FIPS PUB 188

Federal Information
Processing Standards Publication

1994 September 6

U.S. DEPARTMENT OF COMMERCE / National Institute of Standards and Technology

Standard Security Label for Information Transfer

CATEGORY: COMPUTER SECURITY
SUBCATEGORY: SECURITY LABELS

U.S. DEPARTMENT OF COMMERCE, Ronald H. Brown, *Secretary*
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY,
Arati Prabhakar, *Director*

Foreword

The Federal Information Processing Standards Publication Series of the National Institute of Standards and Technology (NIST) is the official publication relating to standards and guidelines adopted and promulgated under the provisions of Section 111(d) of the Federal Property and Administrative Services Act of 1949 as amended by the Computer Security Act of 1987, Public Law 100-235. These mandates have given the Secretary of Commerce and NIST important responsibilities for improving the utilization and management of computer and related telecommunications systems in the Federal Government. The NIST, through the Computer Systems Laboratory, provides leadership, technical guidance, and coordination of Government efforts in the development of standards and guidelines in these areas.

Comments concerning Federal Information Processing Standards Publications are welcomed and should be addressed to the Director, Computer Systems Laboratory, National Institute of Standards and Technology, Gaithersburg, MD, 20899.

James H. Burrows, Director
Computer Systems Laboratory

Abstract

Information Transfer security labels convey information used by protocol entities to determine how to handle data communicated between open systems. Information on a security label can be used to control access, specify protective measures, and determine handling restrictions required by a communications security policy. This standard defines a security label syntax for information exchanged over data networks and provides encodings of that syntax for use at the Application and Network Layers. The syntactic constructs defined in this standard are intended to be used along with semantics provided by the authority establishing the security policy for the protection of the information exchanged. A separate NIST document, referenced in an informative appendix, defines a Computer Security Objects Register (CSOR) that serves as repository for label semantics.

Key words: Application Layer security, computer communications security, Computer Security Objects Register, Federal Information Processing Standard, Information Transfer security labels, Network Layer security, security labels, security protocols.

Federal Information
Processing Standard Publication 188

1994 September 6

ANNOUNCING A

Standard Security Label for Information Transfer

Federal Information Processing Standards Publications (FIPS PUBS) are issued by the National Institute of Standards and Technology (NIST) after approval by the Secretary of Commerce pursuant to Section 111(d) of the Federal Property and Administrative Services Act of 1949 as amended by the Computer Security Act of 1987, Public Law 100-235.

Name of Standard: Standard Security Label for Information Transfer.

Category of Standard: Computer Security, Security Labels

Explanation: Security labels convey information used by protocol entities to determine how to handle data communicated between open systems. Information on a security label can be used to control access, specify protective measures, and determine additional handling restrictions required by a communications security policy.

This standard defines a security label syntax for information exchanged over data networks and provides label encodings for use at the Application and Network Layers. The syntactic constructs defined in this standard are intended to be used along with semantics provided by the authority establishing the security policy for the protection of the information exchanged. A separate NIST document, referenced in an informative appendix, defines a Computer Security Objects Register (CSOR) that serves as repository for label semantics. The CSOR assigns a unique identifier to each set of interpretation and handling rules, this enables the communicating parties to agree on the semantics for the interpretation of the labels. The separation of the label syntax from its semantics enables a few basic label structures to support multiple security policies.

The label presented here defines security tags that may be combined into tag sets to carry security-related information. Five basic security tag types allow security information to be represented as bit maps, attribute enumerations, attribute range selections, hierarchical security levels, or as user-defined data. Because of inherent differences in layer functionality, the security label defined in this document is expressed both as an abstract label syntax specification for the OSI Application Layer and an encoding optimized for use at the Network Layer.

Approving Authority: Secretary of Commerce.

Maintenance Agency: U.S. Department of Commerce, National Institute of Standards and Technology, Computer Systems Laboratory

Cross Index:

a. Federal Information Resources Management Regulations, subpart 201-20.303, Standards, and subpart 201-39.1002, Federal Standards.

b. General Procedures for Registering Computer Security Objects, NISTIR 5308, December 1993.

c. Security Labels for Open Systems - An Invitational Workshop, NISTIR 4362, June 1990.

d. Standard Security Label for GOSIP - An Invitational Workshop, NISTIR 4614, June 1991.

Scope: This standard defines syntactic constructs for conveying security label information when Government sensitive but unclassified data is exchanged over computer networks. The syntactic constructs defined in this standard are intended to be used along with semantics provided by the authority establishing security policy for the protection of the information exchanged. NIST has established a Computer Security Objects Register (CSOR) that will serve as repository for label semantics. Informative Appendix A of this standard provides further details on the CSOR.

This standard does not discuss the physical labeling of information or storage media and information displayed on a computer screen or other peripherals. Labeling of information stored in internal memory and storage media (e.g. hard disks, compact disks, magnetic tapes, etc.) is also outside of the scope of this standard. The protection of data in transit and their associated labels along with the binding between the data and the labels is the responsibility of the communications protocols involved in the transfer and therefore not discussed here. Compliance with this standard does not provide assurance of the suitability of an implementation for the protection of data according to specific security policies. That assessment must be made through the appropriate evaluation and certification processes.

Applicability: This standard applies to U.S. Government communications systems required by agency security policy to label sensitive but unclassified data when exchanged over data networks. Although this standard is intended for use on systems handling unclassified information, it could be adopted by the appropriate authorities for use on systems handling classified information.

Complying implementations shall be capable of transmitting, receiving, and obtaining information from security labels based on the specifications in this document.

Specifications: Federal Information Processing Standard (FIPS) 188, Standard Security Label for Information Transfer (affixed).

Implementation Schedule: This standard becomes effective 1995 March 1.

Waiver Procedure: Under certain exceptional circumstances, the heads of Federal departments and agencies may approve waivers to Federal Information Processing Standards (FIPS). The head of such agency may redelegate such authority only to a senior official designated pursuant to section 3506(b) of Title 44, United States Code. Waiver shall be granted only when:

a. Compliance with a standard would adversely affect the accomplishment of the mission of an operator of a Federal computer system; or

b. Compliance with a standard would cause a major adverse financial impact on the operator which is not offset by Government-wide savings.

Agency heads may act upon a written waiver request containing the information detailed above. Agency heads may also act without a written waiver request when they determine that conditions for meeting the standard cannot be met. Agency heads may approve waivers only by a written decision which explains the basis on which the agency head made the required finding(s). A copy of each decision, with procurement sensitive or classified portions clearly identified, shall be sent to: National Institute of Standards and Technology; ATTN: FIPS Waiver Decisions, Technology Building, Room B-154, Gaithersburg, MD 20899.

In addition, notice of each waiver granted and each delegation of authority to approve waivers shall be sent promptly to the Committee on Government Operations of the House of Representatives and the Committee on Governmental Affairs of the Senate and shall be published promptly in the Federal Register.

When the determination on a waiver applies to the procurement of equipment and/or services, a notice of the waiver determination must be published in the Commerce Business Daily as a part of the notice of solicitation for offers of an acquisition or, if the waiver determination is made after that notice is published, by amendment to such notice.

A copy of the waiver, any supporting documents, the document approving the waiver and any accompanying documents, with such deletions as the agency is authorized and decides to make under United States Code Section 552(b), shall be part of the procurement documentation and retained by the agency.

Where to Obtain Copies: Copies of this publication are for sale by the National Technical Information Service, U.S. Department of Commerce, Springfield, VA 22161. When ordering, refer to Federal Information Processing Standards Publication 188 (FIPSPUB188), and identify the title. When microfiche is desired, this should be specified. Prices are published by NTIS in current catalogs and other issuances. Payment may be made by check, money order, deposit account or charged to a credit card accepted by NTIS.

Federal Information
Processing Standard Publication 188

1994 September 6

Specifications for a

Standard Security Label for Information Transfer

1. INTRODUCTION

U. S. Government agencies are required to protect data essential to their operations. This requirement covers data stored, processed, and transmitted by computer and communications systems. The security label presented in this standard provides syntactic constructs that can be used to convey security information along with electronically exchanged data.

The information on a security label may indicate possible damage due to accidental or malicious disclosure, modification, or destruction of data. Labels can be used to make access control decisions, specify protective measures, and indicate handling restrictions required by the applicable security policy.

This document defines a set of security tags that carry security information. The usage of these tags in support of specific security policies is specified via registration. The registration process associates a unique name to each security tag set definition thus enabling implementations to identify the labels and process them accordingly. Unless explicitly indicated by an organization's security policy, implementations of this standard shall support label sets registered in the Computer Security Objects Register (CSOR) established by NIST. Further information on the CSOR is provided in Appendix A.

2. REFERENCES

[1] International Organization for Standardization (ISO), *Information processing systems - Open Systems Interconnection - Basic Model*, ISO 7498, 1988.

[2] International Organization for Standardization (ISO), *Information processing systems - Open Systems Interconnection - Security Architecture*, ISO 7498-2, 1988.

[3] International Organization for Standardization (ISO), *Information Technology - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1)*, ISO/IEC 8824, 1990.

[4] Housley R., *Security Labeling Framework for the Internet*, Internet RFC 1457, May 1993.

[5] Internet CIPSO Working Group, *Common IP Security Option Version 2.3*, Internet Draft, January 1993.

[6] Nazario Noel, *General Procedures for Registering Computer Security Objects*, NISTIR 5308, December 1993.

3. ACRONYMS AND DEFINITIONS

3.1 Acronyms

ASN.1 - *Abstract Syntax Notation One*.

CSO - *computer security object*.

CSOR - *Computer Security Objects Register*.

FIPS - *Federal Information Processing Standard*.

LSB - *Least Significant Bit*

MSB - *Most Significant Bit*

OSI - *Open System Interconnection* [1].

PDU - *Protocol Data Unit* [1].

3.2 Definitions

computer security object - A resource, tool, or mechanism used to maintain a condition of security in a computerized environment. These objects are defined in terms of attributes they possess, operations they perform or are performed on them, and their relationship with other objects.

Computer Security Objects Register - A collection of CSO names and definitions kept by a registration authority.

domain - See *security domain*.

entity - An active element in an open system [1].

Named Tag Set - Field containing a Tag Set Name and its associated set of security tags.

network byte order - The order defined by a network for the transmission of protocol fields that are larger than one octet. This standard assumes most significant octet first.

open system - A set of one or more computers, the associated software, peripherals, terminals, human operators, physical processes, information transfer means, etc., that forms an autonomous whole capable of processing and/or transferring information that complies with the requirements of OSI standards [1].

Open Systems Interconnection - This term qualifies standards for the exchange of information among systems that are "open" to one another for this purpose by virtue of their mutual use of applicable standards [1].

policy - See *security policy*

protocol data unit - A unit of data specified in a protocol and consisting of protocol information and, possibly, user data [1].

protocol entity - Entity that follows a set of rules and formats (semantic and syntactic) that determines the communication behavior of other entities [1].

registration authority - Organization responsible for assignment of unique identifiers to registered objects.

security attribute - A security-related quality of an object. Security attributes may be represented as hierarchical levels, bits in a bit map, or numbers. Compartments, caveats, and release markings are examples of security attributes.

security domain - A collection of entities to which applies a single security policy executed by a single authority [2].

security label - A marking bound to a resource (which may be a data unit) that names or designates the security attributes of that resource [2].

security level - A hierarchical indicator of the degree of sensitivity to a certain threat. It implies, according to the security policy being enforced, a specific level of protection.

security policy - A set of criteria for the provision of security services [2]. It defines and constrains the activities of a data processing facility in order to maintain a condition of security for systems and data.

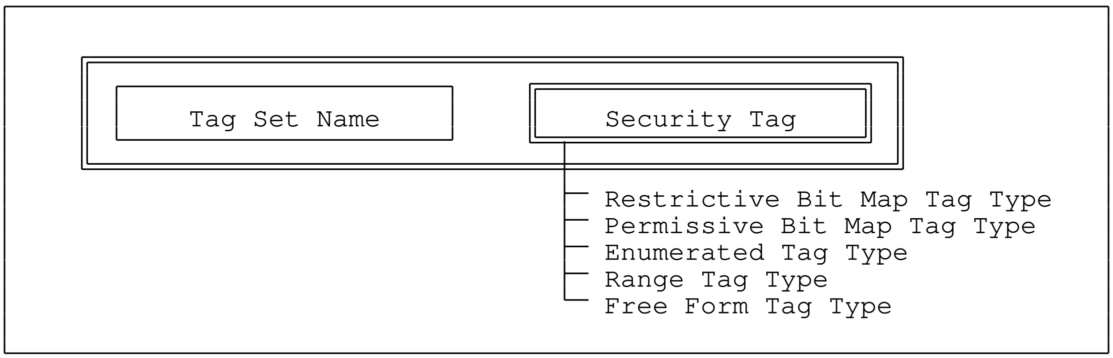
security tag - Information unit containing a representation of certain security-related information (e.g., a restrictive attribute bit map).

security threat - A potential violation of security [2].

Tag Set Name - Numeric identifier associated with a set of security tags.

4. GENERIC STANDARD SECURITY LABEL SYNTAX

The Standard Security Label (SSL) is a collection of one or more Named Tag Sets. Every Named Tag Set contains tags carrying security information preceded by a Tag Set Name. The Tag Set Name identifies a register entry where the tag set and its associated semantics are defined. The security tags carry security attributes of the data being exchanged. Security attributes may be represented in several ways, thus the need for several tag types.



— one occurrence ═ one or more occurrences

Figure 4.1: SSL Named Tag Set

4.1 Named Tag Sets

As shown in Figure 4.1, each label may contain multiple Named Tag Sets (illustrated by double lines). Each set has a Tag Set Name and one or more security tags (a tag set). Tag Set Names are either non-negative integers or ASN.1 Object Identifiers. Tag Set Names with integer values are suitable for use in labels for lower layer protocols. ASN.1 Object Identifiers (OIDs) have the form of a path through the branches of a registration authority tree (e.g., {1.2.840.101.5}) [3]. OIDs are appropriate for use in Application Layer protocols. There are five security tag types which can be combined to carry security-related data. The data is used by protocol entities to maintain the security condition of a resource of the system (e.g., communications system, data file, application process).

Every label must carry, at least, one Named Tag Set. The use of multiple Named Tag Sets is determined by the security policy enforced and restrictions imposed by the protocol using the labels. A possible reason for using multiple Named Tag Sets on a single label is a requirement for protection under multiple security policies. This may be useful in maintaining an appropriate degree of protection when data is shared across security domain boundaries.

4.2 Security Tags

The five security tag types defined are not numbered in sequential order to maintain compatibility with a labeling scheme for non-OSI communications systems [5]. Tag Types 1 and 6 are syntactically identical, only the interpretation of the bit settings is different. Although this difference could be handled via registration, Type 6 is included for compatibility with other standards.

4.2.1 Security Tag Type 1

Tag Type 1, the Restrictive Bit Map Type, contains its type, a non-negative integer, and a bit string. The non-negative integer conveys a security level, a hierarchical security attribute. The higher the value of this attribute the higher the security level of the labeled PDU. This security level could be used to restrict access so that only PDUs with security labels lower or equal to the highest level of the receiving end will be accepted. Restrictions may be extended to not accepting PDUs with labels lower than the lowest level for the receiving end, and so on.

The bit string is used to convey a set of non-hierarchical attributes that apply to the labeled PDU. A bit is assigned to every security policy-defined restrictive attribute. Bits corresponding to restrictive attributes that apply will be set to 1, otherwise bits are set to 0. Access could be restricted to only those PDUs whose set of attributes is a subset of the attributes for the receiving end. Security compartments and caveats are examples of restrictive security attributes.

4.2.2 Security Tag Type 2

Tag Type 2, the Enumerated Type, contains its type, and one or more non-negative integers. The non-negative integer conveys a security level, a hierarchical security attribute. The higher the value of this attribute the higher the security level of the labeled PDU. Each of the integers that follow represent a non-hierarchical attribute that applies to the labeled PDU. This is an alternative to the bit-representation in Tag Types 1 and 6. It is intended to minimize label length in cases where only a few attributes out of a large set apply to the PDU. Attributes enumerated by tags of this type could be restrictive (e.g., compartments) or permissive (e.g., release permissions). Access could be restricted to only those PDUs whose set of attributes is a subset of the attributes for the receiving end. Alternatively, a PDU could be accepted if the receiving end belongs to any of the release groups in the release permission list on the PDU label. The registered tag set semantics indicate how the security attributes on tags of this type are interpreted.

4.2.3 Security Tag Type 5

Tag Type 5, the Range Type, contains its type, and one or more non-negative integers. The non-negative integer conveys a security level, a hierarchical security attribute. The higher the value of this attribute the higher the security level of the labeled PDU. Each of the integers that follow appear in pairs and represent, respectively, the upper and the lower bounds of ranges of non-hierarchical attributes that apply to the labeled PDU. This is an alternative to the bit-representation in Tag Types 1 and 6. It is intended to minimize label length in cases where all attributes beginning with upper-bound and ending with lower-bound apply to the PDU. A single tag may indicate multiple security attribute ranges. These ranges shall be listed in descending numerical order and shall not overlap. Each upper and lower bound attribute is indicated by a fixed size number. Attributes identified by tags of this type could be restrictive (e.g., compartments) or permissive (e.g., release permissions). Access could be restricted to only those PDUs whose set of attributes is a subset of the attributes for the receiving end. Alternatively, a PDU could be accepted if the receiving end belongs to any of the release groups in the release permission list on the PDU label.

4.2.4 Security Tag Type 6

Tag Type 6, the Permissive Bit Map Type, contains its type, a non-negative integer, and a bit string. The non-negative integer conveys a security level, a hierarchical security attribute. The higher the value of this attribute the higher the security level of the labeled PDU. The bit string is used to convey a set of non-hierarchical attributes that apply to the labeled PDU. A bit is assigned to every security policy-defined permissive attribute.
Release markings are examples of permissive security attributes. Bits corresponding to types or groups of entities that are granted access to the PDU are set to 0, all other bits are set to 1. For example, the label on PDUs to be available only to members of an organization's Personnel Office will have the bit assigned to the Personnel Office set to 0 and all others set to 1.

A PDU can be accepted if the receiving end belongs to any of the release groups in the release permission list on the PDU label.

4.2.5 Security Tag Type 7

Tag Type 7, the Free Form Type, is intended as a wild-card tag type that may carry any user-defined type of data appropriate for use with the protocol handling the labels. The full specification of the format of this field shall be provided via registration. Examples of data that may be conveyed with this Tag Type are human/machine readable time stamps, human-readable policy identifiers, and privacy marks.

5. APPLICATION LAYER STANDARD SECURITY LABEL SYNTAX

This section gives the security label specification for use at the Application Layer. The Abstract Syntax Notation One (ASN.1) definition for the labels is given in Section 5.1. This specification is derived from the generic syntax presented in Section 4.

The specification allows multiple Named Tag Sets on a single label, therefore implementations shall be able to skip over unrecognized tag sets until a recognized set is found or the label ends. Failure to recognize a Named Tag Set while scanning the label may constitute a security relevant event (i.e., may represent a violation of the security policy that require further action). Indication of occurrence and the ability to log this and all security relevant events, shall be provided by all SSL implementations. The determination of the action to follow upon detection of a possible security relevant event is a policy decision outside the scope of this standard.

5.1 ASN.1 Definition for the Standard Security Label

```
StandardSecurityLabel       ::= SET OF NamedTagSet

NamedTagSet                 ::= SEQUENCE {
        tagSetName          TagSetName,
        securityTags        SEQUENCE OF SecurityTag }

TagSetName                  ::= OBJECT IDENTIFIER

SecurityTag                 ::= CHOICE {

  -- Type 1 - for restrictive security attributes
     restrictivebitMap          [1] IMPLICIT SEQUENCE {
             securityLevel              SecurityAttribute,
             attributeFlags             BIT STRING      }

  -- Type 2 - security attributes by number
     enumeratedAttributes       [2] IMPLICIT SEQUENCE {
             securityLevel              SecurityAttribute,
             attributeList              SET OF SecurityAttribute }

  -- Type 5 - all security attributes in the range(s)
     rangeSet                   [5] IMPLICIT SEQUENCE {
             securityLevel              SecurityAttribute,
             rangeList                  SET OF SecurityAttributeRange }
```

```
-- Type 6 - for permissive security attributes
    permissivebitMap            [6] IMPLICIT SEQUENCE {
          securityLevel                SecurityAttribute,
          attributeFlags               BIT STRING      }

-- Type 7 - format specified via registration
    freeFormField               [7] IMPLICIT ANY DEFINED BY TagSetName   }

SecurityAttributeRange    ::= SEQUENCE {
    upperBound            SecurityAttribute,
    lowerBound            SecurityAttribute }

SecurityAttribute         ::= INTEGER (0..MAX)
```

6. NETWORK LAYER SECURITY LABEL SPECIFICATION

This section gives the security label specification for use within Network Layer protocols. This format, derived from the Generic SSL Syntax, is optimized for use at in layer 3 protocols and therefore different from the format that would result from encoding the ASN.1 definition in Section 5 using the ASN.1 Basic Encoding Rules.

6.1 Network Layer Security Label Format

The Network Layer label has two main parts, a fixed-length header and a variable-length tag section. All multi-octet fields are defined to be transmitted in network byte order with the most significant bit of every octet being transmitted first. Therefore, all fields are shown with the most significant bit on the left and the least significant bit on the right. Figure 6.1 shows the standard security label format for the lower layers.

Header			Security Tag Set		
Security Label Identifier	Security Label Length	Security Tag Set Name	Security Tag	...	Security Tag
1 octet	1 octet	4 octets		Var	

Figure 6.1 - Security Label Format

6.2 Security Label Identifier

This field is one octet in length. Its value is 134 (10000110_b).

6.3 Security Label Length

This field is one octet in length. Its value is the total length of the option in octets including the type and length fields. The maximum length value is 255. Note that further length restrictions may be imposed by the specific protocol carrying the label.

6.4 Security Tag Set Name

Security data is conveyed using Security Tags. The Security Tag Set Name is the numeric name that identifies the registered semantic rules that give meaning to the security data in the label. The rules for interpretation of labels are registered in a Computer Security Objects Register (CSOR). Information on this register appears in Appendix A. The registered rules include the size, type, and number of security tags that appear on a label.

Unlike in upper layer security labels, the Security Tag Set Name carries a fixed-length value. The length of the field is four octets and valid values are positive numbers from 1 to $0xffffffff_h$. The value 0 is reserved and must not appear as the Security Tag Set Name in any label.

> Note: The registration process assigns both a numeric and an alpha-numeric name to every object (i.e. Named Tag Set). See *General Procedures for Registering Computer Security Objects* [6] for more information.

6.5 Security Tags

A common format for passing security related data is necessary for interoperability. This standard currently defines five types of security tags to carry security attributes of the data in a PDU. Each Security Tag has a one-octet type field and a one-octet length plus a variable size data field. Only tags 1, 2, 5, 6 and 7 are currently defined, this standard reserves all other tag types for future use. Figure 6.2 below shows the general format for security tags.

TTTTTTTT	LLLLLLLL	BBBB ... BBBB
Tag Type	Tag Length	Security Data

Figure 6.2 - General Security Tag Format

6.5.1 Tag Type

This field is 1 octet in length and identifies the format used to represent the security data, e.g., bit map, list of two-octet attribute numbers, pairs of attribute numbers, etc.

6.5.2 Tag Length

This field is 1 octet in length. Its value is the total length of the tag type including the type and length fields.

6.5.3 Security Data

This is a variable-length field. It carries security attributes of the data in the PDU. The Security Data field begins with two one-octet sub-fields. The first one is an all-zero Alignment Octet. Its purpose is to fit the fixed-length part of the tags in a 32-bit field. The second one-octet field is a Security Level. Its value may range from 0 to 255. The values are ordered with 0 being minimum security level and 255 representing the maximum security level. This field is used to convey a hierarchical security attribute. The format of the rest of the Security Data field is different for every tag type and is defined below for each of the currently defined types.

6.6 Security Tag Type 1

Tag type 1 is the Restrictive Bit Map Tag Type. Tags of this type are used to convey restrictive security parameters, such as compartments and protection categories, that may be selected from a set by setting a one-bit flag. Security attributes conveyed by this tag type are used to limit the entities allowed to access the data in the PDU to those with matching attributes. The format of this tag type is as follows:

				0123 ...
00000001	LLLLLLLL	00000000	LLLLLLLL	BBBBBBBBBBBBBBBB
Tag Type	Tag Length	Alignment Octet	Security Level	Bit Map

Figure 6.3 - Security Tag Type 1 Format

6.6.1 Restrictive Security Attribute Bit Map

The length of this field is variable. The maximum length is 245 octets, although further restrictions may be imposed by the protocol where the labels are used. The minimum length is 0 octets. The ordering of the bits is left to right or most significant bit (MSB) to least significant bit (LSB). For example security attribute 0 is represented by the MSB of the first octet and security attribute 15 is represented by the LSB of the second octet. Bit maps shall be padded with 0s to the right (i.e., up to the least significant bit of the last octet), if necessary. Figure 6.4 graphically shows this ordering.

Bit N is binary 1 if attribute N is part of the label for the PDU, and bit N is binary 0 if attribute N is not part of the label.

```
           Octet 0  Octet 1  Octet 2  Octet 3  Octet 4  Octet 5
          |xxxxxxxx|xxxxxxxx|xxxxxxxx|xxxxxxxx|xxxxxxxx|xxxxxxxx|...

Bit        01234567 89111111 11112222 22222233 33333333 44444444
Number                012345 67890123 45678901 23456789 01234567
```

Figure 6.4 - Bit Ordering for Bit Map Tags

6.7 Security Tag Type 2

Tag type 2 is the Enumerated Tag Type. Tags of this type are used when only a few security attributes, out of a large set, apply to the data in a given PDU. This is done by assigning a two-octet non-negative binary number to each security attribute and enumerating those attributes that apply. The format of this tag type is as follows:

```
| 00000010 | LLLLLLLL | 00000000  | LLLLLLLL | CC CC CC CC CC CC CC |

  Tag Type  Tag Length Alignment  Security   Enumerated Attributes
                       Octet      Level
```

Figure 6.5 - Security Tag Type 2 Format

6.7.1 Enumerated Categories

In tags of this type, security attributes are listed by their assigned number value rather than by their position within a bit field. A two-octet number is used to identify each security attribute. Valid values for security attributes are 0 to 65534. Attribute value 65535 is not a valid attribute value.

Note that the two-octet numbers could be used to convey ASCII character pairs as an alternative way of identifying security attributes.

6.8 Security Tag Type 5

Tag type 5 is referred to as the Range Tag Type. It is used to represent labels where all categories in a range, or set of ranges, apply to the data in a PDU. The format of this tag type is as follows:

00000101	LLLLLLLL	00000000	LLLLLLLL	Top/Bottom	Top/Bottom
Tag Type	Tag Length	Alignment Octet	Security Level	Attribute Ranges	

Figure 6.6 - Security Tag Type 5 Format

6.8.1 Security Attribute Ranges

Attribute ranges are pairs of two-octet values that represent the top and bottom security attributes of a range respectively. These range endpoints are included within the range of attributes. All attributes within a range apply to the data in the PDU. The bottom attribute endpoint for the last pair in the tag may be omitted when its value is 0. The ranges must be non-overlapping and be listed in descending order. Valid values for security attributes range from 65534 to 0. Attribute value 65535 is not a valid attribute value.

6.9 Security Tag Type 6

Tag type 6 is the Permissive Bit Map Tag Type. Tags of this type are used to convey permissive security parameters, such as release markings, that may be selected from a set by resetting a one-bit flag. Security attributes conveyed by this tag type are used to indicate groups of entities are allowed to access the data in the PDU. The format of this tag type is as follows:

				0123 ...
00000110	LLLLLLLL	00000000	LLLLLLLL	BBBBBBBBBBBBBBBB
Tag Type	Tag Length	Alignment Octet	Security Level	Bit Map

Figure 6.7 - Security Tag Type 6 Format

6.9.1 Permissive Security Attribute Bit Map

The length of this field is variable. The maximum length is 245 octets, although further restrictions may be imposed by the protocol where the labels are used. The minimum length is 0 octets. Bits in the map shall be numbered left to right starting with the MSB of the first transmitted octet. For example, security attribute 0 would be represented by the most significant bit of the first octet while security attribute 15 would be represented by the least significant bit of the second octet. Figure 6.4 graphically shows this ordering. Bit maps shall be padded with 1s to the right (i.e., up to the LSB of the last octet), if necessary.

Bit N is binary 0 if entities in group N are allowed to access the data in the PDU, and bit N is binary 1 if entities in group N are not allowed access.

6.10 Security Tag Type 7

Tag type 7 is the Free Form Tag Type. Tags of this type are used to convey a free format field of up to 247 octets. The Security Data field of this tag may hold character strings, or any user-defined data relevant to Layer 3 processing. See *Security Labeling Framework for the Internet* [4], for a discussion on relevant security data. The format of that data must be specified via registration.

The format of this tag type is as follows:

00000111	LLLLLLLL	FFFF FFFF
Tag Type	Tag Length	Free Form Field

Figure 6.8 – Security Tag Type 7 Format

Appendix A - The Registration Service (*Informative Appendix*)

This standard relies on the availability of a registration service to assign Tag Set Names and serve as the repository of the semantics, special handling rules, and other details required for the implementation and use of security policy-specific label sets. One such service has been established by NIST. The Computer Security Objects Register (CSOR), is defined in the document *General Procedures for Registering Computer Security Objects* [6]. The document contains generic and object-specific registration procedures for security labels and other security objects. The following are examples of the items whose value and significance shall be provided upon registration of a Named Tag Set:

- number of tags,
- length of the set,
- length of each tag,
- ordering of tags,
- full format and semantics for Type 7 tags, and
- security-relevant conditions

Copies of the document and further information are available from the following address:

Computer Security Objects Register

National Institute of Standards and Technology
Computer Systems Laboratory
Program Coordination and Support Group
Building 225, Room B151
Gaithersburg, Maryland 20899
Telephone: (301) 975-2821
Facsimile: (301) 948-1784

Appendix B - Basic Processing Rules (*Informative Appendix*)

To insure support for existing government computer security policies, implementations of the SSL are required to meet mandatory access control requirements in Section 3.1.1.4 of the Orange Book (DOD 5200.28-STD). This Appendix contains a minimal set of processing rules in support of those requirements.

B.1 Trustworthiness of Transmitted Labels

Security labels used in communication systems are intended as an extension to end system labels. It is therefore necessary to ensure the integrity of the labels and their binding to the corresponding data units. Implementations of the SSL shall support requirements set forth in Section 3.1.1.3 of the Orange Book for the integrity of the labels and their binding to the labeled data.

B.2 Minimum Originator Requirements

a. Security label information shall be obtained from a reliable source within the originating system (e.g., a trusted computing base, security management information base). The establishment of the reliability of the label information at the originating system is a local matter. A translation step may be necessary to express the end system label and other attributes in the appropriate SSL format.

b. The choice of the Named Tag Set and label value(s) depends on the label set negotiated for the security association[1] and the local security attributes of the data being exchanged. The label value for every outbound PDU must be within the range established for the security association. **Note**: The range of values could contain a single label value.

c. Data units with out-of-bounds label values shall be discarded and audited.

d. An attempt to send data outside the value range established by the security association constitutes a security relevant event and shall be reported. Implementations shall

[1] A *security association* is a set of security attributes agreed upon by the communicating parties for the protection of information to be transferred. It may define attributes such as the range of allowed security label values, identifiers of protection algorithms (e.g., for integrity and confidentiality), encipherment keys, et cetera.

provide the option to log the event in an audit trail and to notify the upper layer or application program of the error.

e. The system administrator shall be able to establish, at configuration time, thresholds and parameters such as whether a type of security relevant event is to be audited and whether notification shall be provided to the upper layer or application program. This determination is a policy-based decision.

f. At most one label shall be generated for any protected PDU.

g. If all required security information cannot fit in the appropriate tags, the whole message shall be discarded and audited. Implementations shall provide for an optional error message to be passed to the upper layer.

h. If the security label cannot fit in the corresponding protocol header, the whole message shall be discarded and audited. Implementations shall provide for an error message to be passed to the upper layer.

i. If a routing protocol entity cannot establish an appropriate route based on the label, the PDU shall be discarded and the problem audited. Implementations shall provide for an optional error message to be passed to the upper layer.

B.3 Minimum Receiver Requirements

a. Upon receipt, labels shall be parsed based on the semantic rules pointed to by the value in the Tag Set Name field of the label.

b. PDUs with the wrong Tag Set Name for the security association, missing labels, label errors (i.e., any part of the label fails to follow the SSL format or the registered specifications), or out-of-bounds label values shall be discarded. Receipt of such PDUs may require audit. An optional error report PDU could be returned if allowed by the applicable security policy.

c. All PDUs shall contain at most one label. Detection of more than one label will cause an error. A PDU with multiple labels shall not be accepted.

   Note: Depending on the protocol carrying the label and attributes of the security association between the communicating ends it is possible to carry multiple Named Tag Sets on a single label.

d. Whether or not a label must be present on a PDU depends on the attributes for the security association under which the PDU is protected.

e. Implementations shall be able to log security relevant events, such as label errors, in an audit trail and to return error report PDUs. The system administrator shall be able to establish, at configuration time, thresholds and parameters such as whether a type of security relevant event is to be audited and whether to return an error report PDU. This determination is a policy-based decision.

B.4 Minimum Intermediate System Requirements

Depending on the protocol using the labeling function, intermediate systems may have to process label information.

a. Intermediate systems shall maintain parsing information for the Tag Set Name(s) supported. Upon receipt, labels shall be parsed based on the semantic rules pointed to by the value in the Tag Set Name field of the label.

b. PDUs with the wrong Tag Set Name for the security association, missing labels, label errors (i.e., any part of the label fails to follow the SSL format or the registered specifications), or out-of-bounds label values shall be discarded. Receipt of such PDUs may require audit. An optional error report PDU could be returned if allowed by the applicable security policy.

c. The intermediate system shall provide the option of forwarding or dropping PDUs with unrecognized Tag Set Names and unlabeled or unprotected PDUs. Appropriate behavior is determined by the system administrator at configuration time according to the applicable security policy. Receipt of such PDUs may require audit.

d. At most one label can be present on a PDU. Detection of more than one label will cause an error. A PDU with multiple labels shall not be forwarded.

B.5 Error Report PDUs

Although specific error PDU format is protocol dependent, there is a common set of events for which sender notification could be required. Those events are:

Incoming violation

Out-of-bounds label	-	At least one security attribute value on a label is outside of the range of accepted values for the corresponding security association.
Unrecognized label	-	The Tag Set Name(s) on the incoming label is (are) not valid for the corresponding security association (or unrecognized by the receiving end).

Bad label	-	At least one error is found when parsing the incoming label. This includes the case when more than one Named Tag Set is found on a label for which only one is accepted.
Label missing	-	No security label is found although one is required by the security association; or more than one label is present on any PDU header.
Forwarding violation	-	An intermediate system is unable, due to policy restrictions or inability to obtain a valid security association, to forward a PDU.

No error reports shall be generated due to problems with incoming error report PDUs, although implementations shall provide the option to audit such problems.

B.6 Policy-Based Processing Rules

The following policy-based processing rules support the Department of Defense Mandatory Access Control Policy. That policy separates all system elements into *objects*[2] and *subjects*. The data carried in a PDU is an object and anything that can send or receive objects (e.g., a host, an application) is a subject. Each object has a "sensitivity" label associated to it. Every subject is assigned a range of label values that it is authorized to send and a range of label values that it is authorized to receive. These labels have a hierarchical "sensitivity level" and a set of "sensitivity categories". For a subject to have access to an object, the following criteria must be met:

(1) The upper level of the subject's "receive range" must be greater or equal to sensitivity level of the object,

(2) the lower level of the subject's "receive range" must be less or equal to sensitivity level of the object, and

(3) the set of sensitivity categories for the subject must include all the sensitivity categories for the object.

[2] The definition of the term *object*, in the context of this discussion of security policies, is different from that used throughout the main body of this standard.

Tag Type 1 supports this "restrictive" security policy and should be processed accordingly. Tag Types 2 and 5 may also support the same policy if specified explicitly and unambiguously through registration.

A complimentary security policy based on release authorizations or "release markings" is also available. Under such policy objects and subjects have a list of release categories. For a subject to have access to an object it must have at least one release category in common with the object. For instance, subjects acting on behalf of an organization's Personnel Department staff can have access to objects carrying a release marking for that department.

Tag Type 6 supports this "permissive" security policy and should be processed accordingly. Tag Types 2 and 5 may also support the same policy if specified explicitly and unambiguously through registration.

It is expected that most security label implementations will support these policies. Named Tag Sets that support either or both of these policies may be defined and registered. The specific value range for an instance of communication should be established a priori when setting up a security association between the communicating ends. When tags supporting both policies appear on a label the restrictive tag (supporting the sensitivity policy) is processed first. If that succeeds, then the permissive tag (supporting the release policy) is processed. In a restrictive security tag, the hierarchical component (sensitivity level) is processed first. Only if the sensitivity level is within the valid range for the security association are the sensitivity categories tested. In a permissive tag the security level field could carry a sensitivity level, as in the restrictive tags, or a null value. If both restrictive and permissive tags are carried on the same label only the security level field in the restrictive tag shall contain significant information, the permissive tag must carry a null value in that field. The release markings may only be processed after all other tests are passed.

Appendix C - Special Usage Provision (*Informative Appendix*)

If allowed by an electronic messaging protocol, the SSL Layer 3 encoding may be used by an implementation of that protocol.