Chapter 7

Network Security

7.1 Types of Attack

In the context of communications across a network, the following attacks can be identified:

1. Disclosure: Release of message contents to any person or process not possessing the appropriate cryptographic key.

2. Traffic analysis: Discovery of the pattern of traffic between parties. In a connection-oriented application, the frequency and duration of connections could be determined. In either a connection-oriented or connectionless environment, the number and length of messages between parties could be determined.

3. Masquerade: Insertion of messages into the network from a fraudulent source. This includes the creation of messages by an opponent that are purported to come from an authorized entity. Also included are fraudulent acknowledgments of message receipt or nonreceipt by someone other than the message recipient.

4. Content Modification: Changes to the contents of a message, including insertion, deletion, transposition, or modification.

5. Sequence modification: Any modification to a sequence of messages between parties, including insertion, deletion, and reordering.

6. Timing modification: Delay or replay of messages. In a connection-orientated application, an entire session or sequence of messages could be a replay of some previous valid session, or individual messages in the sequence could be delayed or replayed.

7. Repudiation: Denial of receipt of message by destination or denial of transmission of message by source.

7.2 Security Model

For all kinds of security protocols, there are some common issues that we need to consider which means that with some variations, all the security protocols take a packet from appropriate layer and create a new packet which is authenticated and encrypted. First, we need to create MAC, then we need to encrypt message and probably the MAC. The common structure of security protocol is given below:

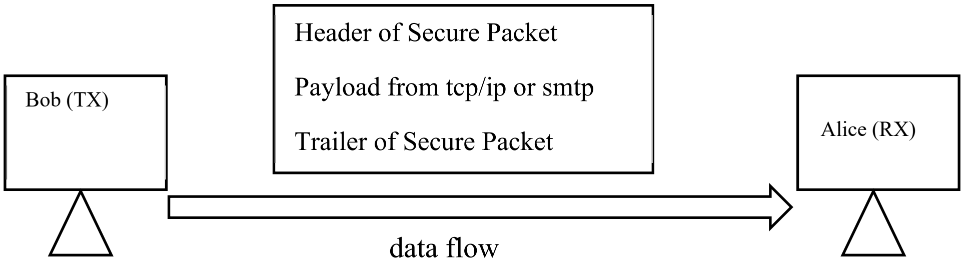
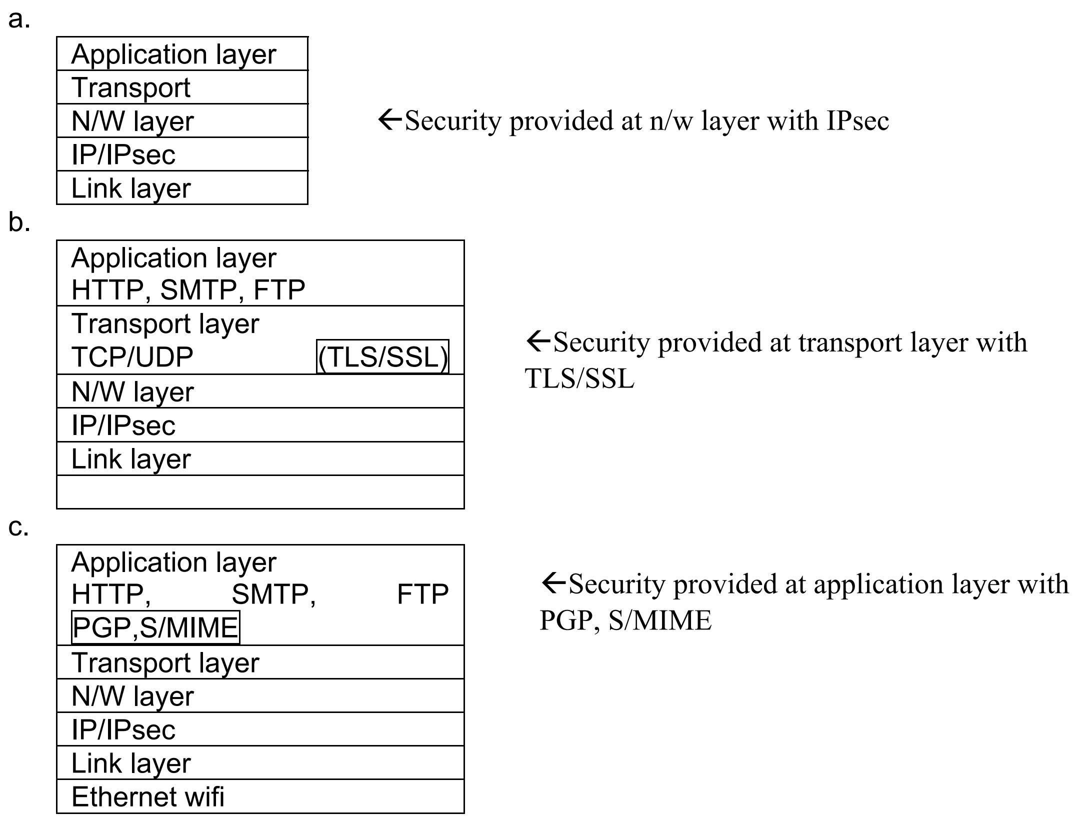


Fig. General Security Model

There are three fundamental aspect of providing information security in the internet applications.

1. Authentication
2. Integrity
3. Key management

Information security may be provided at different layers in the internet communication protocols. In network layer, we can use IPsec, for session layer SSL or TLS may be used and we can also embed security in the application layer itself using PGP, S/MIME etc. as well as shown in the figure below.



a.

Application layer
Transport
N/W layer
IP/IPsec
Link layer

←Security provided at n/w layer with IPsec

b.

Application layer
HTTP, SMTP, FTP
Transport layer
TCP/UDP (TLS/SSL)
N/W layer
IP/IPsec
Link layer

←Security provided at transport layer with TLS/SSL

c.

Application layer
HTTP, SMTP, FTP
PGP,S/MIME
Transport layer
N/W layer
IP/IPsec
Link layer
Ethernet wifi

←Security provided at application layer with PGP, S/MIME

Fig. Confidentiality and Authentication for information security at three different layers

7.3 Email Security (PGP)

PGP is used for signing, encrypting, and decrypting texts, e-mails, files, directories, and whole disk partitions and to increase the security of e-mail communications. Phil Zimmermann developed **PGP** in 1991. However, it is built as open PGP. It has now become an open-source standard described in RFC 4880.

PGP is widely used for protecting data in long term storage. It is designed to create authenticated message and confidential emails. The position of PGP in TCP/IP protocol is shown in the fig below:

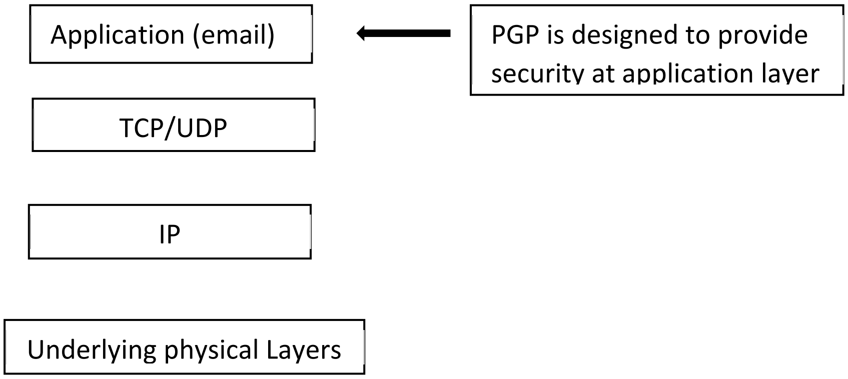

Application (email)
PGP is designed to provide security at application layer
TCP/UDP
IP
Underlying physical Layers

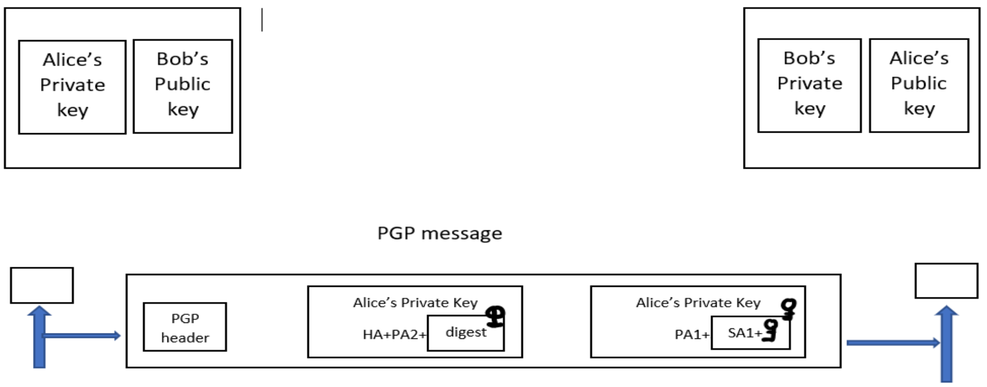
fig. position of PGP in TCP/IP

PGP can provide several services based on the requirements of the user. An email can use one or more of these services:

1. Plaintext: simplest case to send email in plaintext (no services)
2. Message authentication: probably next environment is to let the sender sign the message so that it can be authentic message which can be verified by the receiver.
3. Compression: a furthermore improvement is to compress the message and digest to make a packet more compact. This has no security benefits but eases the traffic.]
4. Confidentiality: in email we can achieve confidentiality by using symmetric key encryption with one time session key.

   PGP may use CAST-128, IDEA or 3DES with CAST-128 being the default choice for block cipher algorithm. 128-bit encryption key called the session key is generated for each mail separately and encrypted with receiver's public key using RSA, Diffie-Hellman or El-Gamal may also be used.

5. Code conversion: PGP uses Radix-64 conversion for those characters not defined in ASCII set. Each character to be sent (after encryption) is converted into Radix-64 code.
6. Segmentation: PGP allows segmentation after code conversion to make each transmitted unit to the uniform size by the underlying email protocol.



PA1→Public Key Algorithm 1(for encrypting session key)

PA2→Public Key Algorithm 2(for encrypting digest)

SA→Symmetric Key Algorithm identification (for encrypting message and digest)

HA→Hash Algorithm identification (for creating digest)

<u>Sender Side</u>

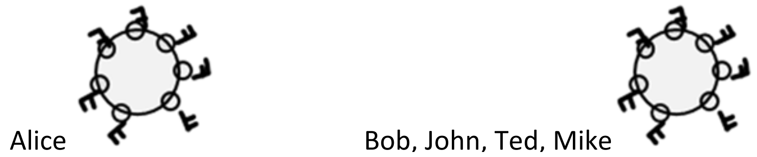
1. Sender creates session key for symmetric encryption and decryption and concatenate with identity at algorithm which will use this key. The result is encrypted with receiver’s public key. This consists of 3 pieces of information.
    a. Session key
    b. Symmetric key algorithm to be used later
    c. Asymmetric key algorithm used for this part
2. a) Sender authenticate message using public key signature algorithm and encrypt with sender’s private key. This part of the message contains signature and two extra pieces of information (encryption algorithm and hash algorithm identifier)
    b) Sender concatenates these three pieces of information created above with message and encrypt the whole thing using session key created in step 1.
3. Sender combines the result of step 1 & 2 and send them to the receiver after adding appropriate header. The algorithm used in PGP is given below. The new algorithms may be added continuously.

Algorithm	ID	Description
Public Key	1	RSA (encryption & signing)
	2	RSA (encryption only)
	3	RSA (signing only)
	17	DSS (signing)
Hash Algorithm	1	MD5
	2	SHA-1
	3	RIPE-MD
Encryption	0	No encryption
	1	IDEA
	2	3-DES
	9	AES

Key Ring:

Sender needs to message a key ring incase of sending message to many people. Sender's key ring of public key consists of keys belonging to each person with whom the sender needs to correspond. In addition, sender needs private/public key ring for changing pair of keys from time to time and correspond with different group of people. Sender may wish to use a different key pair for different group. Therefore, each user needs to have two sets of rings.

1. A ring of private/public key &
2. A ring of public key of other people.



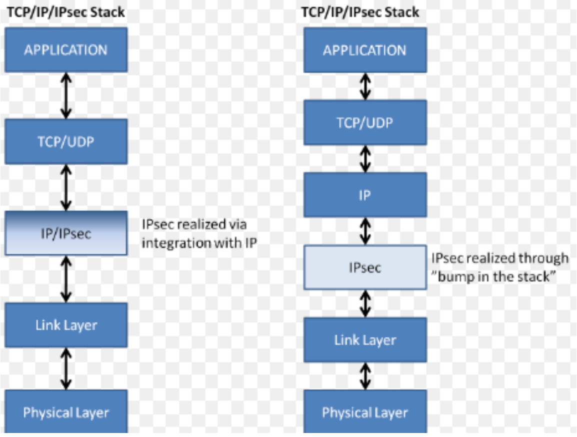
Each person in ring can keep more than one public key for each other since everyone can have more than one public key. Two cases may arise:

1. The person
   a. may need to send a msg to one person in the community
   b. uses receiver's public key to encrypt newly created message
   c. encrypts the msg and sign the session key
2. The person receives a msg from another in the community
   a. Uses private key to decrypt session key
   b. Uses session key to decrypt msg and digest
   c. Uses public key to verify digest

<u>7.4 IPSec (IP Security)</u>

IPSec is a collection of protocols designed by IETF (International Engineering Task Force) to provide security for a packet at the n/w level.

It helps to create authenticated and confidential packets for i/p as shown in the figure below:



IPSec operates on two models:

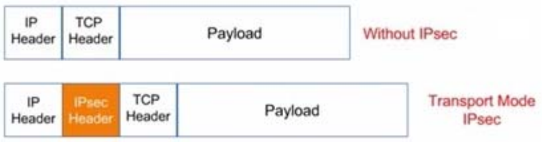
1. Transport mode
2. Tunnel Mode

Transport mode

IPSec protects what is delivered from the transport layer to the n/w layer. In other words, transport mode protects n/w layer payload, the payload to be encapsulated in the n/w layer.

It doesn't protect IP header, it only protects data in transport mode coming from transport layer.

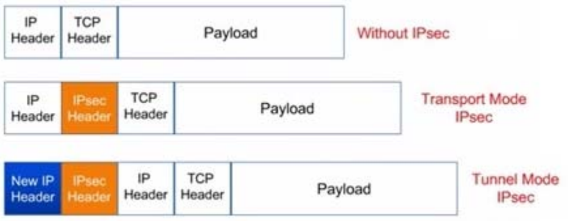
Transport mode is the regular mode for packets to travel from a source to destination except for the fact that the two ends must carry security check on the packets on the info contained in the authentication header.



Tunnel mode

IPSec protects the entire IP packet. It takes entire IP packet including IP header and applies IPSec method to the entire packet and then adds a new next IP header.

Source and destination end points may or may not have the ability and resource to carry out the security checks on the packets.



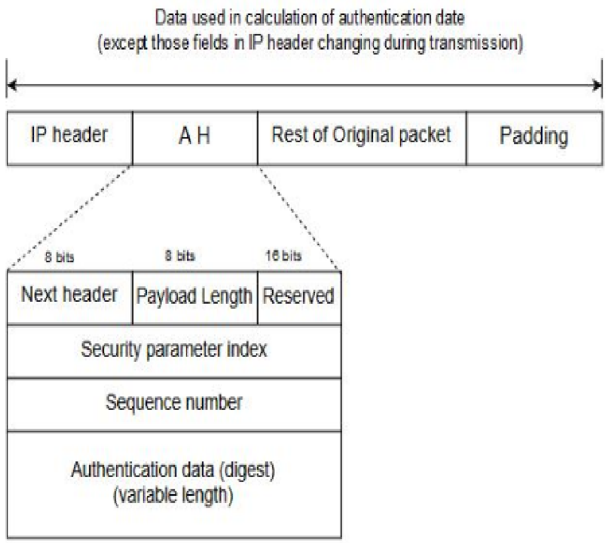
Security Protocols

The various security protocols used in IPSec are as follows:

1. Authentication Header (AH) protocol:

It is designed to authenticate the source host to ensure the integrity of payload carried in IP packet.

The protocol uses a hash function and a symmetric key to create a message digest. The digest is inserted into AH. The AH is then placed in appropriate location based on mode (transport or tunnel).



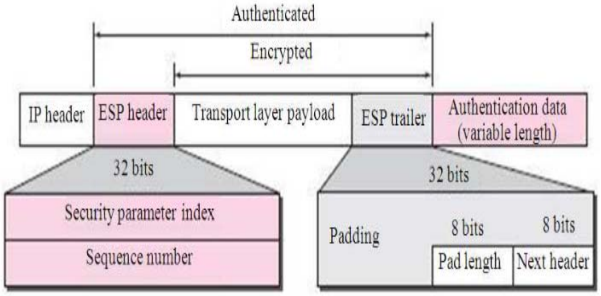
Authentication Header (AH) protocol

The addition of AH follows these steps:

1. An AH is added to the payload with authentication data fields set to zero.
2. Padding may be added to make the total length even for a particular hashing algorithm.
3. Hashing is based on the total packets. However only those fields that do not change during the transmission are included in calculation of digest (authentication code data).
4. Authentication Data are inserted into the AH.
5. The IP header is added after the value of the protocol field is changed to 51.

2. Encapsulating Security Payload (ESP)

The AH protocol doesn't provide privacy only authentication and data integrity.



IPSec later defined an alternative that include privacy called ESP. ESP adds header and trailer.

ESP payload follows the following steps:

1. ESP trailer is loaded into the payload.

2. Payload and the trailer are encrypted.

3. ESP header is added.

4. ESP header, trailer and payload are used to create authentication later.

5. The authentication data are added to the end of ESP which is changed to 50.

7.5. SECURE SOCKET LAYER(SSL)

A transport layer security provides end to end security service for application that uses a reliable transport layer protocol such as TCP.

When customer shops online following security services are desired:

1. Entity Authentication
2. Message Integrity
3. Confidentiality

Two protocols dominant today for providing security at transport layer are SSL & TLS.

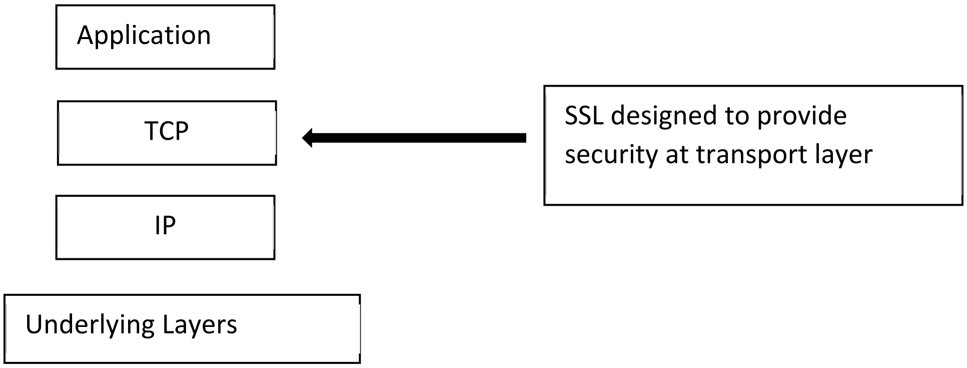


Fig.Location of SSL

SSL Services

It is designed to provide security and compression services to data from application layer. Typically, SSL can receive data from any application layer protocol but usually from HTTP. The data received from application are compressed (optional) signed and encrypted the data is then passed on. The data received from application layer such as TCP. SSL provides services on data received from application layer such as:

1. Fragmentation: firstly, SSL divides data into blocks of 2^{14} bytes.
2. Compression: each fragment of data compressed by using lossless compression negotiated between transmitter and receiver. This service is optional.
3. Message Integrity: to preserve integrity of data, SSL uses keyed hash function to create a MAC.
4. Confidentiality: to provide confidentiality, the original data and MAC are encrypted using symmetric key cryptography.
5. Framing: a header is added to the encrypted payload. The payload is then passed to a reliable Tansport Layer protocol.

7.6. SECURE ELECTRONIC TRANSACTION

(SET) is an open encryption and security specification designed to protect credit card transaction on the Internet. The current version, SETv1, emerged from a call for security standards by MasterCard and Visa in February 1996. a wide range of companies were involved in developing the initial specification, including IBM, Microsoft, Netscape. RSA, Terisa, and VeriSing. Beginning in 1996, there have been numerous tests of the concept, and by 1998 the first wave of SET – compliant products was available.

Key Features of SET:

♦ Confidentiality of information: cardholder account and payment information is secured as it travels across the network. An interesting and important feature of SET is that it prevents the merchant from learning the cardholder's credit card number; this is only provided to the issuing bank. Conventional encryption by DES is used to provide confidentiality.

♦ Integrity of data: payment information sent from cardholders to merchants includes order information, personal data, and payment instructions. SET guarantees that these message contents are not altered in transit. RSA digital signatures, using SHA-1 hash codes, provide message integrity. Certain messages are also protected by HMAC using SHA-1.

♦ Cardholder account authentication: SET enables merchants to verify that a cardholder is a legitimate user of a valid card account number. SET uses X.509v3 digital certificates with RSA signatures for this purpose.

♦ Merchant authentication: SET enables cardholders to verify that a merchant has a relationship with a financial institution allowing it to accept payment cards. SET uses X.509v3 digital certificates with RSA signatures for this purpose.

SET Participants

♦ Cardholder

♦ Merchant

♦ Issuer

♦ Payment Gateway

♦ Certification Authority

Process:

1. The customer opens an account. The customer obtains a credit card account, such as MasterCard or Visa, with a bank that supports electronic payment and SET.

2. The customer receives a certificate. After suitable verification of identity, the customer receives an X.509v3 digital certificate, which is signed by the bank. The certificate verifies the customer's RSA public key and its expiration date. It also establishes a relationship, guaranteed by the bank, between the customer's key pair and his or her credit card.

3. Merchants have their own certificates. A merchant who accepts a certain brand of card must be in possession of two certificates for two public keys owned by the merchant: one for signing messages, and one for key exchange. The merchant also needs a copy of the payment gateway's public-key certificate.

4. The customer places an order. This is a process that may involve the customer first browsing through the merchant's Web site to select items and determine the price. The customer then sends a list of the items to be purchased to the merchant, who returns an order from containing the list of items, their price, a total price, and an order number.

5. The merchant is verified. In addition to the order form, the merchant sends a copy of its certificate, so that the customer can verify that he or she is dealing with a valid store.

6. The order and payment is verified. The customer sends both order and payment information to the merchant, along with the customer's certificate. The order confirms the purchase of the items in the order form. The payment contains credit card details. The payment information is encrypted in such a way that it cannot be read by the merchant. The customer's certificate enables the merchant to verify the customer.

7. The merchant requests payment authorization. The merchant sends the payment information to the payment gateway, requesting authorization that the customer's available credit is sufficient for this purchase.

8. The merchant confirm the order. The merchant sends confirmation of the order to the customer.

9. The merchant provides the goods or service. The merchant ships the goods or provides the service to the customer.

10. The merchant request payment. This request is sent to the payment gateway, which handles all of the payment processing.