# Section 7: Security in POSIX Open System Environment Services

One of the objective of the framework is to provide guidance to the specifiers of component and profile interfaces as to what security requirements and obligations need to be met. This section offers a first pass in providing this guidance.

## 7.1 General Approach

The general approach to the specification and design of security mechanisms and measures within a functional area is summarised here. It is based on the principles, objectives and strategies introduced in earlier chapters. The specification of security measures and mechanisms within functional areas needs to be conducted within the general constraints and objectives of the framework.

• Adopt the concepts of 1 View the functional area as one or more security domains with dependencies and interactions amongst themselves and the security domains representing other functional areas.

• Perform a threat and vulnerability analysis for the functional area using the list of threats and vulnerabilities outlined in A.

• Identify security measures and map these the services within the functional area, and establish which services are security enforcing with respect to each security strategy and measure.

• Define interfaces to the security relevant services which support security aware and unaware callers as defined in 1.

## 7.2 General Functional Requirements

The following sections identify security requirements which need to be considered by the specifiers of major functional areas.

The security requirements common to most functional areas are:

Copyright © 1994 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

**Access Control**

Provision of service elements which meet object reuse requirements by purging redundant service objects of all information.

**Availability**

General compliance with the principles of failure secure and secure recovery.

**Accounting and Audit**

Provision of audit event detection functions

### 7.2.1 Language Services

To the extent that a programming language provides access to services within other functional areas, it will also utilize the security features incorporated in those services. There is no need to levy additional requirements on language services.

**Miscellaneous**

Certain features of a programming language allow programmers to write more secure programs. For example, data abstraction can be used to hide details of security relevant processes. As another example, the ability to write self-documenting code makes it less likely that malicious code could be hidden in a program.

## 7.3 System Services

The main security requirements to be met by the operating system level, have already been introduced in the preceding descriptions of platform domains. They are summarised below:

• Provide low level elements from which the higher level services can be substantiated and on which the protected and separated service domains can be formed.

• Mediate access to all operating system elements according to any one or combination of the appropriate and required access control strategies.

Based on these fundamental security services, the operating system may also be required to provide specific service elements to perform or support:

**Authentication**

Secure storage of and access to authentication credentials.

Support for a trusted path between user interface devices controlled by the operating system and higher level authentication services.

Copyright © 1994 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

### Access Control

In addition to the basic operating system access control services the following may also be required of the operating system:

Secure storage of and access to cryptovariables (keys) and cryptoservices.

Secure storage of and access to the security context information (authentication and authorisation information) associated with each chain of execution passing through the environment controlled by the platform domain.

Secure storage and access to security-relevant object information on behalf of itself or higher-level services.

Support for the combination of authorisations in order to perform n-person operations.

### Accounting and Audit

Provision of the service functional elements for: security audit event detection for operating system functions, recording, alarm reporting, and the setting of audit and alarm selection criteria for operating system events.

Secure storage and management of audit trails.

### Administration

Provision of the low level service elements used to maintain security-relevant operating system management information. For example, access control lists for system, user interface, storage, network, and application software services.

### Miscellaneous

Translation services which map security-relevant information from one representational or semantic form to another.

## 7.4 Communication Services

This mapping of security requirements and obligations on communications services does not, as yet, take the mapping down to specific communication service layers.

### Authentication

Provision of service elements and protocols to support authentication between peer entities.

Copyright © 1994 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

**Access Control**

Provision of confidentiality and integrity protection services to higher level service elements which counter the transmission threats and vulnerabilities.

Secure storage of and access to cryptovariables (keys) and cryptoservices, when not provided by some specific service.

**Accounting and Audit**

Provision of the functional elements responsible for countering communication repudiation threats.

**Administration**

Provision of service elements used to maintain security-relevant communication service management information.

**Miscellaneous**

Translation services which map security relevant communication information from one representational or semantic form to another.

## 7.5 Distributed Processing Support Services

**Authentication**

Provision of service elements and protocols to support authentication between instances of distributed processing support services.

Provision of authentication services which securely pass authentication credentials between the elements of higher-level services.

**Access Control**

Provision of secure association services which securely pass the security context between elements of higher-level services.

Provision of confidentiality and integrity protection service elements which counter the transmission threats and vulnerabilities.

Secure storage of and access to cryptovariables (keys) and cryptoservices, when not provided by some specific service.

**Administration**

Provision of service elements which are used to maintain security-relevant distributed processing support-service management information.

Copyright © 1994 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

**Miscellaneous**

Translation services which map security relevant communication information from one representational or semantic form to another.

## 7.6 Database Services

**Authentication**

**Access Control**

**Accounting and Audit**

**Administration**

## 7.7 Data Interchange Serices

Data description protocols and data format protocols are defined at a level that may not require security functions. Instead, security is wrapped around data encoded in those formats at the time one wants to exchange it. However, particular data formats may want to include authentication information.

## 7.8 Transaction Processing

The security requirements for transaction processing are subsumed by the database, operating system, or communication services under which the transactions occur.

## 7.9 User Command Interface

(See ISO/IEC 9945-2 and 1003.6 amendments) Utilities for user and shell-application control of access control lists and other aspects of the security interface.

Copyright © 1994 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

## 7.10 Character Based User Interface

### Authentication

Provision of service elements and protocols to exchange securely authentication credentials between the user and the system authentication services.

Support for trusted path mechanisms.

## 7.11 Graphical Window System

### Authentication

Provision of service elements and protocols to exchange securely authentication credentials between the user and the system authentication services.

### Access Control

If the window system allows simultaneous display of windows in different security domains, then the window system must maintain separation between these domains. For example, cut and paste functions must be aware of what domains they are moving data between, and must disallow such movement as security policy requires. This may in turn require management functions to specify the policy. Another example is visual labeling of windows, indicating the security domain displayed, as an aid to the user in maintaining separation.

## 7.12 Graphics

Not applicable.

## 7.13 Application Software Development

Configuration management must ensure that malicious code does not get incorporated into developed programs. Any tools that improve the quality of code developed are security-enhancing, since they reduce the chance of code having security vulnerabilities.

Copyright © 1994 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.