OSI AND X.400 SECURITY

Chris J. Mitchell
Hewlett-Packard Laboratories
Filton Road
Stoke Gifford
Bristol BS12 6QZ
England

5th February 1990

SUMMARY

This paper is concerned with security in OSI, and in particular with the security features within CCITT X.400. Some idea is given of how these security features can be used to provide secure store-and-forward messaging, and some limitations of the security provisions are discussed.

1. INTRODUCTION

The main purpose of this paper is to describe the security features within the 1988 CCITT X.400 Recommendations, [2], and to give an idea of how these features may be used to provide a secure store-and-forward message handling system. Certain limitations of these security features are also indicated. All this material can be found in Section 5 of this paper.

As an introduction to this discussion, a brief survey to general standardisation efforts for OSI security is given. The work of the main international standards committees involved in work on security for OSI (i.e. ISO, CCITT and ECMA) can be divided into three main parts. First there is work on underlying techniques, such as: cryptographic algorithms, modes of operation for cryptographic algorithms and peer entity authentication mechanisms. Second there is more general work describing how these techniques may be used to provide security in both OSI applications and various layers of the OSI model, such as: the OSI security architecture, Lower and Upper Layer security models and various security frameworks. These first two areas are very briefly discussed in Sections 3.1 and 3.2 below. The third area of standardisation effort includes the X.400 work, and is concerned with specifying how security should be provided in specific OSI applications; this is covered in Section 4.

Preliminary even to these remarks, in the next section a brief review of the OSI 7-layer model is given in order to set subsequent remarks in context.

2. THE OSI 7-LAYER MODEL

The aim of Open Systems Interconnection (OSI) is to provide a standardised means of communication between diverse computer systems. As a basis for the development of OSI standards, ISO have developed a Reference Model to partition the problem into discrete layers, and to provide a conceptual framework for understanding the complex problems involved.

The Reference Model has seven layers; from the 'bottom up' they are as follows:

1. Physical Layer
2. Link Layer
3. Network Layer
4. Transport Layer
5. Session Layer
6. Presentation Layer
7. Application Layer

The Reference Model specifies the functionality of each layer and the interfaces between adjacent layers. It also defines methods for

achieving layer-specific functionality between cooperating computer systems.

The lowest three layers (Physical (1), Data Link (2) and Network (3)) are concerned with the provision of data transmission. The Physical Layer models the interface of a computer system to the physical medium. It includes such aspects as physical connectors and voltage levels. The Data Link Layer provides a framework around data for transmission by the Physical Layer; detection and correction of errors may be performed by this layer. The Network Layer is particularly concerned with routing and relaying. The services offered by the Network Layer to the Transport Layer conceal from it the numbers and types of sub-network that may be involved in the communication.

The Transport Layer (4) operates end-to-end between computer systems and is concerned with Quality of Service. The Transport Layer is responsible for providing the Session Layer with a reliable data transmission service.

The Session Layer (5) assumes reliable data transmission services between computer systems (i.e. end-to-end communications). It occupies the area between the application-oriented upper layers (6 and 7) and the 'real-time' data communication environment. It provides services for the management and control of data flow between two computer systems.

The function of the Presentation Layer (6) is to provide a common representation of information whilst in transit between computer systems.

The Application Layer (7) provides the communication-based service to end users. The other six layers of the model exist to support and make possible the activities that take place at the Application Layer.

For further information about OSI see, for example, Henshall and Shaw's book, [4.5].

3. OSI SECURITY ACTIVITIES

3.1 Security techniques

Within ISO, work on techniques for security, in particular on cryptographic techniques, has been primarily focussed within ISO/IEC/JTC1 SC20 (and will be continued by its proposed successor SC27). Outside ISO, other work has proceeded within ANSI and the NBS (in the U.S.A.). This work can be conveniently divided into three areas: algorithms (e.g. encryption functions, digital signature functions), peer entity authentication protocols and key management.

After the failure of attempts to standardise specific encryption algorithms, it was decided that ISO would change tack. Instead, it has been decided to adopt the idea of an international register of algorithms, through which any encryption algorithm can be given a standardised identifier. The draft proposal ISO DP 9979, [10], caters for registering proprietary algorithms, the details of which may remain confidential to their owners. An international standard, ISO 8372, [6], specifying modes of use for an arbitrary 64-bit block cipher algorithm has also been produced. A successor to this standard, in the form of a

draft proposal, ISO DP 10116, [11], generalises this further to specifying modes of use for an N-bit block cipher algorithm.

In addition to data confidentiality, a good deal of work has also been done within ISO (and other standards bodies) concerning standardising algorithms for message authentication, integrity checking and digital signature. A draft international standard now exists, ISO DIS 9797, [9], for a data integrity mechanism. Two standards proposals exist relating to digital signature algorithms. The first is a draft proposal, ISO DP 10118, [12], specifying possible methods for computing hash functions for digital signatures; note that one of the methods described there will probably need to be removed in the light of recent work by Coppersmith, [4]. The second is a proposal for a signature algorithm for 'short' messages, ISO DP 9796, [8].

In parallel with the current work within ISO on algorithms, efforts have also been made to standardise the protocol exchanges involved in performing party-to-party authentication. This has resulted in drafts for a multi-part standard.

ISO work on key management is at an early stage of development. Three draft documents exist, entitled: Cryptographic mechanisms for key management: Part 1: Key management overview, Part 2: Key management using secret key techniques and Part 3: Key management for public key register. It is likely to be some time before any of these documents emerge as Draft Proposals, since at the moment none of them are any where near completion.

3.2 Using security mechanisms

Within ISO, the questions of how and where within the OSI model security mechanisms are to be used falls primarily within the scope of SC21/WG1, together with the layer and application specific Working Groups of SC6 and SC21. Work in this area can be divided into three parts, namely: security architectures and models, security frameworks and layer specific standards.

Most of this work is at a very early stage, and we do not discuss it further here. The main exception is the OSI Security Architecture, ISO 7498-2, [5], released as an International Standard in 1988. This document covers a number of important topics, including: standardised definitions of security terminology and security services, a guide to the relationship between security services and mechanisms, an indication of which security services are relevant to which layers of the OSI model and a short introduction to security management.

4. OSI APPLICATION LAYER SECURITY

We now very briefly consider the effort that has been devoted to providing standardised security solutions for specific OSI applications.

The 1988 version of the X.500 CCITT Recommendations on Directory Services, [3], and their corresponding ISO draft standards, [7], include means to use the Directory Service to provide key management and peer-entity authentication through storage of user public keys in the

directory. The 1992 version of these recommendations is also expected to contain detailed provisions for access control to directory entries.

The 1988 versions of the X.400 CCITT Recommendations, [2], include a variety of security features making it possible to provide a variety of security services for electronic mail. We discuss these provisions in more detail in the next section.

In parallel with the general growth in interest in EDI (Electronic Data Interchange), there has also been a very rapid growth in concern regarding the security of EDI messages. For those EDI messages transmitted using X.400 networks, use of the X.400 security features may be sufficient. However, for EDI messages sent by other means, or where security services are required which cannot be provided using the X.400 features, EDI may need to be enhanced to incorporate security elements. This is an area of current debate.

5. SECURITY FOR X.400 STORE-AND-FORWARD MESSAGING

5.1 Introduction

We now consider in more detail security in electronic mail applications, with particular reference to the security features in the 1988 versions of the CCITT X.400 Recommendations, [2]. We devote the remainder of this introduction to a brief review of the fundamental concepts underlying the X.400 electronic mail system.

The 1984 version of the X.400 recommendations, [1], defines two basic types of entity in a 'store and forward' mail network, namely User Agents (UAs) and Message Transfer Agents (MTAs). UAs originate and receive messages on behalf of users. All messages are sent via one or more MTAs, which act as 'store and forward' message nodes. The set of all MTAs collectively form what is known as the Message Transfer Service (MTS).

X.400 is widely used as a generic term for a collection of related C.C.I.T.T. Recommendations, including X.400 itself, X.402, X.411, X.413 and X.420, [2]. The protocols governing communication between pairs of MTAs and between a UA and the MTS are defined in X.411. The protocol governing MTAÄMTA communications is often referred to as P1, and the UAÄMTA protocol as P3. The entire collection of UAs and MTAs is referred to as the Message Handling System (MHS).

In the 1988 version of the X.400 Recommendations, [2], in fact in X.413, a third type of entity is defined, namely a Message Store (MS). Message Stores were not part of the 1984 version of X.400. In some cases it is convenient to only connect a UA to the MTS at very infrequent intervals. However MTAs may only store mail for recipient UAs for a short period of time. The role of a MS is to remedy this problem by acting as an intermediary between a UA and the MTS, with storage of received messages as its primary role. UAs and MSs are in 1Ä1 correspondence, and an MS enables its corresponding UA to obtain summary information about received messages without actually retrieving them. In practice, an MS is likely to be co-located either with an MTA or with its corresponding UA. The Message Store Access Protocol (sometimes referred to as P7), governing the retrieval of messages by a UA from its corresponding MS, is defined in Recommendation X.413. Note that UAs and MSs are collectively referred

to as MTS-users, in that they are both end-users of the Message Transfer Service.

All the protocols so far discussed, namely those in X.411 and X.413, have the role of defining how an object called a message-content is shipped from one UA to another. The form of this content is not constrained by X.411 or X.413, and may be one of a number of different types. It is carried transparently by the MTS. One such type is defined in X.420; this type is defined as suitable for use in Inter-Personal Messaging applications. Other content types may be defined for different applications such as Electronic Data Interchange (EDI).

Finally note that the set of parameters defined in X.411 and X.413, which accompany the message content when it is transferred from one MHS entity to another, are often referred to as the message envelope. This is because in many ways these parameters have roles analogous to those of the addressing and franking information to be found on the envelope of conventional paper mail. However, in X.400 the form and content of the envelope depends on the type of entities which are involved in the transfer, e.g. submission envelopes are used to transfer from a UA to the MTS and delivery envelopes are used to transfer from the MTS to a UA.

5.2 Security services

Before describing electronic mail security services in detail, it is useful to consider what threats these services are intended to counter. Possible threats to electronic mail systems include: masquerade, message replay/re-sequencing, modification of message information, denial of service, leakage of information and repudiation. It is not possible to address all these threats from within a message handling application. For example information leakage will take place if it is possible to monitor the volumes of traffic going from one point in the network to another, even if all the message contents are encrypted. Prevention of this leakage requires the provision of security services in the lower layers of the OSI stack, which is beyond the scope of application services.

There are a considerable number of different security services that could be provided within an electronic mail system. Such services may conveniently be divided into two classes, namely MTS-user to MTS-user services and MTS services (note that this is non-standard terminology).

MTS-user to MTS-user services are those provided from one MTSÄuser (i.e. a UA or an MS) to another, without active participation by the MTS. Such services include: Message origin authentication, Content confidentiality, Content integrity, Message sequence integrity and Non-repudiation of origin.

MTS security services are those provided which involve active participation by the MTS. Such services include: Secure access control to the MTS and between MTAs, Report origin authentication, Proof of delivery, Probe origin authentication, Proof of submission, Non-repudiation of submission, Non-repudiation of delivery and Message security labelling.

The service names used here are those given in the X.400 Recommendations. These do not correspond precisely with the names used in ISO 7498-2, the

OSI security architecture, [5]. This is partly because the OSI security architecture does not mention all the services relevant to electronic mail, and partly because the documents were developed in parallel.

5.3 Approaches to providing security

In order to provide security services for the message content it is normally necessary to transmit with the message a number of 'security parameters', e.g. encrypted keys and authentication checks. These security parameters can either be transmitted in the message envelope or as part of a (specially formatted) message content, or both. The choice of location for the security parameters not only has important system ramifications, but can also affect the type of security service which may be provided.

If security services are required for X.400-1984, or other electronic mail systems without built in security features, then there is no alternative but to put the security parameters in the message content. The same is true for any heterogeneous mail systems, even if they individually incorporate security features. Examples of electronic mail systems in which all the security parameters are in the message content are provided by the SDNS and IAB Internet mail security proposals. However, security parameters within the message content cannot be used to provide MTS security services.

A distinct feature of the 1988 X.400 Recommendations is that the message envelope is used to transfer security parameters, and not the message content. The inclusion of the security parameters in the message envelope enables the provision of MTS security services. However, it does make the provision of certain MTS-user to MTS-user services rather problematical, especially if Message Stores are used.

5.4 Security mechanisms

Before we consider the security mechanisms described in the X.400 Recommendations, we need to consider the provision of cryptographic key management, a fundamental requirement for the provision of communications security services. Key management for the X.400 security facilities is achieved by use of the directory authentication service specified in C.C.I.T.T. Recommendation X.509, [3]. This key management system is based on the use of public key cryptosystems for digital signature and data encryption. Recommendation X.509, [3], allows public keys to be stored in user directory entries.

5.4.1 Certificates

Since the directory service (and communications with it) may not be trusted, means need to be provided for users to verify public keys read from the directory. This is provided for by the use of data structures called certificates, which we now briefly describe.

In order to set up a key management system for X.400, every user who wants to use security services must first exchange public keys with an off-line entity called a Certification Authority (CA). Each user must trust the CA which they appoint to act on their behalf. The CA gives the user a copy of its public key (each CA has its own public key/secret key

pair), and is given in return a copy of the user's public key (each user must also equip themselves with a key pair). The CA then signs a copy of the user's public key, together with the user's name and the period of validity of the key, using the CA's secret key. This forms a certificate and is actually what is put in the directory. Any other user which has a trusted copy of this CA's public key can then check the validity of the certificate, and thereby obtain a verified copy of the user's public key.

The scheme so far described does not cover the situation where two users are served by different CAs. To cover this possibility, one CA may generate a certificate for another CA's public key; such certificates are called 'cross-certificates'. If user A has CA X, and user B has CA Y, then if A is given a cross-certificate containing Y's public key signed by X, then A can obtain a verified copy of Y's public key. Once it has this key, A can then check B's certificate. Such cross-certificates can be made into chains called 'certification paths'.

5.4.2 Tokens

Virtually all the security services built into the X.400 Recommendations make use of a cryptographic construct called a token. Tokens are always formed for a single recipient. A token consists of a series of data fields with a digital signature appended, this signature being computed as a function of all the data fields in the token (using the originator's secret key). These data fields include: recipient-name, date/time of generation, a field called 'signed-data' and a field called 'encrypted-data'. The information within the encrypted-data field is enciphered using the public key of the intended recipient of the token (prior to computation of the signature).

One form of token is called a message-token, and is used in the provision of all the MTS-user to MTS-user security services. Hence, if a message requires such services, then a message-token is sent as one parameter within the message envelope. The precise contents of the signed-data and encrypted-data fields within the message-token depend on which selection of security services is required. However, whichever services are required, the presence of these data within the token prevents them from being changed and/or repudiated, since the token has been signed with the originator's secret key.

In a message-token, the encrypted-data field may be used to contain any of the following items: a cryptographic key (used to encrypt the message content if content confidentiality is required), a content integrity check (used in the provision of content integrity), a message security label, a content integrity key (used to compute the content integrity check) and a message sequence number (used in the provision of message sequence integrity). The signed-data field may be used to contain any of the following items: a content integrity check (used in the provision of content integrity), a message security label, a message sequence number (used in the provision of message sequence integrity) and a proof of delivery request.

The proof of delivery and non-repudiation of delivery services are slightly different from other MTS-user to MTS-user services in that they are provided by the message recipient to the message originator. If a message is received containing a proof of delivery request (in the signed-data field of the message token) then the recipient should compute and return to the MTS a signed version of the (unencrypted) message

content together with other delivery related parameters; however, the recipient cannot be forced to provide this proof. This signature, computed using the recipient's secret key, is returned to the message originator within the delivery report. The message originator then uses this signature to provide the required service(s).

Means are also provided within X.411 and X.413 for a pair of MHS entities to perform peer-entity-authentication prior to opening a connection for the exchange of messages. This protocol exchange again involves the use of tokens. For systems providing Mandatory Access Control services, all messages and entities can be assigned security labels. These labels can be tied to message contents by their inclusion in either the encrypted-data or signed-data fields of the message token (depending on whether or not the label itself is confidential). Inter-entity connections can also be assigned security-labels using the tokens exchanged in the peer-entity-authentication process.

5.5 Limitations of security in X.400-1988

We conclude by very briefly mentioning three important limitations of the current X.400 Recommendations. A more detailed discussion of these shortcomings can be found in [13].

First, proof of delivery to a UA is not available when an MS is used. Because of the way the protocols operate, the proof of delivery must be generated at the time the message is delivered by the MTS to the MTS-user. If this MTS-user is an MS, then it must generate and sign the delivery proof, and not the end user. The message originator then has no proof that the message was ever delivered to the recipient UA, only to the MS belonging to the recipient UA.

Second, proof of delivery by an MS is not possible if the message content is encrypted. The proof of delivery must be computed using the unencrypted message content, which will not be available to the MS (unless the MS is equipped with the UA's secret key).

Third, the specified form of token may allow the 'theft' of message content by third parties even when the content is encrypted. In certain circumstances this can be achieved by a third party replacing the signature on the token of an intercepted message by this third party's own signature. This arises because the signature on the token is computed after the secret data (in the encrypted-data field) is enciphered. The problem would not arise if the order of these two operations was reversed.

REFERENCES

[1] C.C.I.T.T. Recommendations X.400, X.401, X.408, X.409, X.410, X.411, X.420, X.430, C.C.I.T.T. VIIIth Plenary Assembly, October 1984.

[2] C.C.I.T.T. Recommendations X.400, X.402, X.407, X.411, X.413, X.419, X.420, Message handling systems, C.C.I.T.T. IXth Plenary Assembly, October 1988.

[3] C.C.I.T.T. Recommendations X.500, X.501, X.509, X.511, X.518, X.519, X.520, X.521, The Directory, C.C.I.T.T. IXth Plenary Assembly, October 1988.

[4] D. Coppersmith, Analysis of ISO/CCITT Document X.509 Annex D, preprint, IBM Thomas J Watson Research Center, 1989.

[4.5] J. Henshall and S. Shaw, OSI explained, Ellis Horwood (Chichester), 1988.

[5] ISO 7498-2, Information processing systems - Open systems interconnection - Reference Model - Part 2: Security Architecture, International Organization for Standardization, 1988.

[6] ISO 8372, Information processing - Modes of operation for a 64-bit block cipher algorithm, International Organization for Standardization, 1987.

[7] ISO/DIS 9594-1, 9594-2, 9594-3, 9594-4, 9594-5, 9594-6, 9594-7, 9594-8, Information Processing Systems - Open systems interconnection - The Directory, International Organization for Standardization, 1988.

[8] ISO/3rd DP 9796, Data cryptographic techniques - Digital signature scheme giving message recovery, International Organization for Standardization, 1989.

[9] ISO/DIS 9797, Data cryptographic techniques - Data integrity mechanism using a cryptographic check function employing an n-bit algorithm with truncation, International Organization for Standardization, 1988.

[10] ISO/DIS 9979, Data cryptographic techniques - procedures for the registration of cryptographic algorithms, International Organization for Standardization, 1988.

[11] ISO/DP 10116, Information processing - Modes of operation for an N-bit block cipher algorithm, International Organization for Standardization, 1988.

[12] ISO/2nd DP 10118, Information Technology - Data encryption - Hash-functions for digital signatures, International Organization for Standardization, 1989.

[13] C. Mitchell, D. Rush and M. Walker, 'CCITT/ISO standards for secure message handling', IEEE Journal on Selected Areas in Communications 7 (1989) 517-524.