Faculty of Computing and Information Technology

Department of Robotics and Digital Technology

Technical Report 94-10

Overview of Internet Protocol Security

Angus Mackinnon, Tony McGregor and James Breen

October 27, 1994

Enquiries:-

Technical Report Coordinator
Robotics and Digital Technology
Monash University
Clayton VIC 3168
Australia

tr.coord@rdt.monash.edu.au +61 3 905 3402

Contents

List of Tables

List of Figures

Abstract

This report gives an overview of the current research and trends in the area of TCP/IP security. The basic issues in network security and encryption techniques are presented as background material. The leading contenders in the field of secure internet protocols are then reviewed. The review includes the security architecture proposed in the draft standard for the next version of the internet protocol (IPv6). Issues in key management for the encryption and message digests employed in IP security architectures are presented with a brief overview of the ANSI X9.17 standard and the use of key management protocols in SNMP and Kerberos. Finally an overview of possible implementation scenarios for IP security are presented.

Keywords

Internet Protocol, Network Security, Cryptography

Chapter 1

Introduction

1.1 Overview of Network Security

Security can be defined as a method or system that protects a entity from undue loss. Applying this definition of security to the field of computer communications includes protection from loss that arises from manipulation of data or programs and the losses incurred from breaches of confidentiality.

The main contributing factors to an increased emphasis on computer security in recent years are the growth of computer networks, like the internet, that are accessed by an ever increasing number of users, the use of data networks to carry confidential or security sensitive information and the increasing ease of attack on a network due to constantly improving technology being within the financial reach of the average consumer[1]. It is a combination of the first two factors that provide the environment for security breaches. The third factor, the availability of the tools and technology to breach network security, being the mechanism by which security breaches occur.

In order to provide secure network communication the following criteria must be met[2]:

- Data communicated must remain confidential.
- Integrity of Data must be preserved.
- Access to data or network resources must be restricted to authourised users.

The threats that must be countered to in order to provide security in computer networks generally fall into one of the following categories. (This list is derived from [3]).

- *Masquerade:* An entity maliciously or otherwise successfully assumes the identity of another entity.
- *Illegal associations:* Where an entity (intruder) breaches the access or authentication policies of the system while initially satisfying the rules of the authentication or access control services.
- *Non-authorised access:* Where an entity breaches or violates the access control service.

- *Denial of service:* An entity is incorrectly or maliciously denied access to a network service or entity that it requires to function.
- *Repudiation:* A network entity falsely denies, either accidently or deliberately, that it received or sent a message, requested or performed an action.
- *Leakage of information:* Where loss of confidentiality occurs. The undesired exposure of information or loss of anonymity of a user or entity fit into this category.
- *Traffic analysis:* An unauthorised entity monitors network activity and gathers information about the nature of the traffic on the network. This information may include the source, destination and volume of traffic even if the information carried on the network cannot be analysed.
- *Invalid message sequencing:* This may occur if messages on a network are replayed, modified, re-ordered, suppressed or deleted maliciously or by some error.
- *Deduction of information:* The extraction or reconstruction of information from features present in messages by statistical or analytical means.
- *Illegal modification of programs:* The unauthorised and typically malicious modification or deletion of a program. Threats such as viruses and Trojan horses fall into this category.

In order to protect a network from these threats and meet the security criteria cited above, some general security services need to be provided. These services are defined in the International Telecommunication Union (ITU) Open System Interconnection (OSI) Security Architecture standard X.800[4]. The security services are:

- *Authentication service:* Provides assurance of the identity of the peer communicating entity.
- *Access Control service:* Provides protection from the unauthorised use of resources that can be accessed by OSI
- *Confidentiality service:* Ensures that data is protected from disclosure to an unauthorised entity.
- *Data Integrity:* Ensures the integrity of data by countering active threats to the system.
- *Non Repudiation:* This service can provide proof of origin and/or delivery of data.

Authentication

Authentication services are primarily concerned with the providing assurances about the identity of an entity. This means that when an entity claims to have a given identity the claim can be validated. The entity in the network situation could be an end user (person or application) or another system.

Access Control

Access Control is the method used to prevent unauthorised access to resources within the network in question. Access is a general term including any activity involving the disclosure, use, modification or deletion of data or system resources.

Confidentiality

This service prevents The disclosure or revealing of information contained in data to entities not rightfully entitled to such information. Methods such as data encryption can be used to provide such a service.

Data Integrity

Integrity services ensure that the data being transmitted and received is consistent with the security policy. That is, the data is unmodified in any way. This includes copying of data, data modification, addition or deletion.

Non-repudiation

This service provides proof of the integrity and origin of data transmitted across the network [5]. It achieves this by providing irrefutable evidence that a certain event took place. An event could be any transaction or interaction between two or more parties. Hence, non-repudiation provides a mechanism by which a disagreement over whether, or how, an event occurred can be resolved.

The policies and methods used to implement such security services in computer networks are the main focus of this report. In particular the report will concentrate on the provision of security at the network level with an emphasis on networks using the Internet Protocol (IP).

1.2 Open Systems Interconnection Security

While TCP/IP was not developed along the lines of ISO's Open Systems Interconnection (OSI) reference model there are parts of the OSI security recommendations that provide a good basis for general network security design. The security services definition (see section 1.1) and the recommendations in relation to the positioning of security services within a protocol stack are two sections of the OSI recommendations that are relevant to network security in general.

The International Telecommunications Union standard X.800[4] specifies that the Open Systems Interconnection (OSI) framework for network security allow for security to be implemented at a number of different layers in the OSI reference model. Table 1.1 shows the relationship between security services and OSI layers. From this table it can be seen that not all services can be provided at each of the layers. The network and transport layers are able to provide most of the security services outlined above except for non-repudiation. Due to the nature of a non-repudiation service, a typical system uses a third party to arbitrate in cases of repudiation, usually implemented as an application.

Service	Layer						
	1	2	3	4	5	6	7
Peer entity authentication	.	.	Y	Y	.	.	Y
Data origin authentication	.	.	Y	Y	.	.	Y
Access control Service	.	.	Y	Y	.	.	Y
Connection confidentiality	Y	Y	Y	Y	Y	Y	Y
Connectionless confidentiality	.	Y	Y	Y	.	Y	Y
Selective field confidentiality	.	.	.	.	.	Y	Y
Traffic flow confidentiality	Y	.	Y	.	.	Y	Y
Connection integrity with recovery	.	.	.	Y	.	.	Y
Connection integrity without recovery	.	.	Y	Y	.	.	Y
Selective field connection integrity	.	.	.	.	.	.	Y
Connectionless integrity	.	.	Y	Y	.	.	Y
Selective field connectionless integrity	.	.	.	.	.	.	Y
Non-repudiation, origin	.	.	.	.	.	.	Y
Non-repudiation, delivery	.	.	.	.	.	.	Y

Y Yes service is provided
. Service not provided

Table 1.1: Illustration of relationship between security services and OSI layers (taken from ITU X.800)

Annex C of the X.800 recommendation gives an overview of possible placement of encipherment within the OSI structure. The network layer is specified as being suitable for the protection of end to end system communications especially if encryption hardware is to be used. Elements of the security that are not provided by the network layer service can be implemented at another layer. In certain implementations security may be provided by an application, for example; privacy enhanced mail, as well as in a general form in the network layer.

The use of the network and transport layers to provide the bulk of the security services does appear to have merit based on the content of table 1.1. A generic security service can be supplied transparently to all upper layer users, independent of any security features an application layer program may implement. More security features can be provided at the application layer (layer 7) but provision of network security services at this level can be expensive and inefficient[6]. Application layer security is also more intrusive as applications must be written specifically to use the application layer security services.

This section only provides a brief overview of the OSI recommendations in regard to security services. For further information the reader is referred to [4] and for an in depth discussion of the OSI security architecture: [3].

Chapter 2

Encryption

2.1 Introduction

One of the main methods of ensuring data confidentiality in computer communication is to use encryption. Encryption essentially involves encoding a message so that the contents or meaning of the message is not obvious without decrypting the message. The system used to encrypt and decrypt a message is known as a *cryptosystem*. The cryptosystem converts the original form of the message known as *plain text* to its encrypted form; *ciphertext*.

Basic cryptosystems use a key within an encryption algorithm to encipher and decipher a message. This means that the encoded message depends on both the key and the contents of the plain text message. The use of a key in an encryption algorithm provides greater security because even if the encryption algorithm is known the key cannot be determined and hence the message cannot be decrypted. However, the key must be kept secret to avoid the message being decrypted by unauthourised parties.

There are a great number of encryption algorithms. The scope of this report is restricted to an overview of methods that are currently being considered for standards related to internet protocol security or have been implemented in common applications.

2.2 Symmetric Key Cryptosystems

In symmetric cryptosysytems the same key is used to encipher and decipher messages or data. The two communicating entities agree upon a key to be used for encrypting data in their communications. This key must not be disclosed to any other party so that the communications remain secure. Symmetric cryptosystems, the most notable being the *Data Encryption Standard (DES)*, appeared before their public key counterparts.

2.3 Public-Key Cryptosystems

In the mid 1970's Diffie and Hellman [7] proposed a cryptosystem that uses two keys, one of which does not need to be kept secret. This non secret key is called the *public* key and the secret key the *private* key. In this public key system the two keys form a complementary pair. One key is used for the encryption and the other for the decryption. The choice of using the private or public key for encryption depends on the mode in which the system in used. There are two modes of operation in a public key cyptosystem. These modes are known as the *encryption* mode and the *authentication* mode [8].

The encryption mode involves the use of a public key to encode the data/message so that the holder of the corresponding private key is the only one able to read the message. In the authentication mode the public key is used to decrypt a message encrypted using a given private key. If the message is successfully decrypted the recipient can be assured that the holder of the private key created the message. A public key system that operates in both these modes is said to be *reversible*. Public key systems that only operate in the authentication mode are said to be *irreversible*.

The main advantage of public key systems is that the public keys can be distributed without concern for their confidentiality. The resultant reduced overheads in key management due to the lower number of keys required to be kept in a secure network database is also an advantage.

The main disadvantage with public key cryptosystems is that the public key provides additional information about the encryption algorithm. The information provided by the public key could be used in an attack against the system. The protection against the breaking of the cipher is the premise that the mathematical problem presented by the encryption algorithm is inherently difficult to solve even with the information provided by the public key.

2.4 Common Encryption Protocols

2.4.1 Data Encryption Standard (DES)

The Data Encryption Standard (DES) is the current U.S. National Institute of Standards and Technology (NIST) standard for data encryption. It has been in the public arena for over fifteen years. In 1981 the DES became American National Standards Institute (ANSI) standard X3.92 *American National Standard Encryption Standard Data Encryption Algorithm*[5] after having been ratified as a Federal Information Processing Standard (FIPS) in 1977 (FIPS PUB 46).

The DES uses the Data Encryption Algorithm (DEA) which is a symmetric key cryptosystem. It employs a 56 bit key which works on 64 bit blocks of data. The algorithm applies an initial permutation to the data to be encrypted then subjects this to 16 rounds of calculations that are key dependent. Finally the result is adjusted using another permutation.

The DES has multiple modes of operation that use blocks or streams of previously encrypted data employing various algorithms to produce unique ciphertext in all cases:

- Electronic Code Book mode(ECB)
- Cipher Block Chaining mode (CBC)
- Cipher Feedback mode (CFB)
- Output Feedback mode (OFB)

The DES has been criticised because of its key size. It is thought to be too small for the DES to be secure. Despite this concern there still has not been a recorded instance of

DES encryption being broken [9]. This is particularly impressive considering the amount of time DES has been publicly available and the extensive research into methods to break the cipher during this time.

There are other methods that can be employed to improve the security provided by DES. One common method is to apply the DES encryption process to the data three times. The process involves an encryption, then a decryption and finally another encryption using the result of each step and a different key for the next encryption process. This method is known as *triple DES*. This process increases the effective key size of the cryptosystem. This increases the strength of the cryptosystem to a point where it is suggested that triple DES may be secure against all attacks [10].

2.4.2 RSA Algorithm

The RSA algorithm was developed by Rivest, Shamir and Adelman at MIT (hence its name). The RSA is a public key cryptosystem. The RSA algorithm is based on the premise that it is difficult to factor large prime numbers. The product of two large primes is used to generate the public and secret keys used in the cryptosystem.

The main drawbacks for the RSA algorithm are the fact that there are larger costs in processing than for the DES and the fact that the algorithm could be broken by factoring methods. Current factoring methods and technology limit the feasibility of cracking the RSA algorithm using this method. Extra security against this mode of attack can be implemented by adding extra bits to the key length. This exponentially increases the amount of calculations in the factoring process.

The RSA algorithm can complement DES by providing a secure mechanism for key distribution for a secure DES session. The message containing the DES key can be encrypted using RSA in the encryption mode and then distributed to the appropriate parties. The use of RSA for the key distribution for DES allows security to be maintained during key exchange. The use of DES over RSA for the bulk of the data communications is an advantage because DES has less processing overheads than RSA and therefore can allow higher data rates [10].

The RSA algorithm is patented by RSA Incorporated. It is in widespread commercial use both within the United States and elsewhere in the world.

2.5 Hash Functions and Message Digests

A hash function is a computation that uses a variable sized input and produces a fixed length result called a hash value[5]. Hash functions that have the property of being difficult to invert are known as message digest functions . The output of the message digest function is called the *message digest*. Message digests are used to represent a longer message, for example a document, in a compact form. The message digest produced by a message is unique to that message digest function. Message digests are used to provide a means of authentication. As message digests are an unique identifier of a message they can be used in time stamping. The message digest can be time stamped as a proxy for the message so that the contents of a message do not have to be disclosed to the time stamping service. The use of hash functions in this manner is often called a digital signature.

The security provided by a hash function depends on the following conditions. It must be infeasible to reverse the hash function and the hash value should be unique for all inputs.

The common hash functions in use or specified in emerging standards are the MD2, MD4, MD5 functions developed by RSA Incorporated and the secure hash standard (SHS) (see 2.7). "MD" stands for message digest. These hash functions produce an 128 bit output from a given input. MD2 is proposed in the privacy enhanced mail (PEM) standard being developed by the Internet Activities Board (IAB)[11]. MD4 is a weaker hash function than MD2[12]. MD5 is also not as strong as MD2 but it is a stronger hash function than MD4. However, MD4 has lower processing overheads than MD5 due to its simpler algorithm[13]. MD4 and MD5 have been placed in the public domain for evaluation and possible incorporation into future standards.

2.6 Secure Electronic Mail

One of the more common applications for encryption systems and hash functions is in the area of secure electronic mail. Three of the leading systems for providing secure electronic mail services are *Privacy-Enhanced Electronic Mail (PEM)*, *Pretty Good Privacy (PGP)* and *RIPEM*. All the systems are capable of sending authenticated or encrypted messages.

- *PEM* is the current Internet standard for secure electronic mail. It employs the DES in the CBC mode for the encrytpion of messages (a triple DES version is under development). RSA is used for the key encryption and distribution of key certificates. The message digest for message hashing is specified as MD5 but MD2 is used for hashing the key certificates issued by a trusted certifying authority. PEM has the most complex structure of the three schemes. It can provide support for a wider range of encryption and hash functions than either RIPEM or PGP. The key certification architecture using a hierarchy of trusted machines to validate keys is an added complexity that is not shared by PGP or RIPEM.

- *RIPEM* is much like PEM except that it only has some of the features of PEM. RIPEM uses DES CBC, MD5 and RSA for key distribution but does not provide key certificates to validate keys. This means the key's validity must be assessed on a case by case basis by the end user.

- *PGP* differs from the previous two schemes becaues it does not use DES for encryption. The encryption method used is IDEA. IDEA is an encryption algorithm that is similar in operation to DES but uses a 128 bit key. As IDEA is a new algorithm it has not been widely implemented due to the lack of extensive public testing. PGP uses MD5 as the message digest function of choice. The RSA algorithm is employed in the key distribution feature of PGP. PGP uses a certificate structure to validate keys during distribution. Unlike PEM, the certificates are not issued by a trusted certifying authority and hence the validity of the key must still be assessed by the end user. The use of PGP within the United States has been controversial due to the fact that it uses the patented RSA algorithm without permission. Some commercial versions of PGP have been released that use RSA under a licensing agreement.

2.7 Current Developments

The DES is subject to periodic review by NIST. The last review of the DES was in 1993. During the review NIST indicated that an alternative encryption algorithm to replace DES was being considered[14]. The United States' government is currently developing a set of public encryption standards for this purpose. This work is part of the Capstone project that is involves developing standards in the following areas:

- Key exchange protocol
- Hash function
- Digital signature algorithm
- Bulk data encryption algorithm

So far most discussion has centred around Skipjack, an encryption algorithm that is similar in style to DES but with increased strength. It uses an 80 bit key and twice the number of computational cycles (rounds) than DES uses. The United States government proposes to implement this algorithm using an hardware device referred to as the "Clipper" chip. Skipjack is facing strong opposition as the proposed standard requires that the United States government be the key escrow agent. There is also opposition because the details of the Skipjack algorithm are classified and there are doubts about its integrity due to the lack of testing in the public arena. Due to these concerns, raised by public and commercial interests in the United States, a review of the Skipjack proposal is being undertaken by the United States government. The use of the Clipper chip for computer communications now appears to be unlikely.

The Digital Signature Standard (DSS) is proposed for the Capstone project. The DSS specifies a Digital Signature Algorithm (DSA). This standard has been issued as a working draft by the American National Standards Institute (ANSI)[15]via NIST.

The hash function specified in Capstone is a United States Government standard known as the Secure Hash Standard (SHS) [16]. There has been no announcement on the key management protocol yet.

Chapter 3

Security for Internet Protocol Networks

3.1 Overview of Current Internet Protocol Security

During the development of the Internet Protocol (IP) the area of security for the protocol suite was not addressed. As a consequence there are a number of weaknesses in the security provided by IP[17]. The main security areas lacking in the internet protocol are explicit methods of providing data integrity, authentication and confidentiality. Recent attempts have been made to redress this.

The United States Department of Defence, which was a major driving force behind the development of IP, has produced some IP Security Options (IPSO) by introducing extensions to the IP protocol. These extensions were designed for use in U.S. military applications. This standard uses the security label of the IP services option to define the type of security option to be used.

The protocol types defined in the IPSO standard are the Department of Defence (DoD) Basic Security (type 130) and the DoD Extended Security (type 133)[18]. The basic security option provides for the classification of the data and hence the level of protection to be afforded to it. The protection rules to be applied to the data in order to maintain the appropriate security are also specified. The basic security option allows the transmission of data across a network using security labels as a form of routing control. Paths taken through the network must provide a level of security appropriate for the classification of the data. Datagrams can be validated by the intermediate and end systems in this scheme[18].

The extended security options enable extra security labelling information to be included in the IP header. The extended option format is flexible enough to allow other security information that is relevant to the intermediate systems to be included in the datagram. The extended option must be used in conjunction with the basic option.

The limitation of the DoD IP security extension is that it provides security by routing control. Packets are routed through trusted networks providing adequate security. This does not make it entirely useful in hostile environments. The draft standards containing the DoD IP security options acknowledge that the DoD security options are not suitable for commercial applications. Work is continuing on the development of the security options for the wider Internet but no security options draft standards are available for the current version of IP as yet.

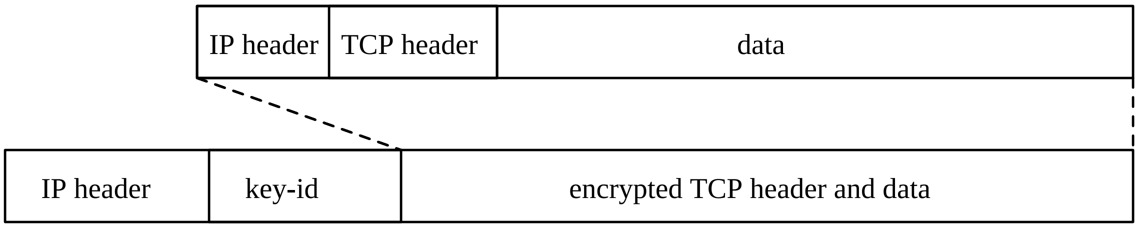


Figure 3.1: TCP segment encapsulation at the Transport level

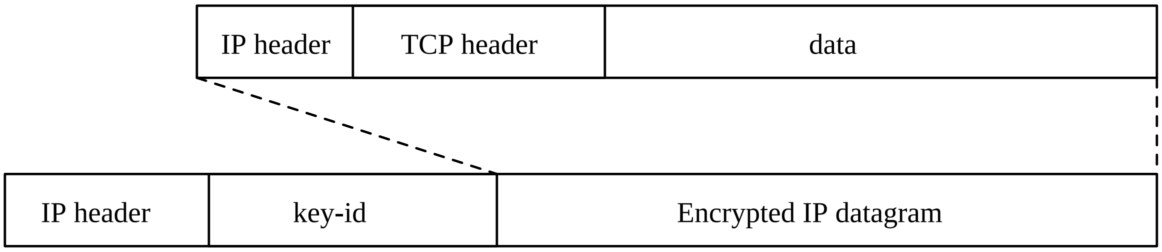


Figure 3.2: Encapsulation of an IP datagram at the network level

The U.S. Department of Defense has indicated that it will eventually move towards an OSI structure for its networks and hence OSI style security. It seems however that this migration will take a substantial amount of time to occur and IP is likely to remain the leading interworking protocol for a number of years[19].

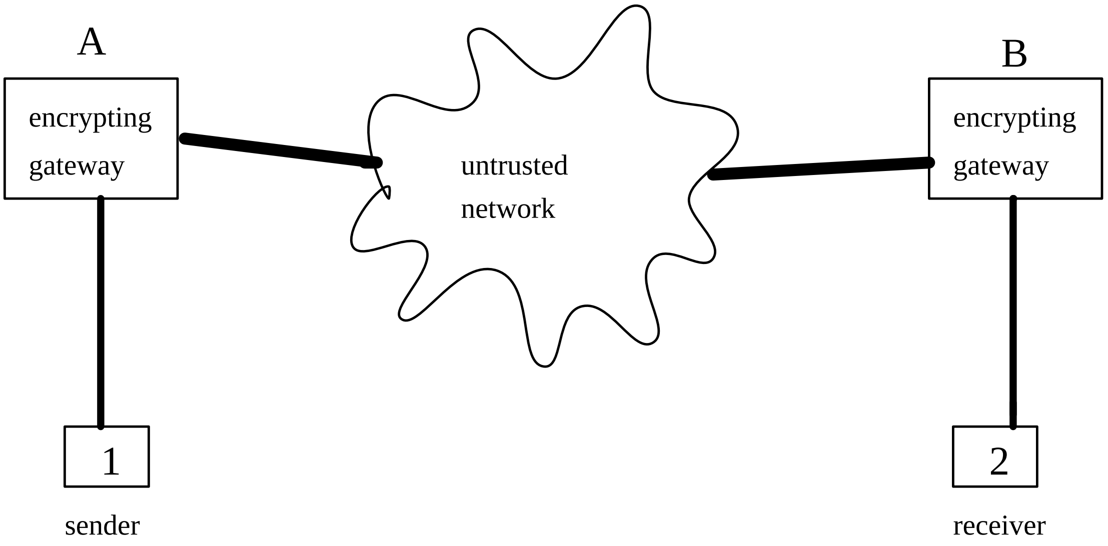
Currently there is a substantial amount of work being undertaken in the area of IP security, especially for the security architecture for the next generation of IP . The emerging standards for TCP/IP security and encryption have similarities with the International Standards Organisation (ISO) draft standards for encryption at the network layer: The *Network Layer Security Protocol (NLSP)*[20] and transport layer: The *Transport Layer Security Protocol (TLSP)*[21].

The basis of the security services outlined in the ISO standards is the encryption of the payload at a the transport or network layer and then encapsulating it within a new datagram. A key identifier is appended to the encrypted payload and the payload is given a new header for transmission. In the case of transport layer encryption in a TCP/IP network, the payload is the Transport Control Protocol (TCP) segment The TCP segment is encrypted and passed down to the network layer and an IP header and key identifier is attached to the segment to form a datagram. (see figure 3.1).

For the case of encryption at the network level, the entire IP datagram would be encrypted and a new IP header would be attached to the encrypted datagram (see figure 3.2). For a system that employed network gateways to provide the IP security service the new headers would contain gateway addresses as the source and destination addresses (see figure 3.3). This would provide users of the secure IP gateway service some protection against traffic analysis as the source and destination addresses remain hidden in the encapsulated datagram.

If the transmitting and receiving parties were on the same subnetwork it may be possible to omit the new IP header and use the key id and link layer addresses, such as ethernet addresses, to determine the correct recipient.

The security architectures proposed for the current and future versions of IP incorporate aspects of the network level encapsulation and encryption of TLSP/NLSP. An



Packets travelling from 1 to 2 are encrypted at the network level at A and the address of the gateway B is appended to the encrypted datagram. When the datgram arrives at B it is decrypted and the address in the decrypted datgram is used to forward the datagram to its destination 2.

Figure 3.3: Encapsulation/encryption of IP traffic by a routing gateway

overview of the proposed security architectures for the current and next generation of IP is presented in the remainder of the chapter.

These architectures do not specify any particular security mechanism. Instead they support mechanisms that selection of security mechanisms and their parameters.

3.2 The swIPe IP Security Protocol

The contents of this overview are taken from an internet draft describing the swIPe IP security protocol for the current version of IP, version 4 (IPv4). SwIPe was presented by Ioannidis and Blaze[22] as a means of providing IP security. SwIPe aims to provide data integrity/source authentication and confidentiality services or any combination of the two services.

The swIPe protocol uses encapsulation of IPv4 datagrams in a similar fashion to that specified for NLSP. The structure of a swIPe datagram is shown in figure 3.4.

The structure of the swIPe header (figure 3.5) contains a packet type indicator which specifies the type of security datagram being transmitted. The packet types include plain (no security), authenticated, encrypted or both authenticated and encrypted. There are also reserved values that are used for key and policy management applications.

The datagrams are processed on the basis of the local security policy which is defined by a policy identifier in the header of the swIPe datagram. This is similar to the security association identifier (SAID) proposed for the next generation of IP (see section 3.4.1). The receiving party uses the policy identifier to access information regarding the method of processing the datagram from a local database or table. This information includes the

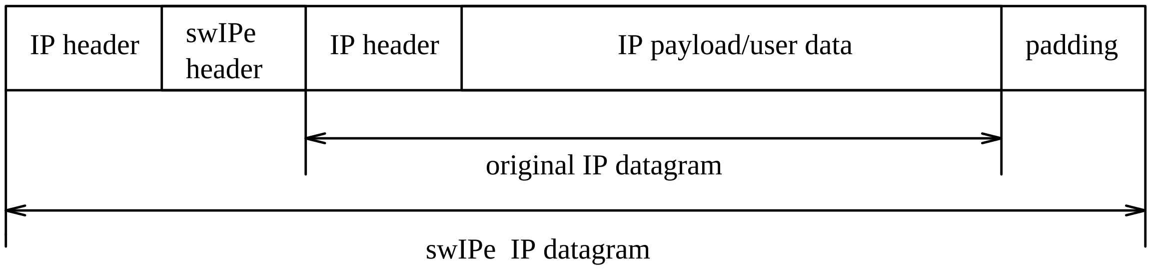


Figure 3.4: Structure of the swIPe datagram

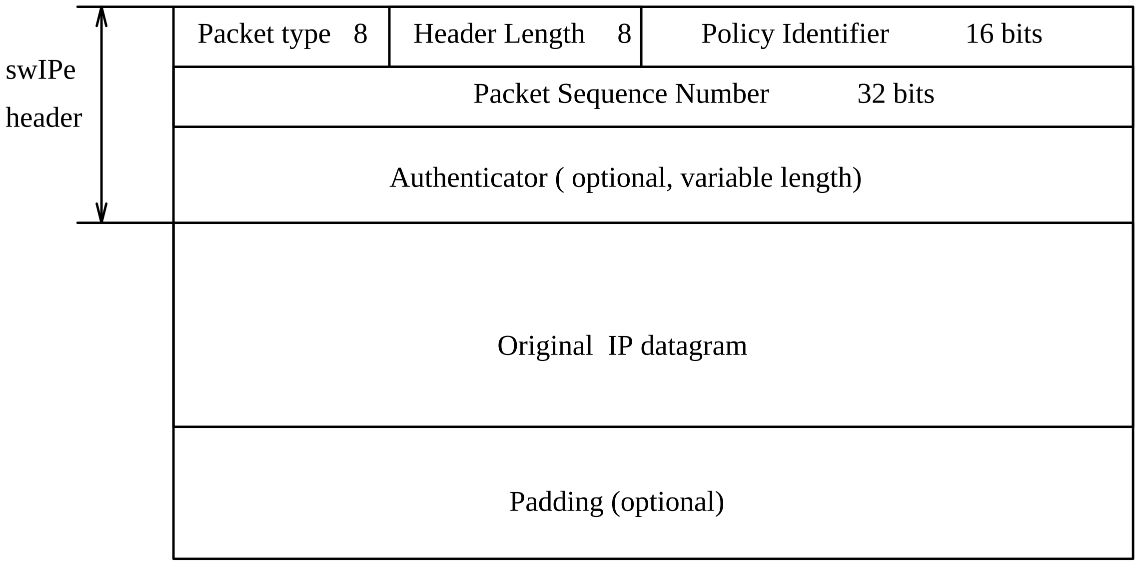


Figure 3.5: Detail of the structure of the swIPe header and datagram

key and encryption algorithm. The authentication process involves the calculation of some authenticator (message digest) and appending it to the encapsulated packet. The authentication calculation is performed across the original datagram and the SwIPe header. This provides authentication and data integrity for both the encapsulated datagram and the swIPe Header. Any encryption desired is then performed on the datagram. For encryption systems like DES padding may be required to bring the length of the original datagram up to a integral block length so it can be processed. A new IP header is then attached and the datagram is transmitted. The receiver decrypts and then authenticates the datagram using the security information contained in the swIPe header. Any errors are recorded according to the requirements of the local security policy.

The swIPe protocol is temporarily using the protocol number of 94 to identify it in its current implementations. This protocol number specifies that the next higher level protocol type is IP-within-IP[23]. SwIPe, like most of the other proposals for IP security does not specify any key management or control protocols to be used and therefore allows flexibility in the key management selection and implementation.

3.3 TUBA Network Layer Security for the Internet Protocol

TUBA stands for "TCP and UDP with Bigger Addresses". TUBA was put forward as a contender for the next generation of the internet protocol (IPng). The TUBA security architecture is designed to provide security in a dual protocol stack system. The two protocols it is specifically designed for are the ISO OSI connectionless network protocol (CLNP) and the current version of the internet protocol. The security scheme is called the Internetwork Layer Security Protocol (I-NLSP). I-NLSP also borrows much of its basic structure from the ISO NLSP style encapsulation.

The encapsulation of the datagram provides any combination of confidentiality and authentication services. There are two possible levels at which the encapsulation may take place. In the case of end-system to end-system communication the encryption/authentication of just the TCP/UDP data is all that is required for secure communications. This scheme is similar to the method used in TLSP (see figure 3.1). It has the advantage of reducing the overheads from processing a whole IP datagram to those associated with processing the smaller UDP/TCP segment. Where intermediate systems are involved, encryption/authentication of the entire IP datagram at the network level is used. This is preferred because of the associated problems of fragmenting datagrams before encapsulation when encrypting at the TCP/UDP level[24]. These two options are known as the TCP/UDP encapsulation protocol and the Datagram Forwarding Protocol (DFP) respectively.

The TUBA architecture employs security attributes which identify the key and encryption/authentication methods that are to be used for the communication between parties[1]. The security attribute is appended to the encapsulated datagram in the same fashion as for NLSP/TLSP after the authentication and/or encryption are completed.

The specification for the key management, encryption system or the message digest

[1]This is like the policy identifier in swIPe and security attribute identifier (SAID) in IPv6

function is outside the scope of the TUBA security architecture proposal.[2]

The future development of TUBA is uncertain due to the fact that it was not chosen as the draft standard for IPv6. With much of the detail of the preferred key management, encryption and authentication methods still to be presented it will be interesting to see the path TUBA takes.

The proposed security architecture for IPv6 is outlined in the following section.

3.4 Internet Protocol Version 6 (IPv6) Security Architecture

The next generation of the internet protocol will be version number six (IPv6). This new version will eventually replace the current version of the internet protocol (version 4). The IPv6 specification has been published as an internet draft by the Internet Activities Board (IAB). The latest version of the draft was published in October 1994[26]. There were a number of proposals put forward for IPv6. The contender that has become the draft standard was formerly known as the Simple Internet Protocol Plus (SIPP). When SIPP was accepted for IPv6 the name was changed from SIPP to IPv6.

The development of SIPP came about from the work in three separate Internet Engineering Task Force (IETF) working groups: The IP address encapsulation (IPAE), the Simple Internet Protocol (SIP) and the "P" Internet Protocol (PIP) groups. The many parallels in the work undertaken by these groups lead to their almagamation . Aspects of the work of each group were incorporated into one proposal which became known as the Simple Internet Protocol Plus[27].

This new version of the internet protocol incorporates many new features such as extended addressing (128 bits) and routing capabilities, simplified header structures, quality of service capability (QOS) and security features. The header structure proposed in the internet draft has improved support for extra protocol options . Amongst these options is support for security. The architecture and mechanisms proposed for IPv6 security are outlined in the following section. For more information on IPv6 the reader is referred to the IPv6 internet drafts [26] and [25] which present a detailed overview of the protocol.

3.4.1 Security Mechanisms

IPv6 proposes two security mechanisms. These mechanisms are:

- The authentication header(AH)[28]. The AH provides a check for data integrity and authentication of contents but does not encrypt the data in any way. Hence there is no confidentiality provided by this mechanism.
- Encapsulating Security Payload (ESP)[29] The ESP can provide authentication and data integrity but its main purpose is to provide data confidentiality. The level of security provided by ESP is dependent on the encryption used.

[2]The Internet draft for I-NLSP for TUBA[24] mentions support for DES CBC has a possible option for encryption. The Diffie Hellman exponential key distribution algorithm is also mentioned as a possible inclusion in the key management protocol.

Next Payload	Length	RESERVED
Security Association Identifier (SAID)		
Authentication Data (variable number of 64 bit words)		
Authentication data continued.		

Figure 3.6: Authentication Header Format

The Authentication Header (AH)

The Authentication Header (AH) is placed between the "Hop-by-Hop" header and the "end-to-end" header. These other headers are used to route the packet from source to destination. The authentication header is designed to be transparent to systems that can't process it.

The AH can be used to provide intermediate authentication at each hop. However, in the case of a symmetric cryptosystem the intermediate nodes along the route would require a secret key to process the message. This could compromise network security because possession of the key could allow the intermediate routers to forge or modify transmissions if they were subverted in some manner. This problem is avoided if a public key cyptosystem was to be used for the authentication. In a public key cryptosystem implementation the routers would use a public key for their processing. Knowledge of the private key would be required by the intervening routers to modify traffic.

The AH structure is shown in figure 3.6.

The main fields associated with security in the AH are

- *Security Association Identifier (SAID)*. The SAID is a 32 bit value that identifies the security association for the IP packet. The SAID implies a key for an authentication algorithm, the algorithm itself, the mode used for the algorithm as well as the block size (where applicable) and the vectors used to initialize the algorithm (where applicable).

  The key management implementation uses the SAID to look up a table containing the parameters outlined above. Typically the SAID value is uni-directional. (i.e. the communicating parties use different SAID identifiers for their transmissions).

- *Authentication Data*. This is a variable length field of 64 bit multiples. The field contains the authentication data. If the authentication data is not a multiple of 64 bits then the receiver must ignore the unused trailing bits.

The other fields in the AH include the Next Payload descriptor which specifies the contents of the datagram following the AH and the Length field (8 bits) which specifies the length of the authentication data in 64 bit double words. The current draft standard[28] specifies that the reserved bits must be set to zero when the datagram is transmitted.

The authentication data is normally generated using an appropriate message digest algorithm such as MD5[13]. (Support of keyed MD5 is mandatory in the current IAB draft standard as is manual distribution of keys). The key used is fed to the algorithm first followed by the fields of the datagram that will not be altered during the transmission across the network. The fields that do vary are also included in the calculation but are replaced with the dummy value zero.

Not all the fields in the IPv6 Hop by Hop Options header are always included in the authentication data calculation. The inclusion of an option in the calculations is specified by a bit within the Hop by Hop Options header[25]. This allows datagrams with future versions of Hop by Hop Options to be processed using an existing AH implementation by excluding options from the authentication data calculations where necessary.[3] The basic constraint for calculating the authentication data values is that the calculation must be across the whole packet that is seen by the receiver.

Upon receiving a datagram the authentication data is calculated by the receiver from the datagram and compared with the value with that was contained in the authentication data field in the received datagram. If the values match, the datagram is authentic. When a datagram is received that fails the authentication it can be discarded and the failure is audited according to the security policies of the network.

Encapsulating Security Payload (ESP)

The IPv6 Encapsulating Security Payload (ESP) is designed to provide data authentication, confidentiality and data integrity[29]. ESP works by taking either an entire IPv6 packet or the higher layer data portion and encapsulating it. The encapsulation involves encryption of the datagram or higher layer user data segment and then adding a new IPv6 header to it.

The fact that the ESP can be either the entire packet or just the higher layer data enables it to be used in two ways. The first use is for end to end security by encrypting just the higher layer user data. This selective encapsulation also has the advantage that if it is not necessary or desired to encapsulate the entire IPv6 datagram, the overheads of processing the packet are reduced by processing just the user data. In the second mode the entire datagram can be encrypted for transmission across an untrusted section of the network and then decrypted and forwarded to its destination at the far side. These two modes are similar to TLSP and NLSP in operation.

ESP is designed to work with multi-cast and uni-cast traffic. However, the multicast case may be difficult to implement as the key management issues involved with dynamic multicast groups have not been resolved by current published mechanisms for key management[30].

The ESP syntax allows the the ESP to appear anywhere in the packet after the IPv6 header. The payload is made up of the ESP header and the encrypted data. The structure of an IPv6 datagram using ESP is shown in figure 3.7.

The IPv6 header(s) are conventional headers as defined in the IPv6 (SIPP) internet draft[25].

The fields within the packet are as follows:

[3] For more information on the IPv6 Hop by Hop Options header the reader is referred to [25].

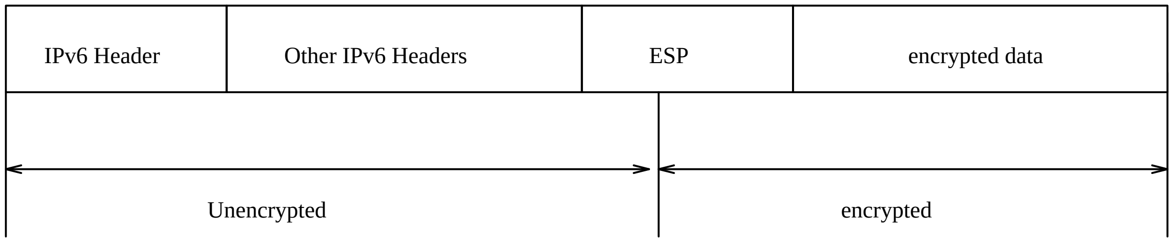


Figure 3.7: Diagram of a secure IPv6 datagram

- *Security Association Identifier (SAID).* This is the same as defined for the Authentication Header see section 3.4.1. This field is unencrypted.
- *Synchronisation field.* This is an optional field. The synchronisation field contains cryptographic synchronisation/initialisation information. This is used for synchronising block oriented encryption algorithms[20]. The presence of this field is determined by the security association implied by the SAID value. The absence or presence of the synchronisation field is remains constant for all datagrams using this SAID value. This field is not encrypted, though its contents are not typically meaningful to the casual observer.
- *Sequence Number.* This field is the encrypted sequence number of the packet. The presence of the sequence number is designed to prevent instances of transmission replay.
- *Hop by Hop Header.* The standard Hop by Hop Header as defined in the IPv6 specification[25]. This is used to provide information about the rest of the packet.
- *Encrypted data.* The encrypted data can be either the user data or an entire encapsulated datagram.

  It is important to note that all routing information in the encapsulated datagram should be preserved and the receiver should use the decrypted information rather than the appended headers to determine routing information. This use of the internal encrypted routing headers helps prevent attacks that use routing information to breach security.

The overall process of transmitting and receiving a packet using ESP involves the sender encrypting the original IPv6 datagram or user data and appending the appropriate headers so that the encrypted datagram (or data) becomes the payload of this new datagram. The receiver extracts the ESP from the datagram and decrypts it and processes the contents like any other IPv6 datagram. If there is a failure in the decryption process or an invalid encrypted packet is received, the error is audited in an appropriate fashion.

While the encryption method used is independent to the ESP protocol, the draft standard does require that any implementation supports the Data Encryption Standard (DES) algorithm operating in Cipher-Block-Chaining (CBC) mode.

One important aspect of the security architecture that has not been specified yet is the area of key management[30]. The only option that is specified as mandatory in the IPv6 draft standard is the ability to manually distribute keys. There is also the strong suggestion that all implementations should support the IPv6 key management protocol when it is standardised. This presents a problem for those implementations that may come into service before the standard is complete. The simplest solution would be to extend the existing implementation to include the specified key management features. The draft standard also notes that in order for the IPv6 security mechanisms to be used successfully on a large scale the choice of key management protocol is a very important issue.

The IPv6 security architecture is designed to sit with all other IPv6 option implementations. It is also designed to coexist with other security measures such as link encryption and application level security. One of the driving forces in the IPv6 specification is the backwards compatibility with IPv4. The implementation of IPv6 is seen in terms of a software upgrade from IPv4[27]. There is some doubt as to IPv6 compatibility with IPv4 security due to the fact that the working group responsible for developing a security for IPv4 have not yet published a standard for the security architecture[30]. If the IPv4 standard and the IPv6 standards in this area coincide the IPv6 standard will be updated where appropriate.

3.5 Key Management and Other Issues

In all the internet drafts and related descriptions of IP security the issue of key management has been left open. The IPv6 key management has been influenced by the features of the Simple Network Management Protocol, especially version 2, that provide for key management [31] but little detail is given about possible strategies. This lack of definition allows implementors flexibility but also increases the probability that software developed will be proprietary. Possible key management schemes are discussed in chapter 4.

In all the schemes presented the authentication headers and the encryption process will increase the overheads of transmitting data due to the extra processing required. Authentication of datagrams should have a lower impact on efficiency than encryption due to the fact computing message authenticators requires less processing than for encryption in most cases.

In each of the strategies the mechanisms for authentication and encryption can be employed independently or combined when there is a need for extra security. This allows local security policy to dictate the level of security required.

None of the mechanisms outlined provides security from end-to-end traffic analysis. Tunnelling of encrypted packets is one method that could be employed to achieve limited protection from traffic analysis (a possible implementation using this scheme is presented in section 3.1). Other techniques, such as encrypting data streams on specific links (bulk link encryption), are not within the scope of the architectures presented.

There are no widespread implementations of any of the schemes. Work is progressing on the IPv6 implementation but the standard is still being developed. TUBA, as stated before, faces an uncertain future as it was competing with SIPP for acceptance for IPv6. Swipe has been implemented in an Unix environment[22] and appears to be designed specifically to address IPv4 security but like TUBA was not integrated into the architecture proposed for IPv6 and so may have a limited lifetime.

Chapter 4

Key Management

4.1 An Overview of Key Management

One of the main issues in providing network security is the management of keys used in cryptosystems, message digest or general hash functions. The area of key management can be broken down into five areas[32]. These are:

- User registration
- Key generation
- Key distribution
- Key storage
- Key deletion

The handling of the key distribution needs careful consideration in implementing any security system. Protection of the key from disclosure during distribution is paramount. Various schemes have been proposed, including:

- All systems are individually configured manually. This has limited application because it is expensive in large networks.
- Needham and Schroeder[33] propose a key management algorithm using a centralised key distribution system. This algorithm is implemented in the Kerberos authentication system developed at MIT. Kerberos is discussed in section 4.2.
- Diffie and Hellman[7] propose an exponential key distribution algorithm that does not use a centralised key distribution scheme. However the early versions of this system are purported to be susceptible to active attacks. Modifications to this algorithm have rectified these problems and a modified scheme is used in the key certificate feature of privacy enhanced mail. One of the main drawbacks with the Diffie-Hellman key distribution method is it uses far more exchanges to set up a secure session than other possible schemes.
- Another option to provide this key certificate is to add a public key to the Domain Name System (DNS) or have the public key available through a UDP service.

From this list it can be seen that key management schemes fall into two basic categories. The first category use a central key management centre that controls the use and distribution of keys. The second category involve techniques that use direct exchanges between parties. The Needham-Schroeder method falls into the first category and the Diffie-Hellman algorithm, the second.

The ANSI X9.17 standard covers key management for secure communications between financial institutions. The standard covers all aspects of key management for systems using the DES. In this standard key distribution is performed using a hierarchical system where a manually distributed key encryption key is used to provide a mechanism that can be used to distribute data keys to be used for transmissions between communicating parties[34]. This distribution of data keys using the key encryption key is fully automated. The system has two tiers. A third tier can be added by manually distributing a key encrypting key which is used to encrypt a second key encrypting key that in turn is used for encrypting the communicating parties' data keys during distribution (see figure 4.1). The use of the third tier makes the task of attacking the system more complex and as a consequence the key distribution system is more secure. ANSI X9.17 specifies the encryption algorithm for encryption of keys is specified as triple DES.

One important part of the standard is the specification of the control protocol and messages used for inter party key management communications and exchanges between the key management stations and the communicating parties. ANSI X9.17 is suitable for use in systems that use central key management stations or in point to point key distribution provided that one station can perform the required key management functions[5].

The use of unique key encryption keys for the distribution of data keys by the key management server to each of its clients means that this system does not scale well. Large numbers of keys must be generated and stored for large networks using this scheme. One solution to this problem is to use exchanges between domain servers to obtain data keys to be used for inter domain communications by their local clients. A variation of this mechanism is implemented in the Kerberos network authentication service architecture.

There has been recent interest in network management and authentication schemes that have architectures that support security and key management. Two examples of schemes that involve the implementation of key management for security services, Kerberos and the Simple Network Management Protocol (SNMP), are reviewed in the remainder of the chapter.

4.2 The Kerberos Network Authentication Service

The Kerberos Network Authentication Service was developed at MIT to provide a means of authenticating users. It was developed for project Athena. Kerberos is currently an internet draft standard and is undergoing public review[35].

The main parts of the Kerberos system are the authentication protocol, the key distribution protocol and the single sign-on protocol. The authentication protocol is essentially an exchange where an encrypted message containing details about one user is passed to another user for verification. The message contains the sender's address and a time stamp amongst other information. The message is encrypted using a key known only to the sender and receiver. If the message passes the receiver's authentication checks (the message decodes correctly and the time stamp is valid) then communication can commence.

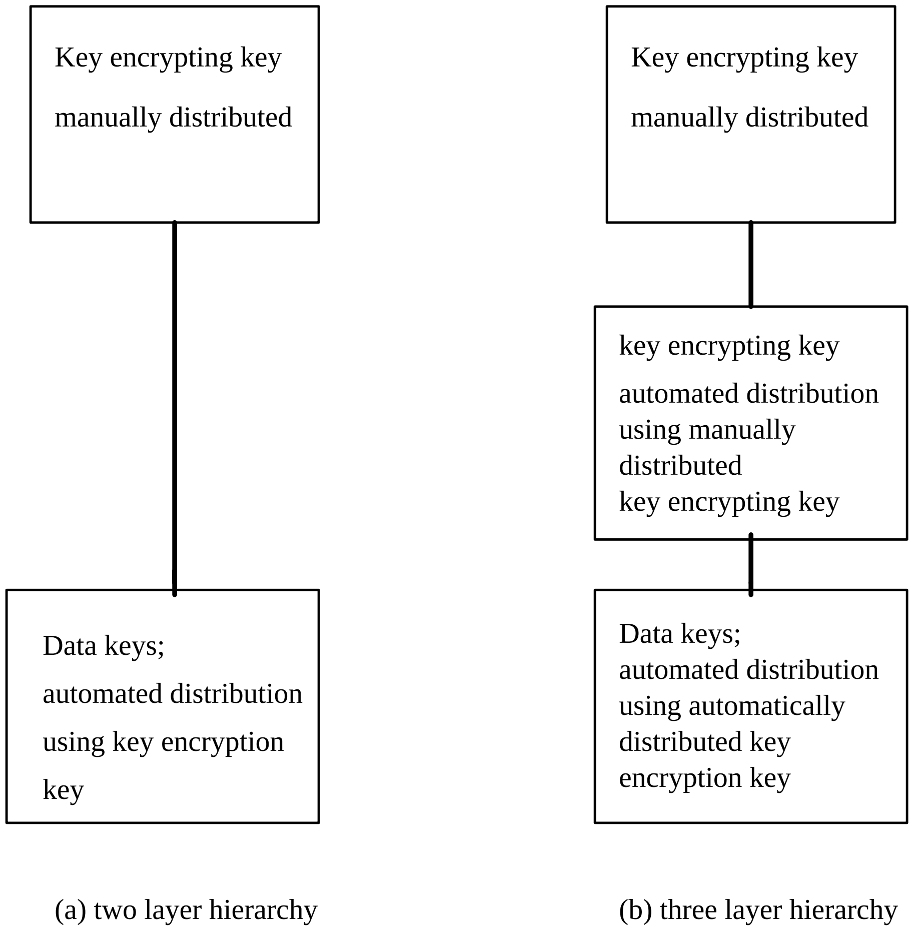


Figure 4.1: ANSI X9.17 Key Distribution Hierarchy

The secret key used for the authentication exchange can be used to provide privacy for the following communications. The authentication procedure assumes that both parties already possess their shared secret key. The key distribution protocol endeavours to provide such a basis.

The key distribution protocol employs a key distribution centre (KDC) to provide the secret keys for communication sessions between "principals" within its area of control, known as a realm. The key distribution centre must be a trusted third party. It shares its own secret keys with all the principals called master keys. When a client wishes to attach to a server it requests the appropriate key from the KDC. The KDC issues a key to the client and a ticket to the server containing information about the key to be used for the session with the client. These exchanges use unique master keys for encrypting the exchanges between the KDC and the client or the server.

Kerberos also has a second protocol that is used for user authentication called the single sign-on protocol. The single sign-on protocol provides a method of converting human user passwords into a stronger keys. This is done by a trusted third party called the Authentication Server (AS). The AS verifies the password and issues a strong key to be used when the client requests a key and ticket to be issued for a client-server session. This procedure is similar to the key distribution protocol outlined previously.

The Kerberos specification allows the use of any cryptosystem or hash function in its implementation. However, the formats to be used for the DES and RSA encryption algorithms are specified. The MD4 and MD5 hash functions are also specified as being the desired options for authentication.

Kerberos is designed to operate across organisational and/or network boundaries[35]. The hierarchical structure allows the trusted servers to share keys with other servers controlling other realms. This means that machines in different realms can be authenticated across realms provided there is a path between the realms constructed from links between authentication servers[36]. Hence the Kerberos system could span many domains provided there are trusted and authenticated connections between them.

There are some limitations in the design of Kerberos. Kerberos was specifically designed to operate on single user workstations where the problems associated with providing security on multiple user machines are not an issue. In many Unix systems this is not the case. Kerberos is not yet fully integrated with the Unix architecture and providing adequate security for keys in a multi-user Unix environment has not been completely investigated. Kerberos also uses synchronised clocks for time stamping which implies the use of an authenticated time service to provide the clock. The provision of a secure authenticated time service is an undesirable overhead. Providing adequate security for a trusted ticket granting/authentication server may be difficult in some cases. The security of the trusted machine is paramount because if it is violated the whole network is vulnerable.

Kerberos does have the advantage of providing a transparent service to the user and it does provide extra security that is often lacking in Unix networks. Kerberos is currently the only readily available (and free) authentication option available for Unix based systems[36]. This is likely to change as research in the area of Unix network security continues to progress.

4.3 Simple Network Management Protocol (SNMP) Security for Version 2 of SNMP (SNMPv2)

One management protocol that is suitable for the management of IP networks is the Simple Network Management Protocol (SNMP). SNMP has been ratified by the IAB as an internet standard[37]. The second version of the SNMP standard (SNMPv2) is under development and has been published as a draft standard.

SNMP was developed within the Internet community via the Internet controlling bodies. SNMP was developed as a temporary measure until the Common Management Information Protocol (CMIP) was developed and then ratified by the appropriate OSI bodies[38]. Due to the time taken for CMIP to progress towards standardisation SNMP has become widely accepted in the market place as the preferred network management protocol[39]. SNMP was designed to provide a frame work that can be used to monitor and control network elements[40]. SNMP usually is implemented running above the User Datagram Protocol (UDP) and IP. Security aspects were a secondary consideration in SNMP's development. This has been addressed in some degree by the second version of SNMP (SNMPv2). It will include extra management options, including security and will be incompatible with the older version.

The security architecture for SNMPv2 aims to provide data integrity, authentication and confidentiality at the network management level[39]. The implementation of security in the SNMPv2 architecture is designed to have little impact on the reliability and performance of the network management as possible[41].

The protocols specified for SNMPv2 must be implemented in such away that data integrity and authentication are provided. Confidentiality for network management information is a service that can only be provided if data integrity and authentication services already exist.

Data integrity is provided by using a message digest algorithm on the data where a key is prefixed to the message before the message digest is computed. This provides authentication of both the data and the source. The receiver decodes the message using the key and can therefore authenticate the message. A time stamping function is also included to prevent the replaying of messages but there is no provision within SNMPv2 draft standard to counter loss or suppression of messages.

Like the other Internet security proposals discussed, the confidentiality service is provided by a symmetric encryption algorithm. The draft standard for SNMPv2 security aims to create a set of protocols that are independent of the exact algorithms used to provide the authentication and confidentiality. The proposed standard does recommend the use of the Data Encryption Standard in Cipher block Chaining Mode for confidentiality. For general authentication and data integrity services support for the MD5 message digest algorithm is recommended[31].

The distribution of keys and general management of the SNMPv2 security is undertaken by a management station. The management station is responsible for the generation and distribution of keys for the network. The management station is also responsible for the synchronisation of clocks in each station for time stamping purposes. This part of the architecture is similar to Kerberos (see section 4.2) and therefore poses the same problem of the provision of an authenticated time service. Key distribution can be achieved easily in SNMPv2 if a public key cryptosystem is used. This allows initial key distribution to be performed without disclosure of the key. The draft standard does not specify a method to use that is suitable for distribution of an initial key in a symmetric key cryptosystem. In symmetric cryptosystems the initial key distribution may have to be performed manually. If the system already has keys in use then distribution of new keys can be performed using the current key to encrypt the new key for distribution[31]. The other option is to use a exponential key distribution system based on the Diffe-Hellman algorithm (see section 4.1) to exchange information about the new key.

4.4 Discussion

Parts of the key management architecture of Kerberos and SNMPv2 may prove useful in the implementation of a key management scheme for secure IP. Both Kerberos and SMPv2 use a central server for key management. This has the benefit of integrating all the management services in one machine and allows the generation and distribution of keys to be provided from one secure location. The main concern with using a single management station for this purpose is that if the security of the management station is breached then the security of the whole network will be compromised. SNMPv2 style key management may be more useful because it is designed from the network management perspective whereas schemes like Kerberos are influenced by the needs of distributed systems security especially user authentication. The main factor in choosing between the key management aspects of the Kerberos and SNMPv2 is that the security structure of IPv6 was influenced by elements of the SNMPv2 security architecture. For a system that implements an IPv6 style of security service the SNMPv2 approach to key management may be more appropriate than other methods.

The use of a direct exchange mechanism in key management does present another useful alternative. The parties involved can exchange keys without a third party being involved. However, this requires all communicating parties to have the ability to generate and manage their keys. A big issue is that this mechanism does not provide a means for authenticating the identity of the communicating parties. Typically authentication must be performed by a third party that is trusted with this task. Hence a management service handling authentication must be provided. This is usually provided by a management station and introduces the security risk associated with a central management station.

Therefore it appears that for key management there must be a central service that can perform all aspects of key management and authentication services. For many networks, providing adequate security for the machine performing these tasks is not difficult and so becomes an attractive option as management of the network can be centralised.

The systems using the key management schemes presented within this chapter are only a representative example of user authentication or general security services that are being implemented in computer networks. Kerberos and SNMP were presented to give an insight into the implementation of key management services using a central key management server.

Chapter 5

Implementation Issues

5.1 The Future - IPv6 Implementation

The real driving force towards the migration from IPv4 to IPv6 is the limited address space provided by IPv4. As the IPv4 address space is filled users will be forced to employ a method like the larger addressing capabilities of IPv6 to extend the Internet's address space.

The move to IPv6 is not likely to be a universal upgrade. Individual sites are likely to upgrade as the need arises[42]. These initial upgrades will result in routers that are capable of handling IPv6 traffic as well as the IPv4 traffic that may still be routed through a given network.

This means that in order for a secure network service to be provided, routers that handle IPv6 security must be available across the sections of the Internet that require this security service. This does not exclude the tunnelling of secure IPv6 datagrams through IPv4 only sections of networks.

The time frame for the move from IPv4 to IPv6 is not clear. It is not likely to be in the short term. This means that for IP security to be viable for the present compatibility with the current version of IP is a major factor in the development of any security system.

5.2 Implementation Options

Due to the need to retain compatability with IPv4, for at least the near future, any scheme chosen must allow secure IP to pass transparently through the network carrying it. This means the retention of IPv4 packet header structure. As the current trend in IP security is for encapsulation of datagrams in the style of NLSP or TLSP, it would appear that a scheme employing either the mechanisms and some of the architecture of IPv6 or another IP security scheme such as swIPe would be an appropriate choice.

SwIPe has the fact that it has been implemented [22] in its favour. IPv6 security, though still in the development stages, has the advantage that its security architecture will become the standard in the future. Using the IPv6 security architecture means that the security architecture developed for IPv4 security should be able to be incorporated in future IPv6 implementations provided there are no major changes to the IPv6 architecture during the standardisation phase. This factor suggests that an IPv6 security architecture style implementation for secure IP would have a longer lifespan and tilts the balance in its favour.

The use of IPv6 style security architecture would involve the use of DES in the CBC mode and the MD5 message digest algorithm. Key management would have to be at least be able to be performed manually, with the key management employing features of the ANSI X9.17 standard, SNMP or Kerberos key management protocols. The fact that none of the proposals for IP security specify the key management protocols gives an indication of the concern there is about the stability and security of the existing key management protocols. With the continued development of systems such as SNMP and Kerberos these concerns are likely to ease.

The location of the security service provider is also a major issue. Using a gateway or a firewall to provide the security services for IP traffic passing between a trusted network to an outside untrusted network is one option[43]. This centralises the security services and in the case of the firewall also provides security by filtering traffic and mediating such services as electronic mail, news and external logins[44]. The main concern with such an implementation is that security could be lost or at least violated if the firewall or gateway is subverted in some manner.

The ease of implementation and protection afforded in a firewall/gateway scheme make it an attractive option for attaching networks that have little existing security to an untrusted network. Hosts on a network that already implement their own security will not benefit from such a scheme but hosts with weak security benefit from being behind the firewall.

The other option is to provide IP security on a host by host basis within a network. Depending on the structure of the network and the local security policies governing the network, like the need for local network security to be provided, this may be desirable. Kerberos and SMNP security management was designed for this type of scheme.

For general protection from threats from an outside network, the firewall strategy appears to have the simplest implementation and management. A possible implementation is presented in the next section.

5.3 Overview of a Firewall Gateway Style Implementation

The firewall gateway is essentially a routing gateway that also provides extra security features. The term firewall used here to cover any router or gateway that supplies security services to a network. The proposed implementation could be placed in an ordinary routing gateway or possibly a stand alone configuration. The location of the IP security service in the network architecture is the main feature of the scheme. The proposed structure for the scheme is shown in 5.1

The parts of the firewalls providing the IP security services across the untrusted network would negotiate the appropriate security and management parameters for the provision of the secure IP service. In this implementation the security service provided would be transparent to the users of the end networks A and B. This would allow existing applications including those that use their own security mechanisms to continue to operate without reconfiguration. Protection from traffic analysis can be provided by the gateway as discussed in section 3.1.

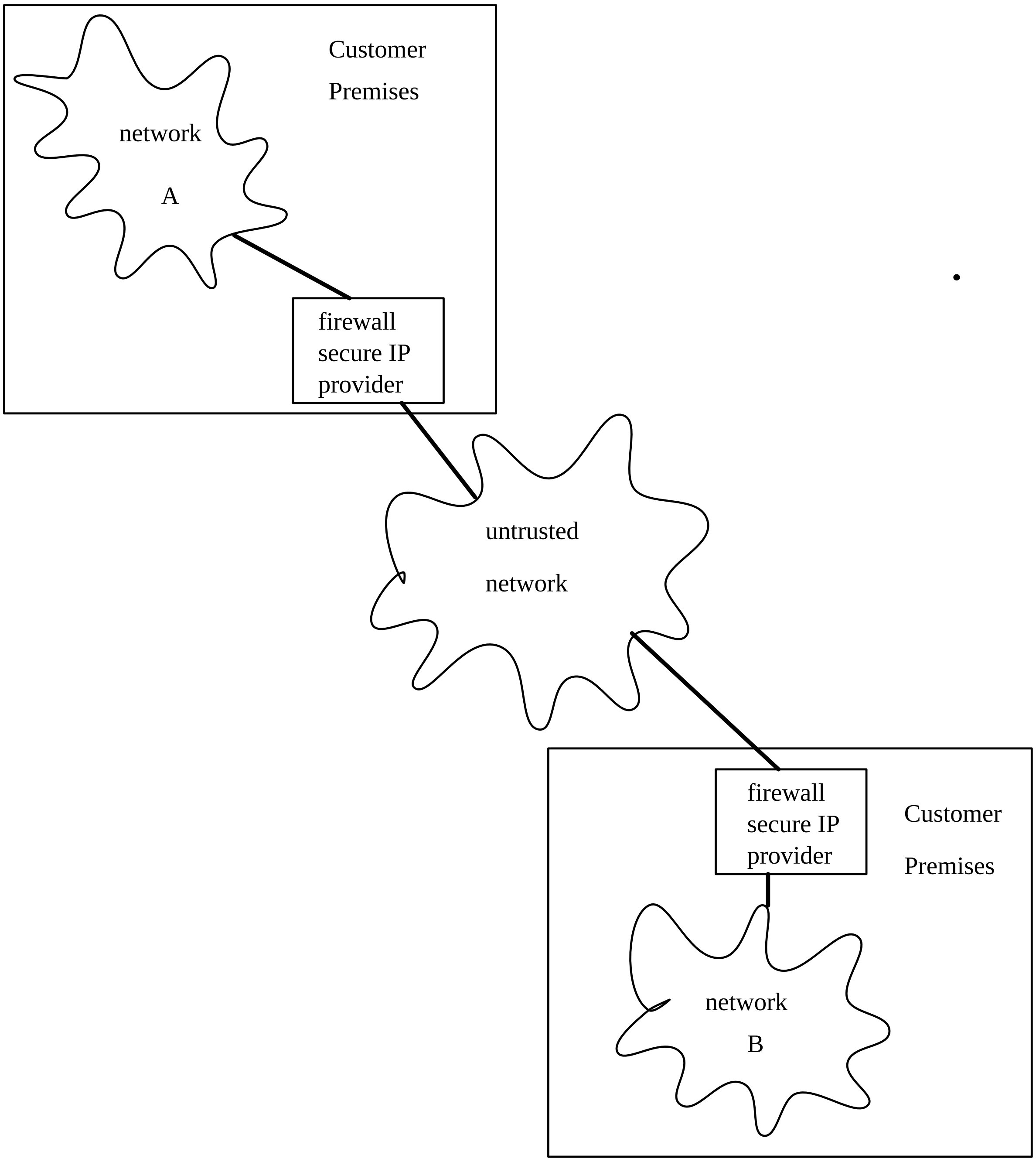


Figure 5.1: IP security provided across untrusted network by firewall gateways

An appropriate management scheme such as Kerberos or SNMP could be implemented to provide the required key management and end user authentication. The management scheme could cater for connections across the wider untrusted network as well as enforcing local security policy. The security management engine could possibly reside on another machine inside the trusted network to provide it with extra security from attacks that may be directed at the firewall from the untrusted network.

In conclusion it appears that the firewall implementation of an IP security service is an attractive option due to the simplicity of its architecture, ease of implementation or integration and its centralised management.

Bibliography

[1] W. Ford, *Computer Communication Security. Principles, Standard Protocols and Techniques*, p. 1. PTR Prentice Hall, 1994.

[2] M. Devargas, *Network Security*, p. 5. NCC Blackwell, 1993.

[3] S. Muftic, A. Patel, P. Sanders, R. Colon, J. HeijnSdijk, and U. Pulkkinen, *Security Architecture for Open Distributed Systems.* Wiley, 1993.

[4] International Telecommunication Union, *Security Architecture for Open Systems Interconnection (OSI)For CCITT Appliciations. Recommendation X.800*, 1991. Also ISO/IEC 7498-2.

[5] D. Longley, M. Shain, and W. Caelli, *Information Security. Dictionary of Concepts, Standards and Terms.* M Stockton Press, 1992.

[6] W. R. Cheswick and S. M. Bellovin, *Firewalls and Internet Security, Repelling the Wiley Hacker*, p. 229. Addison Wesley, 1994.

[7] W. Diffie and M. Hellman, "New Directions in Cryptography," *IEEE Transactions on Information Theory*, no. 6, p. 644, 1976.

[8] W. Ford, *Computer Communication Security. Principles, Standard Protocols and Techniques*, pp. 72–73. PTR Prentice Hall, 1994.

[9] D. E. Denning, "The Data Encrytpion Standard. Fifteen Years of Public Scrutiny.," in *Proceedings of the Sixth Annual Computer Applications Conference*, pp. x–xv, IEEE Computer Society Press, December 1990.

[10] P. Fahn, "Answers to Frequently Asked Questions About Today's Cryptography," FAQ, RSA Laboratories, September 1993.

[11] S. Crocker, N. freed, J. Galvin, and S. Murphy, "PEM security services and MIME," Internet Draft, Internet Activites Board, JULY 1994. Available from ftp:munnari.oz.au /internet-drafts/draft-ietf-pem-mime-06.txt.Z.

[12] R. L. Rivest, "The MD4 Message-Digest Algorithm," Request For Comments (RFC) 1320, Internet Activities Board, April 1992.

[13] R. L. Rivest, "The MD5 Message-Digest Algorithm," Request For Comments (RFC) 1321, Internet Activities Board, April 1992.

[14] W. Ford, *Computer Communication Security. Principles, Standard Protocols and Techniques*, p. 71. PTR Prentice Hall, 1994.

[15] American National Standards Institute (ANSI), *Working Draft: American National Standard X9.30-199X: Public Key Cryptography Using Irreversible Algorithms for the Financial Services Industry: Part 1: The Digital Signature Algorithm (DSA)*, March 1993.

[16] National Institute of Standards and Technology (NIST), *FIPS Publication 180: Secure Hash Standard (SHS)*, May 1993.

[17] S. M. Bellovin, "Security Problems in the TCP/IP Protocol Suite," *Computer Communication Review*, vol. 19, p. 32, April 1989.

[18] S. Kent, "U.S. Department of Defence Security Options for the Internet Protocol," Request For Comments (RFC) 1108, Internet Activites Board, November 1991.

[19] W. Stallings, *Handbook of Computer Communications Standards. Department of Defense (DOD) Protocol Standards. Volume 3.* MacMillan, 1987.

[20] ISO/IEC JTC1/SC6, *ISO DIS 11577, Network Layer Security Protocol*, November 1992.

[21] ISO/IEC JTC1/SC6, *ISO DIS 10736, Transport Layer Security Protocol*, October 1991.

[22] J. Ioannidis and M. Blaze, "The swIPe Security Protocol," Internet Draft, Internet Activites Board, December 1993.

[23] J. Reynolds and J.Postel, "Assigned Numbers," Request For Comments (RFC) 1340, Internet Activities Board, July 1992.

[24] K. R. Glenn, "Integrated Network Layer Security Protocol," Internet Draft, Internet Activites Board, May 1994.

[25] S. Deering, "Simple Internet Protocol Plus (SIPP) Specfication (128-bit address version)," Internet Draft, Internet Activites Board, July 1994. Available from ftp:munnari.oz.au /internet-drafts/draft-ietf-sipp-spec-01.txt.

[26] R. M. Hinden, "IP Next Generation Overview," Internet Draft, Internet Activites Board, October 1994. Available from ftp:munnari.oz.au /internet-drafts/draft-hinden-ipng-overview-00.txt.Z.

[27] R. M. Hinden, "Simple Internet Protocol (SIPP) Plus White Paper," Internet Draft, IETF SIPP Working Group, February 1994. Available from ftp:munnari.oz.au /internet-drafts/draft-ietf-sipp-whitepaper-00.txt.Z.

[28] R. Atkinson, "IPv6 authentication header," Internet Draft, Internet Activites Board, August 1994. Available from ftp:munnari.oz.au /internet-drafts/draft-ietf-sipp-ap-04.txt.Z.

[29] R. Atkinson, "SIPP encapsulating security payload (ESP)," Internet Draft, Internet Activites Board, April 1994. Available from ftp:munnari.oz.au /internet-drafts/draft-ietf-sipp-espi-02.txt.Z.

[30] R. Atkinson, "IPv6 security architecture," Internet Draft, Internet Activites Board, September 1994. Available from ftp:munnari.oz.au /internet-drafts/draft-ietf-sipp-sa-03.txt.Z.

[31] J. Galvin and K. McCloghrie, "Security Protocols for version 2 of the Simple Network Management Protocol (SNMPv2)," Request For Comments (RFC) 1446, Internet Activites Board, April 1993.

[32] W. Fumy, "Key management," in *Computer Security and Industrial Cryptography - State of the Art and Evolution* (B. Preneel, R. Govaerts, and J. Vandewalle, eds.), Springer-Verlag, 1991.

[33] R. Needham and M. Schroeder, "Using encrytpion for authentication in large networks," *Communications of the ACM*, pp. 993–999, December 1978.

[34] W. Ford, *Computer Communication Security. Principles, Standard Protocols and Techniques*, pp. 249–251. PTR Prentice Hall, 1994.

[35] J. Kohl, "The Kerberos Network Authentication Service (V5)," Internet Draft, Internet Activites Board, April 1993.

[36] B. De Decker, "Unix security and kerberos," in *Computer Security and Industrial Cryptography - State of the Art and Evolution* (B. Preneel, R. Govaerts, and J. Vandewalle, eds.), Springer-Verlag, 1991.

[37] J. Case, M.Fedor, M. Schoffstall, and J. Davin, "A Simple Network Management Protocol (SNMP)," Request For Comments (RFC) 1157, Internet Activites Board, May 1990.

[38] M. Devargas, *Network Security*, p. 63. NCC Blackwell, 1993.

[39] M. A. Miller, *Internetworking: A Guide to Network Comminucations. LAN to LAN; LAN to WAN*. M and T Books, 1991.

[40] J. Galvin and K. McCloghrie, "Administrative Model for version 2 of the Simple Network Management Protocol (SNMPv2)," Request For Comments (RFC) 1446, Internet Activites Board, April 1993.

[41] S. Waldbusser, "Overview of SNMPv2 Simplified Security Conventions," Internet Draft, Internet Activites Board, March 1994.

[42] W. A. Simpson, "IPv6 Deployment," Internet Draft, Internet Activites Board, September 1994.

[43] R. Braden, D. Clark, S. Crocker, and C. Huitema, "Report of IAB Workshop on Security in the Internet Architecture February 8-10, 1994," Request For Comments (RFC) 1636, Internet Activites Board, June 1994.

[44] W. R. Cheswick and S. M. Bellovin, *Firewalls and Internet Security, Repelling the Wiley Hacker*, ch. 3. Addison Wesley, 1994.