object use and maintenance (e.g., to access information for remote security administration maintenance, security protocol and algorithm operation, certificate processing)
- Maintenance and retrieval of security information from the X.500 Directory using the directory access protocol
- MSP processing for staged delivery secure messaging for both transmission and receipt
- SAMP operations for establishment of interactive distributed security contexts, including security protocol operation, termination, and recovery, plus maintenance of the SAID and ASSR structure for each security association established
- Cryptographic and key management functions for security service and security protocol operation
- General-purpose management protocol operation (e.g., CMIP) to

accomplish secure exchange of security information between distributed SMAPs or network management information requested by network management systems

7.2.2 Security Management Information Base

Additional information is required in the end system SMIB and the information domain SMIBs to support transfer system operations.

Additional information domain SMIB information items include:

- X.509 certificates to carry appropriate security information, such as SDNS key management certificates
- User access control information for distributed operations
- Traffic and message keys
- Accumulated audit data, including records of distributed security context utilization

Additional end system SMIB information items include:

- Key management, encipherment, integrity, and signature algorithm identifiers, and security protocol objects
- End system access control information for distributed operations
- Encryption algorithm initialization information
- Security association configuration information (e.g., ASSRs, SAID tables)
- Compromise action information (e.g., revoked certificates lists)
- Contingency plan parameters (e.g., auto-purge and security policy replacement actions under emergency conditions)

Some SMIB items may be held in Directory Service Agents for ease of access by many users. Such items might include key management information (i.e., SDNS certificates and user keying material) used by MSP implementations. SMIB information stored in X.500 Directories must be integrity protected.

7.2.3 Security Protocols

Several security protocols, either existing or in development, are candidates for use in end systems implementing the DGSA. Others may be added over time.

The Transport Layer Security Protocol (TLSP) is an ISO standard (ISO, 1992c) and the Network Layer Security Protocol (NLSP) is a Draft ISO standard (ISO, 1992a). The IEEE 802.10 SILS Secure Data Exchange (SDE) protocol standard (IEEE, 1992) is appropriate for LCS security services (beyond availability) when needed. MSP is the DoD standard for electronic messaging.

No current SAMP meets DGSA needs, but an ISO project under way that is developing such a protocol. The initial version is based on IEEE 802.10 SILS Part 3. The GULS SESEP will carry SAMP exchanges. SESEP is expected to be the carrier for all new application and presentation layer security protocols, so it will be included, at least implicitly, among security protocols for DGSA implementation. If the SAMP standard that emerges from ISO contains the functionality as currently planned, it will be a suitable protocol for DGSA implementation.

There are many existing and planned physical layer encryption and transmission security devices (which are necessarily communications technology-specific). When traffic flow security services are required, these devices may be used (see section 7.3.1).

Table 7-1 identifies the security services supported (actual or planned) by each of the security protocols discussed.

Table 7-1. DGSA Security Protocols and Security Services

7.2.4 Cryptographic Support

The creation of distributed security contexts, which provide communications security services and strict isolation adequate for sensitive information, is usually dependent on cryptographic mechanisms. Thus, the availability of low-cost cryptographic devices is a critical element of the DGSA. These cryptographic devices must be sufficiently flexible to support

requirements of different information domains in the same end system.

This flexibility will be achieved if the devices accommodate multiple cryptographic algorithms and multiple key management schemes, including public key encryption schemes and various key distribution center schemes Otherwise, a multiplicity of cryptographic devices will be needed, resulting in increased costs. To manage these devices, there must be a registry of cryptographic algorithms and key management schemes so that the specific choices can be negotiated for a particular security association.

Currently available cryptographic and key management devices do not meet these flexibility criteria. Very large scale integration (VLSI) chip technology may now have reached a sufficient density to achieve a cost-effective single-chip design which can support multiple algorithms and a variety of key management schemes, along with a cache memory capable of handling reasonable quantities of key material. The cryptographic devices must be capable of a minimum throughput rate of 10 megabits per second to be useful with high-performance workstations. Isolation techniques must accommodate concurrent algorithm execution and Red/Black separation (in software, hardware, or both). The DGSA is achievable only if this kind of cost-effective technology is available. In addition to creating low-cost devices, COMSEC custodial functions must be minimized through the use of electronic key management technology.

7.2.5 Distributed Management Systems

Distributed management of information systems both supports the transfer system and relies upon the transfer system for its operation. Management systems will rely upon the same transfer system security structures (distributed security contexts, security associations, and security protocols) as any other application.

When distributed information systems become very large, their management becomes very complex. To make the complexity manageable, hierarchical management approaches are often adopted. It then becomes necessary to coordinate the levels of delegated management authority. The coordination is achieved by the way management information is organized and through the control of that information as required by security policies. Hierarchical management relationships are not reflected in the way management applications communicate with one another. That is, management protocols are peer oriented, not hierarchically related. When the term "hierarchical management system" is used, it must be understood that a set of information relationships is being described, not a

communications structure. This means that the hierarchical aspect of management is a human, organizational function. The organizations and administrators that manage information systems may be organized hierarchically. Management information may reflect that organization, but the end systems in which management applications are implemented only communicate as peers.

Management systems are composed of management applications implemented in end systems. Some management applications must coexist with other applications in end systems, but for logistical reasons it may be desirable to dedicate some end systems to management system activities. Management systems can be grouped into three categories based on the particular type of management function being performed. While these categories are logically separate, they often support one another. The three categories are network management, security management, and information management.

Traditional network management systems are network control centers that monitor and configure network components, perform fault isolation functions, and collect accounting and performance information. Security management systems typically provide information to support security services and mechanisms in end systems and relay systems. Most often the support is for cryptographic mechanisms. Example systems are the DoD EKMS, BLACKER Access Control Centers and Key Distribution Centers, and the CANEWARE security management components. Information management systems include are X.500 Directory systems, the Internet Domain Name Service and the Network Information Center.

Although these three logical categories of management systems could be implemented in end systems dedicated to the functions of only one of them, as a practical matter, some of the functions can be expected to be supported on common end systems. However, each logical category may require unique technical administrative expertise. In some cases, it will not be prudent to assign multiple administrative functions to individuals because too much control might be entrusted to them.

7.3 DGSA TRANSFER SYSTEM ISSUES

Two aspects of the DGSA transfer system deserve further discussion. One is the use of traffic flow security mechanisms, and the other is potential limitations on distributed processing functions.

7.3.1 Traffic Flow Security in Open System Communication

Environments

The DGSA open system and common carrier communications requirements result in the allocation of security of information in transfer to LSEs, particularly end system security support for the transfer system. The use of common carrier CNs precludes the use of full traffic flow security (TFS) mechanisms. Full TFS mechanisms operate at the physical protocol layer. Only if communications facilities are owned or controlled by user organizations can full TFS be applied.

The clear cost disadvantages of owning and operating private CNs means that there must be a careful examination of threats and vulnerabilities to determine whether full TFS is required. Unless it is necessary to subject all communications to full TFS, the DGSA requirements for open system and common carrier communications can be met with multiple communications connectivity. The strict isolation mechanisms required in end systems make it possible to support multiple communications connections among the information domains supported. Partial TFS mechanisms should be considered as alternatives to full TFS when judged to be appropriate to the known threats and vulnerabilities.

7.3.2 Limitations on Distributed Processing

Some communications technologies are inherently of a broadcast nature (e.g., radio, broadband LANs). Broadcast technologies make it possible to communicate with any end system that has access to the medium without the need to explicitly address information to specific end systems. Broadcast-like effects, called multicasts, can be achieved over non-broadcast communications systems through various methods that address and send information to (possibly large) groups of recipient end systems or users (e.g., groups of electronic mail recipients).

Certain limitations are encountered if cryptographic mechanisms are used to support security services for broadcast (and some multicast) communications. There are two basic choices. First, for true broadcasts, a single encryption key must be shared among all recipients. The use of a shared key among large numbers of recipients not only increases the likelihood that the key will be compromised, but the distribution and use of one or more shared keys is difficult to coordinate. (The same considerations apply to multicast services that depend on broadcast media.)

Second, for multicasts that are addressed to the group of recipients, a single key can be used for the security mechanism applied to the

information to be sent and that key can be replicated and protected with a cryptographic mechanism using a different key known to each recipient (e.g., MSP confidentiality service for multiple recipients).

Thus, if it is desired to broadcast information to all the members of an information domain, group multicasts are likely to be sufficient for most purposes since the member addresses are known. The only real limitation on broadcast communications is that the inherent broadcast capabilities of some media cannot be used.