INS

Dealer:

65 = 00

SRI VENKATESHWARA XEROX CENTER

Contact:
VENKATESH
Mob: 9448926729

ENGINEERING NOTES FOR ALL SUBJECTS (ALL SEM AND ALL BRANCHES) ARE AVAILABLE IN THIS SHOP

S.V. XEROX
No: 34/A, Near RNS IT College,
Uttarahalli-Kengeri Main Road,
Channasandra, Bengaluru - 560 061.
Mob: 9611148853, 9886552702

INS

65=00

8TH SEM CSE/ISE

INFORMATION AND NETWORK SECURITY

ASHOK KUMAR K

VIVEKANANDA INSTITUTE OF TECHNOLOGY

MOB: 9742024066

E-MAIL: celestialcluster@gmail.com

Note:

This Booklet includes the notes for only

Unit 2 • 4, 5

Unit 6

Unit 7

The notes for unit 1 and unit 4 will be mailed to you on 29th may 2010. Students can obtain the softcopy of these units on request to the email id: celestialcluster@gmail.com

SPECIAL THANKS

I would also like to thank some of the students for their valuable feedback on my previous notes.

RAJESH RN, *JVIT, Ramanagaram*
SHWETHA, *DBIT, Bangalore*
JANU KHANDARI, *SKIT, Bangalore*
BIPIN, *JSSATE, Bangalore*
SHILPA, *SRSIT, Bangalore*
LAKSHMI, *SKIT, Bangalore*
MANASA, *JSSATE, Bangalore*
RAMYA, *RNSIT, Bangalore*
BRADLEY, *Mangalore*
SUMANTH, *JSSATE, Bangalore*
APEKSHA, *RNSIT, Bangalore*
SRIKANTH, *Tumkur*
RUKMINI, *Hassan*
KARTHIK RAO, *PESSE, Bangalore*
NISHITHA, *PESCE, Mandya*
SHIVU, *BNMIT, Bangalore*
KARTHIKA, *AMCEC, Bangalore*
BHARATH, *JVIT, Ramanagaram*
JAGANNATH, *BMS Evening College, Bangalore*
HEMANTH, *Shimoga*
NACHAPPA, *JSSATE, Bangalore*
PRADEEP, *DSCE, Bangalore*
DARSHINI, *JVIT, Ramanagaram*
MRUDULA, *GAT, Bangalore*
DEEPAK, *RNSIT, Bangalore*
SUHAS K.M, *Bellary*
JAYPRADEEP, *Sapthagiri College of Engineering, Bangalore*
ASHRITHA ALVA, *NMIT, Bangalore*
SANDEEP PAI, *HKBKCE, Bangalore*
HARISH, *Chikmaglur*
GIRISH, *JVIT, Ramanagaram*

Dedicated To:

DEEPIKA N

Sri Venkateshwara College of Engineering, Bangalore

NAYANA M C

Sri Venkateshwara College of Engineering, Bangalore

DEEPIKA C

Global Academy of Technology, Bangalore

MALA B C

BGSIT, Nagamangala, Mandya

UNIT 2:

SECURITY TECHNOLOGY - 1

Syllabus

* Introduction
* physical Design
* Firewalls
* protecting Remote connections

— 6 Hours.

~~INTRODUCTION~~

PHYSICAL DESIGN

* physical design extends the logical design of the information security program.
* Team responsible for physical design performs following tasks
  - → selects specific technologies to support information security blueprint.
  - → Identifies complete technical solutions based on these technologies.
  - → designs physical security measures to support the technical solution.
  - → prepares project plans for the implementation phase

FIREWALLS

* A Firewall in an information security program prevents specific types of information from moving between the outside world (untrusted network) and the inside world (trusted network)
* A Firewall may be a separate computer system, a software service running on existing router or server or a separate network.

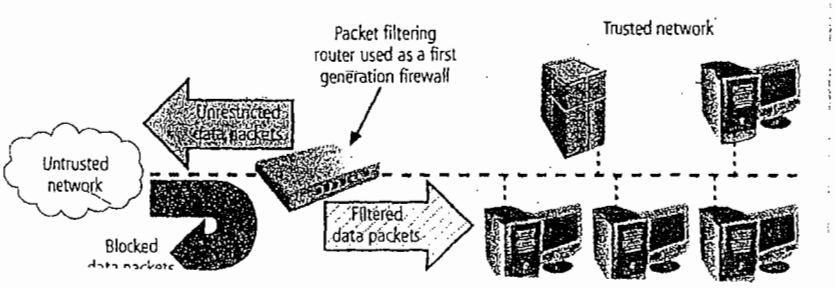
Firewalls can be categorized by
- → processing mode

Firewalls categorized by processing mode.

1. packet Filtering Firewalls
2. Application Gateways
3. Circuit Gateways.
4. MAC Layer firewalls.
5. Hybrids - uses the combination of above four.

packet Filtering firewalls (or simply filtering firewall)

* Filtering firewall examine every incoming packet header and selectively filter packets based on header information such as "dest" address source address, packet type, & other key information.
* when installed on a TCP/IP based n/w, they functions at the IP level.
* they inspect packets at the n/w layer, or layer 3 of the OSI model.
* simple firewall models examine two aspects of packet header: dest^n & source address. They enforce address restrictions, rules designed to prohibit packets with certain addresses or partial addresses from passing thru' the device



* there are three subsets of packet filtering firewalls.
  - → static filtering
  - → Dynamic filtering
  - → ~~stat~~ Stateful inspection.

* Static Filtering Firewall
  - → It requires that filtering rules be developed & installed with the firewall
  - → the rules are created and sequenced either by a person directly editing the rule set, or by a person using a programmable interface to specify the rules and the sequence.
  - → Any changes to the rules require human intervention
  - → this type of filtering is common in n/w routers & gateways

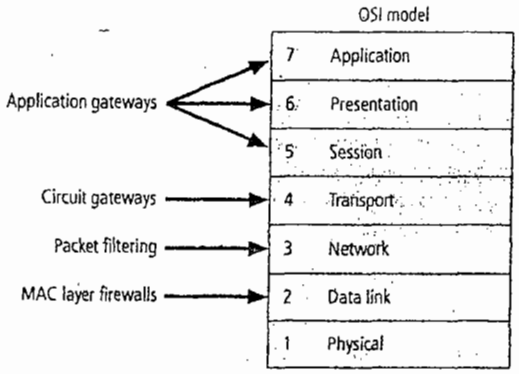
* Dynamic Filtering Firewall
  - → It can react to an emergent event & updat or create rules to deal with that event.
  - → this reaction ~~could be~~ could be
    - positive, as in allowing an internal user to engage in specific activity upon request
    - negative, an in dropping all packets from a particular address when increase in malformed packet is detected.
  - → while static filtering allows ~~only a~~ entire sets of one type of packet to enter in response to authorized requests, the Dynamic packet filtering firewall allows only a particular packet with a particular source, destn, & port address to enter.

Circuit Gateways

* They operate at transport layer
* Like Filtering firewalls, circuit gateway firewalls do no usually look at traffic flowing b/w one n/w & anothe but they do prevent direct connections b/w one n/w & another.
* They accomplish this by creating tunnels connecting specific processes or systems on each side of the firewall.

MAC Layer Firewalls.

* They are designed to operate at the media access control sub-layer of the data link layer of the OSI n/w model.



* Stateful inspection firewalls:

→ It keeps track of each n/w connection b/w internal and external system using a state table

→ fig shows state table entries:

Source Addr.	Source port	Destn addr	Destn port	Time remaining (sec)	Total time (sec)	Protoco
192.168.2.5	1028	10.10.10.7	80	2725	3600	TCP

Application Gateways (or proxy server)

* Also known as Application-level firewall or application firewall.
* It is frequently installed on a dedicated computer separate from the filtering router, but is commonly used in conjunction with a filtering router
* one common ex of an application firewall is a firewall that blocks all requests for and responces to requests for web pages & services from the internal computers of an organization, & instead makes all such requests & responses go to intermediate computers (or proxies) in the less protected areas of the organization's n/w.

Firewalls categorized by generation.

1. First generation Firewalls - static packet filtering firewalls.
2. Second generation Firewalls - Application level firewalls or proxy servers.
3. Third generation Firewalls - Stateful inspection firewal
4. Fourth generation Firewalls - Dynamic packet filtering firewalls.
5. Fifth generation Firewalls -

   These are kernel proxy, a specialized form that works under windows NT executive, which is the kernel of windows NT.

   This type of firewall evaluates packets at multiple layers of the protocol stack, by checking security in the kernel as data is passed up and down the stack.

Firewalls categorized by structure

→ Commercial grade Firewall Appliances
→ Commercial grade Firewall systems
→ Small office / Home office (SoHo) firewall appliances
→ Residential Grade Firewall software.

Commercial grade Firewall Appliances

→ These are stand alone, self contained combinations of computing hardware and software.

Commercial grade Firewall Systems

→ consists of application software that is configured for the firewall application and run on a general purpose computer.

SOHO Firewall appliances (or resident grade firewall Appliances)

→ It is the effective method of improving computing security in the residential setting.

→ SOHO devices, also known as broadband gateways or DSL/Cable modem routers, connect the user's LAN or a specific comp system to the internetworking device - in this case, the cable modem or DSL router provided by ISP.

→ The SOHO firewall serves first as a stateful firewall to enable inside to outside access and can be configured to allow limited TCP/IP port forwarding and/or screened subnet capabilities.

Residential Grade Firewall Software

→ Here software firewall is installed directly on the user's system to protect the residential user.

Firewall Architectures.

* The configuration of an firewall that works best for an particular ~~of~~ organization depends on three factors:
  - the objectives of the n/w
  - the organization's ability to develop and implement the architectures
  - the budget available for the function.

* There are four common architectural implementations of firewalls:
  1. packet filtering routers
  2. screened host firewalls
  3. Dual horned firewalls
  4. screened subnet Firewalls.

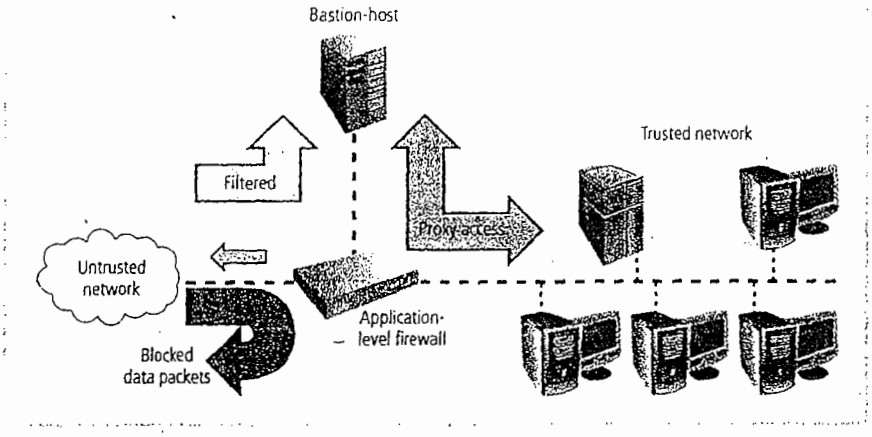
packet Filtering Routers: for fig, refer pg no 3

* Here routers is placed at the boundary b/w the organization's internal n/ws and external service provider.
* These routers will be configured to reject packets that the organization does not allow into the n/w.

* Drawbacks:
  - → lack of auditing and strong ~~auth~~ authentication.
  - → Complexity of the access control lists.

Screened Host Firewalls

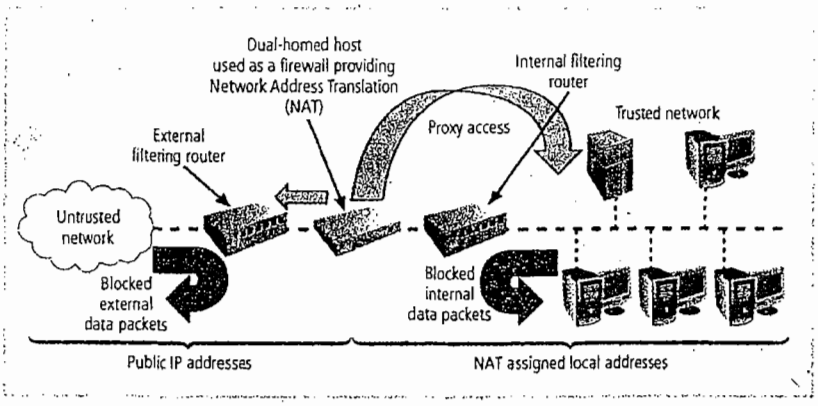
* It combines the packet filtering router with a separate, dedicated firewall, such as an application proxy server.
* This approach allows the router to prescreen packets to minimize the network traffic and load on the internal proxy.
* The application proxy (or bastion host or sacrificial host) examines an application layer protocol, such as HTTP, and performs the proxy services.



Dual-Homed Host Firewalls

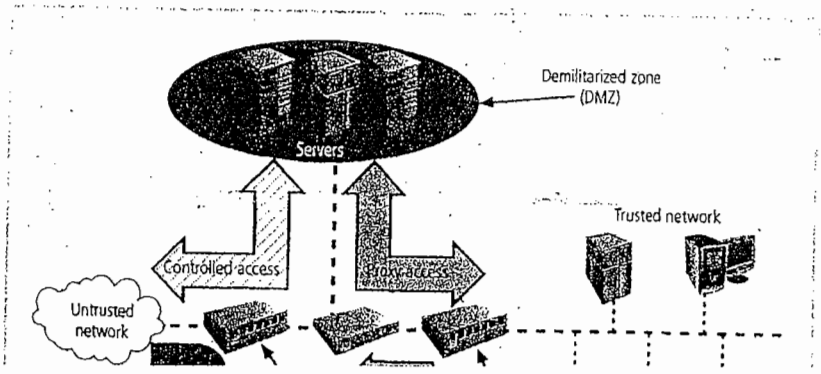
* Here, the bastion host contains two NICs rather than one.
  → One NIC is connected to the external n/w
  → the other is connected to the internal n/w
  thus providing an additional layer of protection.

* implementation of this architecture often makes use of NAT. (n/w address translation) NAT is a method of mapping real, valid, external IP address to special ranges of non routable internal IP addresses, thereby creating yet another barrier to intrusion from external attackers.



- Screened Subnet Firewalls (with DMZ) - latest domain architecture

* This architecture provides a DMZ. The DMZ can be a dedicated port on the firewall device ~~linked~~ linking a single bastion host, or it can be connected to a screened subnet, as shown



* Until recently, servers providing services through an untrusted network were commonly placed in the DMZ. Ex of these include Web servers, FTP servers, & certain DB servers.

* In this architecture, the subnet firewall consisting of two or more internal bastion hosts behind a packet filtering router, with each host protecting the trusted n/w.

+ There are many variants of this architecture

1. First general model consists of two filtering routers, with one or more dual-homed bastion hosts b/w them
2. In second general model (shown in fig), the connections are routed as follows -

→ Connections from the outside or untrusted n/w are routed thru' an external filtering router.

→ Connections from the outside or untrusted n/w are routed into - and then out of - a routing firewall to the separate network segment known as the DMZ.

→ Connections into the trusted internal n/w are allowed only from the DMZ bastion host servers.

Selecting the Right Firewall for an organization

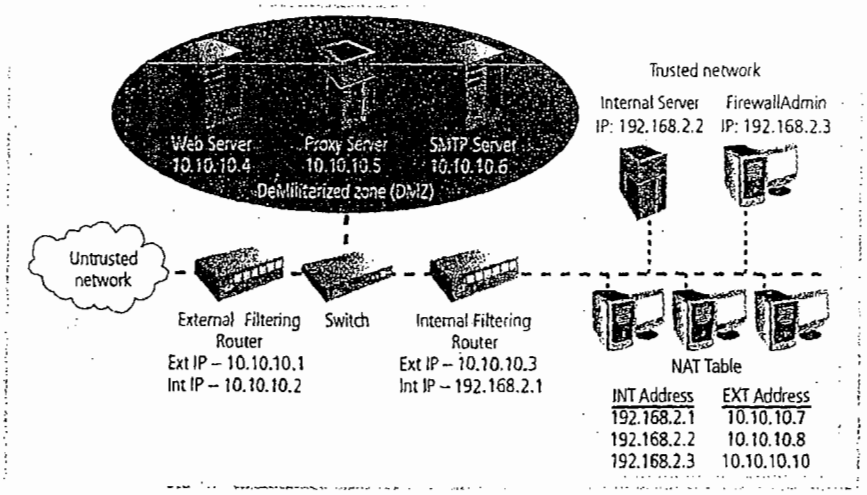
Questions to be considered -

1. What type of firewall technology offers the right balance b/w protection and cost for the needs of the organization?
2. What features are included in the base price? What features are available at extra cost? Are all cost factors known?
3. How easy it is to set up and configure the firewall? How accessible are the staff technicians who can competently configure the firewall?
4. Can the candidate firewall adapt to the growing n/w in the target organization?

Configuring and Managing Firewalls.

* Configuring firewall policies is as much an art as it is a science.
Each configuration rules must be carefully crafted, debugged, tested, and placed into the access control list in the proper sequence.

Firewall Rules (not important) Ex for rulesets.



Ruleset 1: Responses to internal requests are allowed.

Ruleset 2: The Firewall device is never accessible directly from the Public n/w.

Ruleset 3: All traffic from the trusted n/w is allowed out

Ruleset 4: Ruleset for SMTP data

Ruleset 5: All ICMP data should be denied.

Ruleset 6: Telnet access to all internal servers from the public n/w s should be blocked.

Ruleset 7: When webservices are offered outside the firewall, HTTP traffic should be blocked from the internal n/w

Ruleset 8: The cleanup Rule, if a request for service is not explicitly allowed by policy, that request should be denied by rule.

Content Filters.

* Content filter is another utility that can help protect an organization's systems from misuse and unintentional denial-of-service problems, and which is often closely associated with firewalls. ~~is the content fil~~
* Content filters are also called reverse firewalls because their primary purpose is to restrict internal access to external material.
* Content filters has two components:
  1. Rating
  2. Filtering

→ Rating is like a set of firewall rules for websites, & is common in residential content filters.
It can be
- complex, with multiple access control settings for different levels of the organization.
- simple, with a basic allow/deny scheme like that of a firewall.

→ The filtering is a method used to restrict specific access requests to the identified resources, which may be web sites, servers, or whatever resources the content filter administrator configures.

PROTECTING REMOTE CONNECTIONS

Virtual private Networks [VPNs]

* The virtual private Network Consortium (VPNC) defines a VPN as:

> "A private data network that makes use of the public telecommunication infrastructure, maintaining privacy through the use of a tunneling protocol and security procedures"

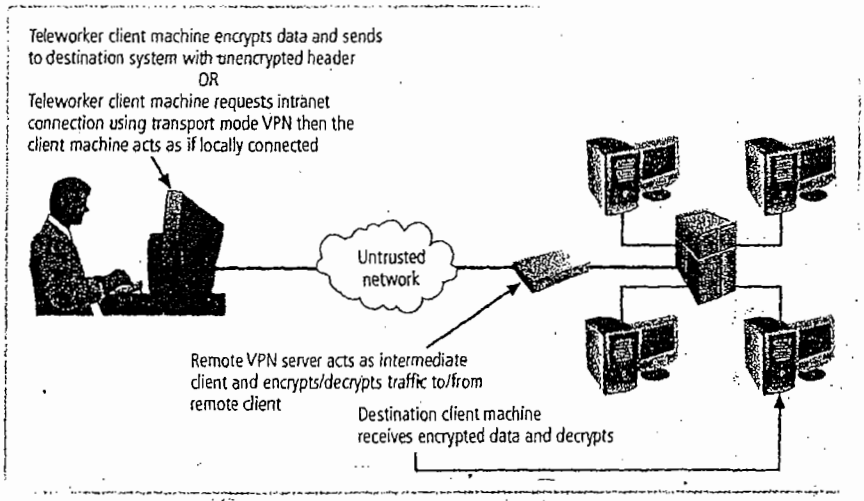
* VPNs are commonly used to securely extend an organization's internal n/w connections to remote locations.
* The VPNC defines three VPN technologies:
  - → trusted VPNs.
  - → secure VPNs
  - → Hybrid VPNs.
* Trusted VPN (legacy VPN) uses leased circuits from a service provider and conducts packet switching over these leased circuits. The organizn must trust the service provider.
* Secure VPNs use security protocols and encrypt traffic transmitted across unsecured public networks like the internet.

* A hybrid VPN combines the two, providing encrypted transmissions over some or all of a trusted VPN network.

* A VPN must accomplish the following.
  - Encapsulation of incoming & outgoing data
  - Encryption of incoming & outgoing data
  - Authentication of the remote computer.

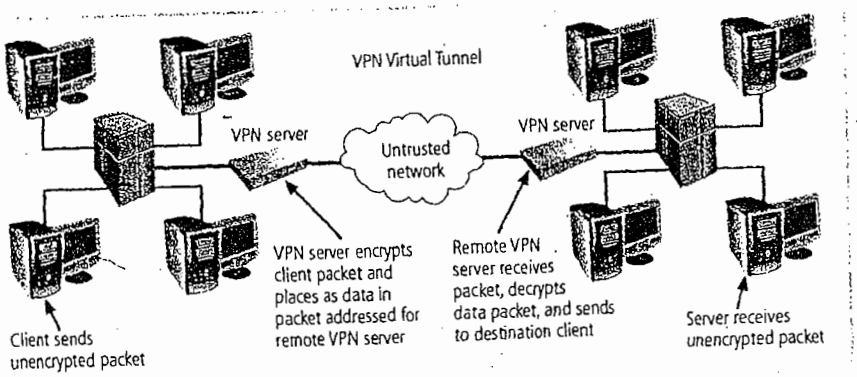
* there are a number of ways to implement a VPN.
  - Transport Mode
  - Tunnel mode

Transport mode



* In Transport mode, the data within an IP packet is encrypted, but the header information is not.
  This allows a user to establish a secure link directly with the remote host, encrypting only the data contents of the packet.
* Adv: Eliminates the need for special servers and tunneling software, and allows the end users to transmit traffic from anywhere.
* Disadv: packet eavesdroppers can still identify the destination system. Once the attacker knows the dest^n, he/she may be able to compromise one of the end nodes & acquire packet information from it

Tunnel mode



* In <u>tunnel mode</u>, the entire client packet is encrypted and added as the data portion of a packet addressed from one tunneling server and to another.
* the receiving server decrypts the packet and sends it to the final address.
* the primary benefit to this model is that an intercepted packet reveals nothing ~~but~~ about the true destination system.

Question: what is the relationship b/w TCP and UDP packet? will any specific transaction usually involve both types of packets?

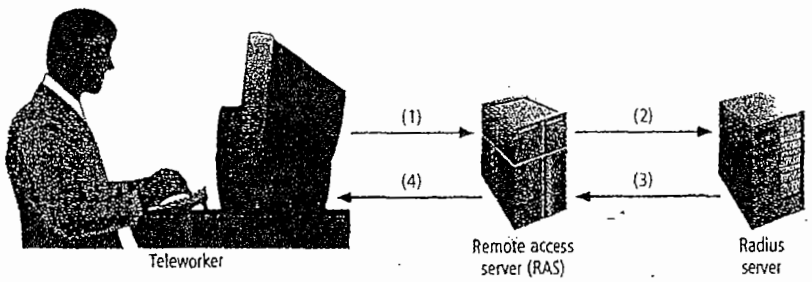
~~Answer:~~

Question: what is RADIUS? what advantages does it have over TACACS?

Answer:

RADIUS and TACACS are systems that authenticate the credentials of users who are trying to access an organization's n/w via dialup connection.

The RADIUS (Remote ~~ccc~~ authentication Dial-In User Service) system centralizes the management of user authentication by placing the responsibility for authenticating each user in the central RADIUS server.



1. Remote worker dials RAS and submits username and password
2. RAS passes username and password to RADIUS server
3. RADIUS server approves or rejects request and provides access authorization
4. RAS provides access to authorized remote worker

* The terminal Access Controller Access control System (TACACS) is based on client/server configuration.
* Like RADIUS, it contains a centralized database and it validates the users credentials at this TACACS server.
* There are three versions of TACACS.
  i.) original TACACS - combines authentication and authorization services.
  ii.) Extended TACACS - separates the steps needed to ~~verify that~~ authenticate the individual or system attempting access from the steps needed to verify that the authenticated individual or system is allowed to make this type of connection.
  iii.) TACACS+ - uses dynamic passwords and incorporates two-factor authentication.

Unit-1 Cryptography

Syllabus

* Introduction
* A short History of cryptography.
* principles of cryptography.
* cryptography tools
* Attacks on cryptosystems

- 8 Hours

INTRODUCTION.

* Cryptology is the science of encryption, it encompasses cryptography and cryptanalysis

→ cryptography is the process of making and using codes to ~~ensure~~ secure the transmission of information.

kryptos → hidden. } in Greek.
graphein → to write }

→ Cryptanalysis is the process of obtaining original message (plaintext) from an encrypted message (ciphertext) without knowing the algorithms & keys used to perform the encryption.

* Encryption is the process of converting an original message into a form that is unreadable to unauthorized individuals.

* Decryption is the process of converting the ciphertext message back into plaintext so that it can be readily understood.

PRINCIPLES OF CRYPTOGRAPHY

Terminologies

1. **Algorithm**
   The programmatic steps used to convert an unencrypted message into an encrypted sequence of bits that represents the message

2. **Cipher or cryptosystem**
   - An encryption method or process encompassing the algorithm, key(s) or cryptovariable(s), and procedures used to perform encryption & Decryption.

3. **Ciphertext or cryptogram.**
   encoded message resulting form an encryption

4. **Code**
   process of converting components (words or phrases) of an unencrypted message into encrypted components.

5. **Decipher**
   to decrypt or convert ciphertext into the equivalent plaintext

6. **Encipher**
   encrypt or convert plaintext into the

7. Key or crypto variable

The information used in conjunction with an algorithm to create the ciphertext from plaintext or derive the plaintext from ciphertext.

8. keyspace

entire range of values that can be used to construct an individual key.

9. Link encryption.

Series of encryptions and decryptions b/w a no. of systems, where each system in a n/w decrypts the msg sent to it & then reencrypts it using diff. keys and sends it to the next neighbor, & this process continues until the message reaches the final dest^n

10. plaintext or cleartext

original unencrypted message, or a message that has been successfully decrypted.

11. Steganography.

Hiding of messages

12. Work factor.

Amount of effort (usually in hours) required to perform cryptanalysis to decode an encrypted message when the key or algorithm (or both) are unknown.

Cipher methods

1. Substitution cipher
2. Transposition cipher
3. Exclusive OR
4. Vernam cipher
5. Book or Running key cipher.
6. Hash Functions.

Substitution cipher.

- Here you substitute one value for another.

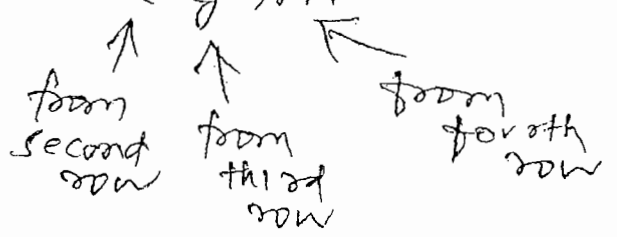
eg: ABCD .... VWXYZ is substituted by
DEFG .... YZABC

This type of substitution is called monoalphabetic substitution because it uses only one alphabet.

* Example for poly alphabetic substitution
(substitution ciphers uses two or more alphabets)

plaintext: ABCD.... WXYZ
Substitution cipher1: DEFG.... ZABC.
Substitution cipher2 GHIJ.... CDEF
Substitution cipher3 JKLM.... FGHI.

To ~~code~~ encode BAD,
ciphertext is EGM.



Vigenere cipher

advanced type of substitution cipher that uses a simple poly alphabetic code.

- the cipher is implemented using Vigenere square made up of 26 distinct cipher alphabets.

	A	B	C	D	....	X	Y	Z
1	B	C	D	E	...	Y	Z	A
2	C	D	E	F	...	Z	A	B
3	D	E	F	G	...	A	B	C
	:	:	:	:		:	:	:
23	X	Y	Z	A	...	V	W	W
24	Y	Z	A	B	...	W	X	X
25	Z	A	B	C	...	X	Y	Y
26	A	B	C	D	...	Y	Z	Z

plaintext : SECURITY
ciphertext : TEFYWOAG.

Transposition cipher (permutation cipher)

* It simply rearranges the values within a block to create the ciphertext.
This can be done at bit level or byte level.

eg: key pattern : $1\to4,\ 2\to8,\ 3\to1,\ 4\to5$
$5\to7,\ 6\to2,\ 7\to6,\ 8\to3$

Bit locations	87654321	87654321
plaintext 8 bit block	00100101	01101011
Ciphertext	00001011	10111010.

* Exclusive OR (XOR) operation is a function of Boolean algebra in which two bits are compared, and if the two bits are identical, the result is a binary 0. If two bits are not the same, the result is a binary 1.

Truth table

First bit	Second bit	Result
0	0	0
0	1	1
1	0	1
1	1	0

* XOR encryption.

plaintext: CAT is converted into
key: VVV binary representation.

plaintext: 0 1 0 0 0 0 1 1 0 0 1 0 1 1 0 0 0 0 0 0
key: 1 0 1 0 1 0 1 1 0 0 0 1 0 1 1 0 0 0 1 0 1
Cipher: 1 1 1 0 1 1 1 0 0 0 0 0 0 0 0 0 1 0 1

Vernam cipher. (one time pad)

* uses a set of characters only one time for each encryption process.

Plaintext: S A C K G A U L S P A R E

Plaintext value: 19 01 03 11 07 01 21 12 19 16 01 18 05

One time pad text: F P Q R N S B I E H T Z L

One time pad value: 06 16 17 18 14 19 02 09 05 08 20 26 12

Sum of plain text & pad: 25 17 20 29 21 20 23 21 24 24 21 44 17

Ajko modulo substraction: 03 18

Ciphertext: Y Q T C U T W U X U R Q.

Hash functions

* Hash functions are mathematical algorithms that generate a message summary or message digest (fingerprint) to confirm the identity of specific message & to confirm that there have not been any changes to the content.

* Hash Algorithms are public functions that create a hash value (message digest) by converting variable length messages into a single fixed length value.
  - It is like fingerprint of author's message.

Cryptographic hash functions do not require the use of keys, but it is possible to attach a message Authentication code (MAC) — a key dependent one way hash function — that allows only specific recipients to access the message digest.

* Because hash functions are one way, they are used in password verification systems to confirm the identity of the user.

* Time-memory tradeoff attack:

- If attackers gain access to a file of hashed passwords, they can use a combination of brute force and dictionary attacks to reveal user passwords.
Well constructed passwords take a long time to crack. But by using a rainbow table (a database of precomputed hashes from sequentially calculated passwords), the rainbow cracker simply looks up the hashed password & reads out the text version, no brute force required.
This type of attack is more properly classified as a time-memory tradeoff attack.

Cryptographic Algorithms

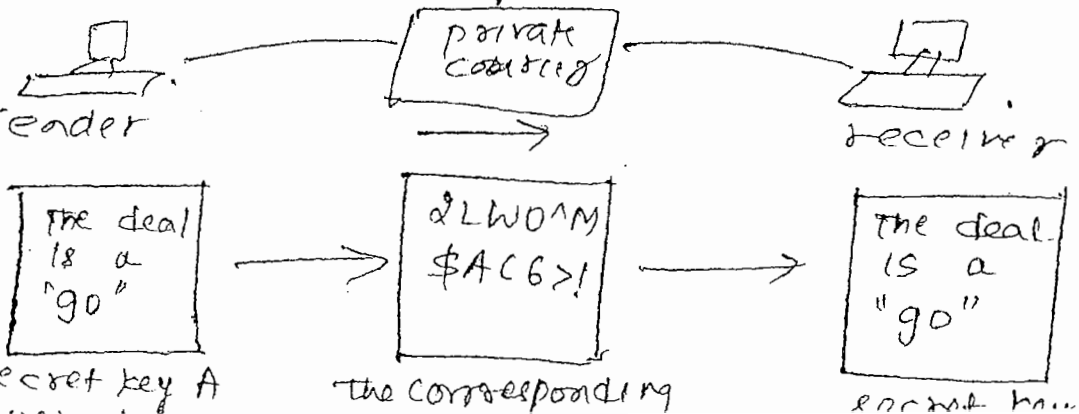
Cryptographic algorithms are often grouped into

1. Symmetric Encryption
2. Asymmetric Encryption.

Combination of these two are also in use today.

Symmetric Encryption.

* Encryption methodologies that requires the same secret key to encipher and decipher the message are using what is called private key encryption or symmetric encryption.
* Symmetric encryption methods use mathematical operations that can be programmed into extremely fast computing algorithms so that the encryption and decryption processes are executed quickly by even small computers.



If the key falls in wrong hands, messages can be decrypted by others and the sender and intended receiver may not know the message was intercepted.

* The primary challenge of symmetric encryption is getting the key to the receiver, a process that must be conducted out of band to avoid interception.

* There are a number of popular symmetric encryption cryptosystems:

1. Data Encryption Standard (DES)
2. Triple DES (3DES)
3. Advance Encryption Standard (AES)

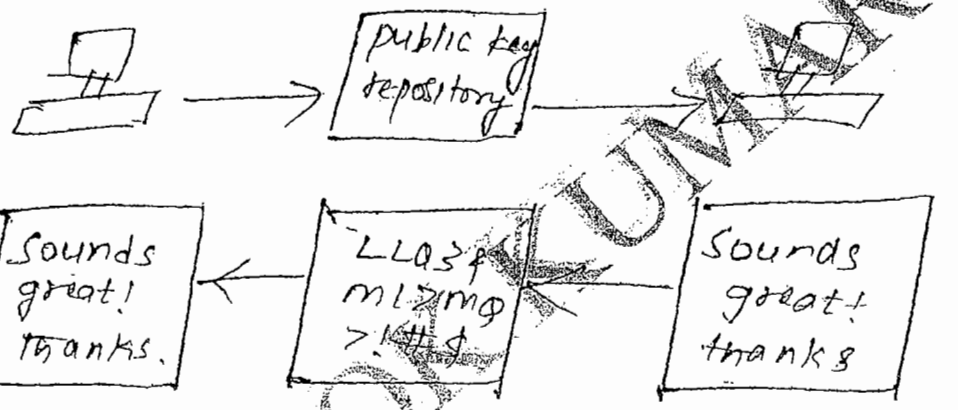
* DES
  - → Developed by IBM
  - → uses 64 bit block size and a 56 bit key.
  - does not provide Acceptance level of security

- Triple DES.
  - → Advanced application of DES.

~~Advanced Encryption standard~~.

Asymmetric Encryption. (or public key encryption)

* Asymmetric encryption uses two different but related keys, and either key can be used to encrypt or decrypt the message.
* However, if key A is used to encrypt, only key B can decrypt it, and if key B is used to encrypt a message, only key A can decrypt it.



* Asymmetric Algorithms are one-way functions. A one way function is simple to compute in one direction, but complex in the opposite direction.
* A mathematical trapdoor is a "secret mechanism that enables you to easily accomplish the reverse function in a one-way function".

or decrypt the ciphertext, but not both, thus requiring two keys.

The public key becomes be the true key, and the private key is to be derived from the public key using the trapdoor.

* Ex:
  RSA algorithm.

CRYPTOGRAPHIC TOOLS

- public key Infrastructure (PKI)
- Digital Signatures
- Digital certificates
- Hybrid cryptography systems
- Steganography

public key Infrastructure (PKI)

* PKI is an integrated system of software, encryption methodologies, protocols, legal agreements, and third party services that enables users to communicate securely.
* PKI systems are based on public key cryptosystem

PKI (Public Key Infrastructure) protects the transmission and reception of secure information by integrating the following components

→ A certificate Authority (CA), which issues, manages, authenticates, signs, and revokes user's digital certificates, which typically contain the username, public key, and other identifying information.

→ A registration Authority (RA), which operates under the trusted collaboration of the certificate authority and can handle day to day certification functions such as verifying registration information, generating end user keys, revoking certificates, & validating user certificates

→ certificate directories, which are central locations for certificate storage that provide a single access point for administration & distribution.

→ management protocols, which organize and manage the communications b/w CAs, RAs, and end users

→ policies and procedures, which assist an organizn in the application and management of certificates the formalizations of legal liabilities & limitations

Certificate management include
- Systems to issue digital certificates to users and servers
- Directory enrollment
- Key issuing systems
- Tools for managing the key issuance.
- Verification and return of certificates.

Digital Signatures

* Non repudiation is the principle of cryptography that underpins the authentication mechanism collectively known as a Digital Signature. Digital signature are encrypted messages that can be mathematically proven authentic.
* The management of Digital signatures has been built into most web browsers.
* Digital signatures should be created using processes and products that are based on the Digital signature Standard (DSS).

Digital Certificates

* Digital certificate is an electronic document or container file that contains a key value and identifying information about the entity that controls the key.

* The certificate is often issued and certified by a third party, usually a Certification Authority.
* A digital signature attached to the certificate's container file certifies the file's origin and integrity. This verification process often occurs when you download or upload software via internet.
* Unlike digital signatures, which help authenticate the origin of a message, digital certificates authenticate the cryptographic key that is embedded in the certificate.
* Fig shows X.509v3 certificate structure.

fig 302 pg

* most common hybrid system is based on the Diffie-Hellman Key Exchange, which is a method for exchanging private keys using public key encryption.

Steganography

* "steganography" - art of secret writing
  - steganos - "covered"
  - graphein - "to write"
* It is technically not a form of cryptography, it is another way of protecting the confidentiality of information in transit.
* the most popular version of steganography involves hiding information within files that contain digital pictures or other images.
* How images are stored in computer?
  - RGB to represent a pixel. Each of these color requires 8 bit code for color intensity.
    ex 00000000 - no red
    11111111 - max. red.
  - image having 1024 X 768 resolution contains 786432 pixels or three quarters of a megapixel.

In this case, there is no difference b/w a pixel with red intensity 00101001 & 00101000. This provide a steganographer with one bit per color (or 3 bits per pixel) to use for encoding data into an image file.

* If a steganographic process uses 3 bits per pixel for all 786,432 pixels, it will be able to store 236 KB of hidden data within the uncompressed image.

ATTACKS ON CRYPTOSYSTEMS

1. Known plaintext attack scheme:
   Here, an attacker obtain duplicate texts, one in ciphertext and one in plaintext, and thus reverse-engineer the encryption algorithm.

2. Selected plaintext attack scheme:
   Here, an attacker sends a specific text to potential victims that they are sure the victims will forward on to others.

Four general categories of attacks:

1. Man in the middle attack.
2. Correlation attacks.
3. Dictionary Attacks.
4. Timing attacks.

Man-in-the-middle attack.

* It attemps to intercept a public key or even to insert a known key structure in place of the requested public key.
- Thus, attackers attempt to place themselves b/w the sender and receiver, and once they have intercepted the request for key exchanges, they send each participant a valid public key, which is known only to them.
* Establishing public keys with digital signatures can prevent the traditional man in the middle attack, as the attacker can not duplicate the signatures.

Correlation Attacks

- These are a collection of brute force methods that attempt to deduce statistical relationships b/w the structure of the unknown key and the ciphertext generated by the cryptosystem. The only defense against this attack is the selection of strong cryptosystems that have stood the test of time, thorough key management, and strict adherence to the best practices of cryptography in the frequency of key changes.

* Here the attacker encrypts every word in a dictionary using the same cryptosystems as used by the target in an attempt to locate a match between the target ciphertext and the list of encrypted words.
* Dictionary attacks can be successful when ciphertext consists of relatively few characters.

Timing Attacks

* Here, the attacker eavesdrops on the victim's session and uses statistical analysis of patterns and inter key stroke timings to discern sensitive session information.
* while timing analysis may not directly result in the decryption of sensitive data, it can be used to gain information about the encryption key and perhaps the cryptosystem.
* Having broken the encryption, the attacker may launch a replay attack; which is an attempt to resubmit a recording of the deciphered authentication to gain entry into a secure source.

INTRODUCTION TO NETWORK SECURITY, AUTHENTICATION APPLICATIONS.

- note: only the part of the 5th unit has been covered here.

ASHOK KUMAR K (mob: 9742024066)

INTRODUCTION TO NETWORK SECURITY, AUTHENTICATION APPLICATIONS.

note: only the part of the 5th unit has been covered here.

SECURITY ATTACKS

* Security attacks: Any action that compromises the security of information owned by an organization.

* Security mechanism: A mechanism that is designed to detect, prevent or recover from a security attack.

* Security service: A service that enhances the security of data processing systems and the information transfers of an organization.

* Security Attacks
  - passive Attacks
  - Active Attacks.

passive Attacks.

* passive attacks are in the nature of eavesdropping on, or monitoring of, transmissions
* The goal of opponent is to obtain information that is being transmitted.

Passive attacks:

- Release of message contents
- Traffic Analysis.

+ Release of message contents: refer fig (a).

→ A telephone conversation, an email message, and a transferred file may contain sensitive or confidential information.

→ We would like to prevent an opponent from learning the contents of these transmissions.

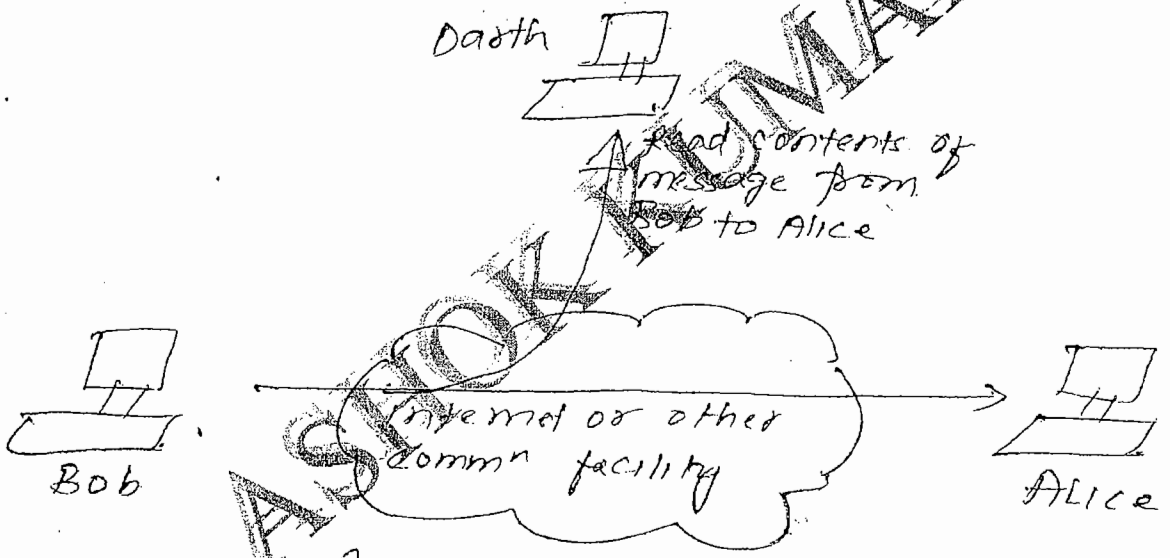


fig (a) Release of message content.

+ Traffic Analysis refer fig (b).

→ Suppose we had masked the contents of message using encryption, the opponent could not extract the information from the message.

→ But, he can still be able to observe the pattern of these messages.

 the location & identity of communicating hosts and could observe the frequency & length of messages being exchanged.
This information might be useful in guessing the nature of the communication that was taking place.

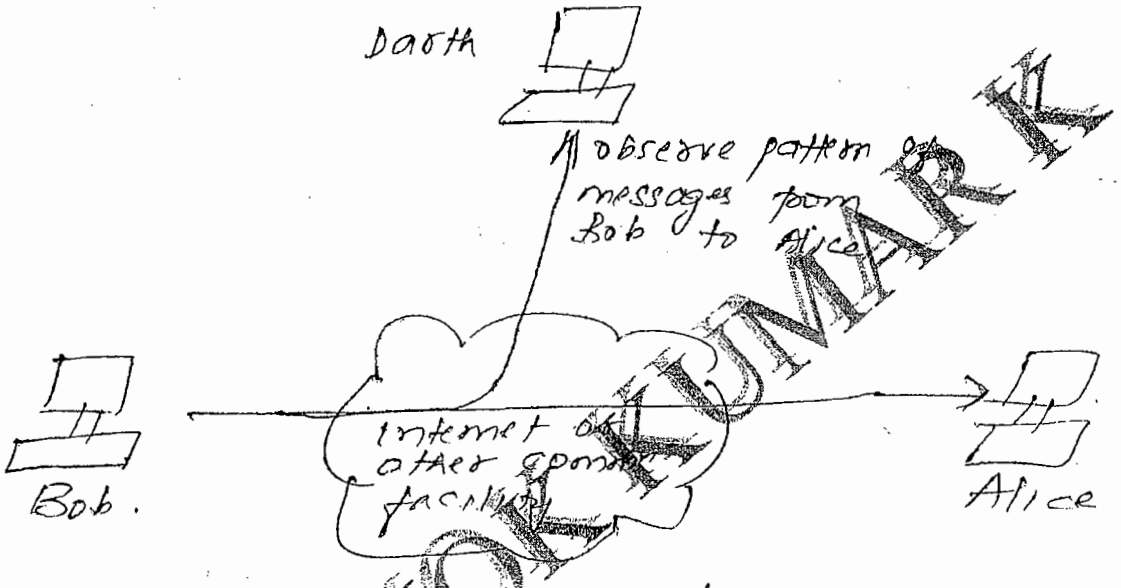


fig (b) Traffic analysis

* passive attacks are very difficult to detect because they do not involve any alteration of the data.
* Neither the sender Nor the receiver is aware that a third party has read the messages or observed the traffic pattern.
* Thus, emphasis in dealing with passive attacks is on prevention rather than detection.

Active attacks

* Active attacks involve some modification of the data stream or the creation of a false stream and can be subdivided into four categories
  - masquerade
  - Replay.
  - ~~modifiability~~.
  - modification of messages
  - Denial of service

* masquerade: refer fig (a)

→ masquerade takes place when one entity pretends to be a different entity.

→ This attack usually includes one of the other forms of active attacks.

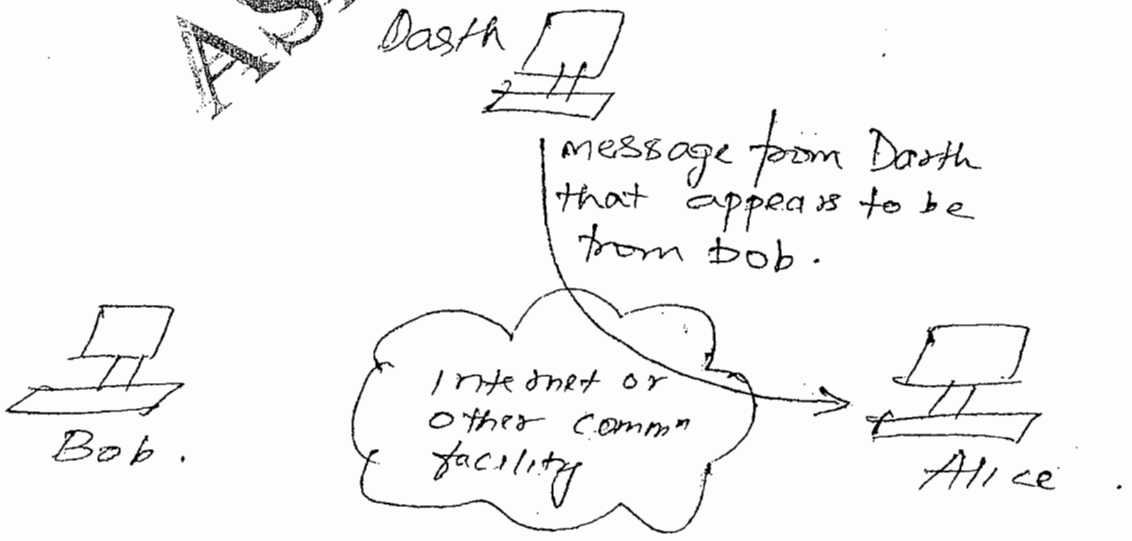
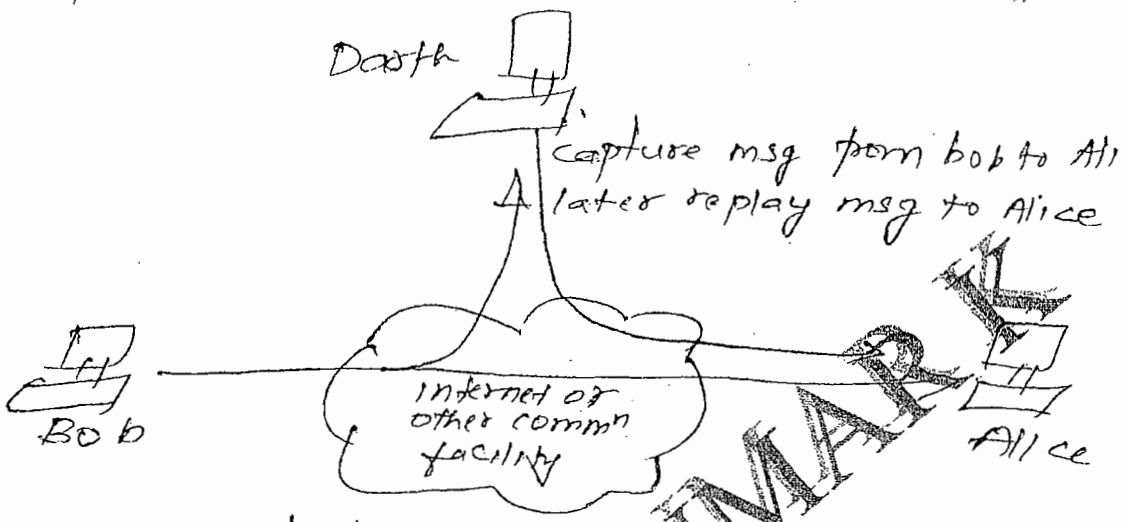


fig (a): Masquerade.

Replay

→ Replay involves the passive capture of a data unit and its subsequent ~~transmission~~ retransmission to produce an unauthorized effect.



fig(b): Replay.

* Modification of messages: refer fig (c).

→ It simply means that some portion of a legitimate message is altered, or that messages are delayed or reordered to produce an unauthorized effect.

→ ex: a msg meaning "Allow John to read confidential file" is modified to mean "Allow ~~John~~ Fred to read confidential file".

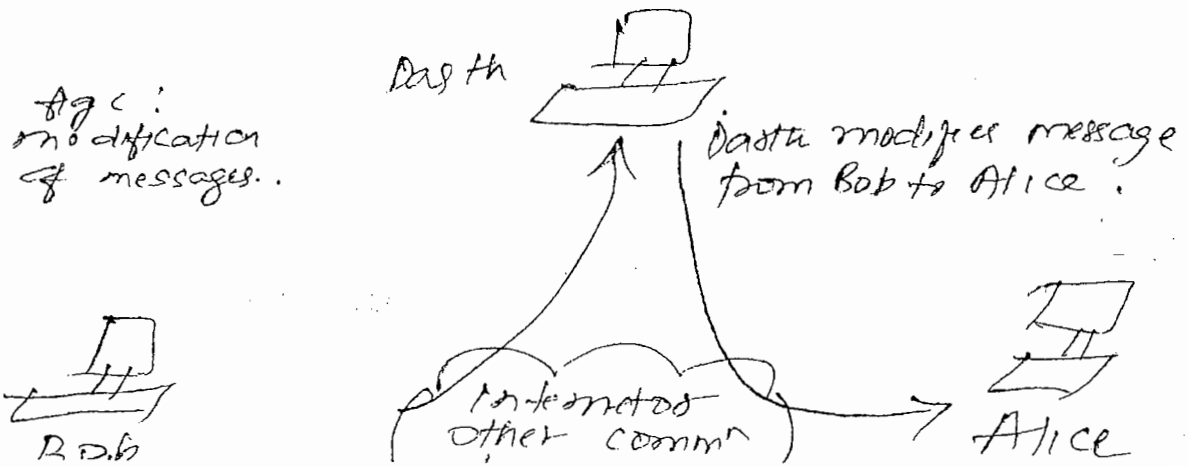
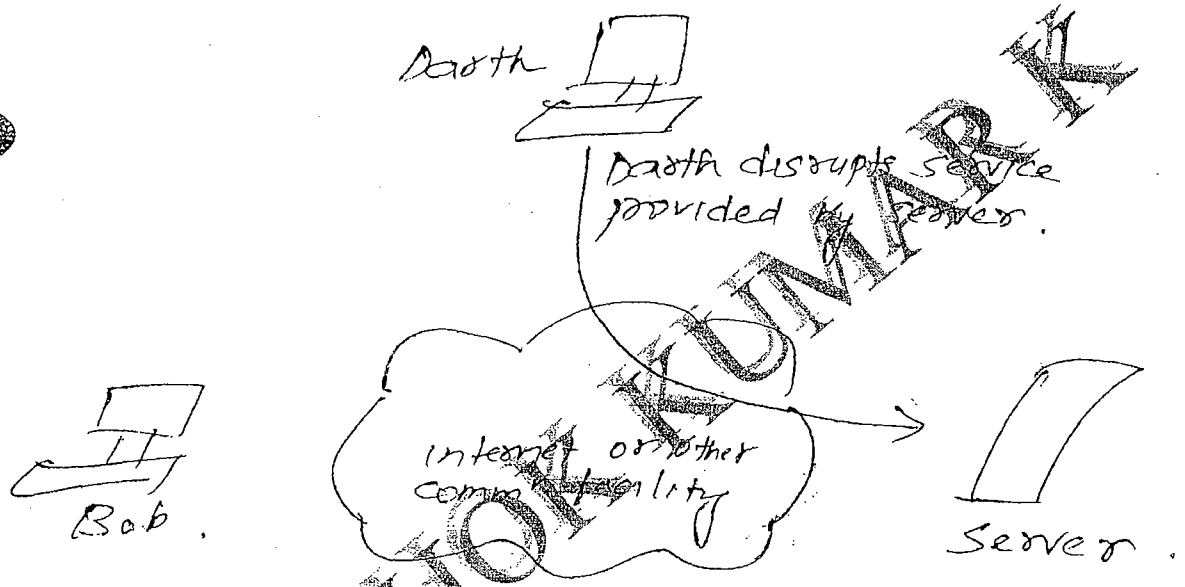


 fig (a)

→ It prevents or inhibits the normal use or management of communication facilities
→ An entity may suppress all messages directed to a particular dest^n.
→ It can disrupt the entire n/w either by disabling the n/w or by overloading it with messages so as to degrade performance.



* It is quite difficult to prevent active attack.
The goal is to detect them & to recover from any disruption or delays caused by them.

ASHOK KUMAR K (mob: 9742024066)

Table 1.2 Security Services (X.800)

AUTHENTICATION	DATA INTEGRITY
The assurance that the communicating entity is the one that it claims to be.	The assurance that data received are exactly as sent by an authorized entity (i.e., contain no modification, insertion, deletion, or replay).
Peer Entity Authentication Used in association with a logical connection to provide confidence in the identity of the entities connected.	**Connection Integrity with Recovery** Provides for the integrity of all user data on a connection and detects any modification, insertion, deletion, or replay of any data within an entire data sequence, with recovery attempted.
Data-origin Authentication In a connectionless transfer, provides assurance that the source of received data is as claimed.	**Connection Integrity without Recovery** As above, but provides only detection without recovery.
ACCESS CONTROL	**Selective-Field Connection Integrity** Provides for the integrity of selected fields within the user data of a data block transferred over a connection and takes the form of determination of whether the selected fields have been modified, inserted, deleted, or replayed.
The prevention of unauthorized use of a resource (i.e., this service controls who can have access to a resource, under what conditions access can occur, and what those accessing the resource are allowed to do).	**Connectionless Integrity** Provides for the integrity of a single connectionless data block and may take the form of detection of data modification. Additionally, a limited form of replay detection may be provided.
DATA CONFIDENTIALITY	**Selective-Field Connectionless Integrity** Provides for the integrity of selected fields within a single connectionless data block; takes the form of determination of whether the selected fields have been modified.
The protection of data from unauthorized disclosure.	**NONREPUDIATION**
Connection Confidentiality The protection of all user data on a connection.	Provides protection against denial by one of the entities involved in a communication of having participated in all or part of the communication.
Connectionless Confidentiality The protection of all user data in a single data block	**Nonrepudiation, Origin** Proof that the message was sent by the specified party.
Selective-Field Confidentiality The confidentiality of selected fields within the user data on a connection or in a single data block.	**Nonrepudiation, Destination** Proof that the message was received by the specified party.
Traffic-flow Confidentiality The protection of the information that might be derived from observation of traffic flows.	

Two specific authentication services are defined in the standard:

- **Peer entity authentication:** Provides for the corroboration of the identity of a peer entity in an association. Two entities are considered peer if they implement to same protocol in different systems; e.g., two TCP modules in

ASHOK KUMAR K (mob: 9742024066)

Table 1.3 Security Mechanisms (X.800)

SPECIFIC SECURITY MECHANISMS May be incorporated into the appropriate protocol layer in order to provide some of the OSI security services.	PERVASIVE SECURITY MECHANISMS Mechanisms that are not specific to any particular OSI security service or protocol layer.
Encipherment The use of mathematical algorithms to transform data into a form that is not readily intelligible. The transformation and subsequent recovery of the data depend on an algorithm and zero or more encryption keys.	**Trusted Functionality** That which is perceived to be correct with respect to some criteria (e.g., as established by a security policy).
Digital Signature Data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and integrity of the data unit and protect against forgery (e.g., by the recipient).	**Security Label** The marking bound to a resource (which may be a data unit) that names or designates the security attributes of that resource.
Access Control A variety of mechanisms that enforce access rights to resources.	**Event Detection** Detection of security-relevant events.
Data Integrity A variety of mechanisms used to assure the integrity of a data unit or stream of data units.	**Security Audit Trail** Data collected and potentially used to facilitate a security audit, which is an independent review and examination of system records and activities.
Authentication Exchange A mechanism intended to ensure the identity of an entity by means of information exchange.	**Security Recovery** Deals with requests from mechanisms, such as event handling and management functions, and takes recovery actions.
Traffic Padding The insertion of bits into gaps in a data stream to frustrate traffic analysis attempts.	
Routing Control Enables selection of particular physically secure routes for certain data and allows routing changes, especially when a breach of security is suspected.	
Notarization The use of a trusted third party to assure certain properties of a data exchange.	

decrypted. Irreversible encipherment mechanisms include hash algorithms and message authentication codes, which are used in digital signature and message authentication applications.

Table 1.4, based on one in X.800, indicates the relationship between security services and security mechanisms.

ASHOK KUMAR K (mob: 9742024066)

A Simple Authentication Dialogue and more secure Authentication Dialogue.

Simple Authentication Dialogue

* Servers must be able to confirm the identities of clients who request service.
* Solution to do this is to use an Authentication server (AS) that knows the passwords of all users and stores these in a centralized database.
* Consider the following hypothetical dialogue.

(1) $C \rightarrow AS$: $ID_c \| P_c \| ID_v$

(2) $AS \rightarrow C$: Ticket

(3) $C \rightarrow V$: $ID_c \| Ticket$

$Ticket = E(K_v, [ID_c \| AD_c \| ID_v])$

where;

C = client
AS = Authentication server
V = server
ID_c = identifier of user on C
ID_v = identifier of V.
P_c = password of user on C
AD_c = n/w address of C
K_v = secret encryption key shared by AS & V

With this ticket, C can now apply to V for

* Here, user logs on and requests access to server V.
  It sends message to AS that includes user ID, server ID, and user's password.
* AS checks its database to see if the user has supplied proper password for this user ID, & whether this user is permitted access to server V.
* If both tests pass, the AS accepts the user as authentic & must now convince the server that this user is authentic.
* To do this, AS creates ticket & encrypts using secret key shared by AS & this server.
* The ticket is then sent back to C.
* With this ticket, C can now apply to V for service.

More Secure Authentication Dialogue

* we have to solve some of the problems of authentication in an open n/w environment.
  1. minimize the no. of times that a user has to enter a password.
  2. plaintext transmission of the password (earlier case, message [1])

once per user logon session

(1) $C \rightarrow AS$: $ID_c \,||\, ID_{tgs}$

(2) $AS \rightarrow C$: $E(K_c, Ticket_{tgs})$

once per type of service

(3) $C \rightarrow TGS$: $ID_c \,||\, ID_v \,||\, Ticket_{tgs}$

(4) $TGS \rightarrow C$: $Ticket_v$

once per service session:

(5) $C \rightarrow V$: $ID_c \,||\, Ticket_v$

$Ticket_{tgs} = E\left(K_{tgs}, [ID_c \,||\, AD_c \,||\, ID_{tgs} \,||\, TS_1 \,||\, lifetime_1]\right)$

$Ticket_v = E\left(K_v, [ID_c \,||\, AD_c \,||\, ID_v \,||\, TS_2 \,||\, lifetime_2]\right)$

ticket granting server

* The new service, TGS, issues tickets to users who have been authenticated to A.S.

1. The client requests a ticket granting ticket on behalf of the user by sending its user's ID. and password to the AS, together with the TGS ID, indicating a request to use the TGS service.
2. AS responds with a ticket that is encrypted with a key that is derived from the user's password.

3. the client requests a service-granting ticket on behalf of the user

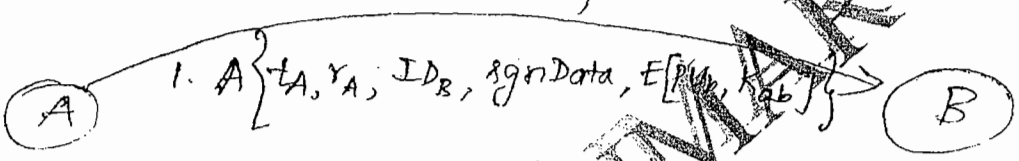
4. TGS decrypts the incoming ticket and verifies the success of the decryption by the presence of its ID. It checks to make sure that the lifetime has not expired. Then it compares the userID & n/w address with the incoming info. to authenticate the user. If the user is permitted access to the server V, the TGS issues a ticket to grant access to the requested service.

5. Client requests access to a service on behalf of the user. For this purpose, the client transmits a message to the server containing the user's ID & the service granting ticket. The server authenticates by using the contents of the ticket.

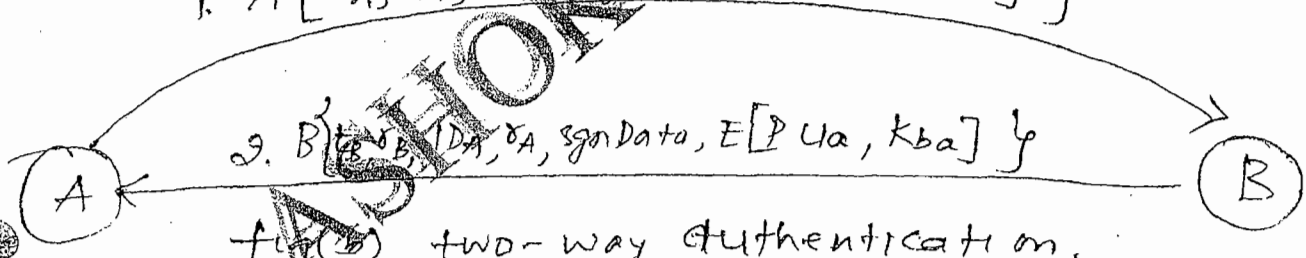
X.509 AUTHSNTICATION SERVICE

Authentication procedures

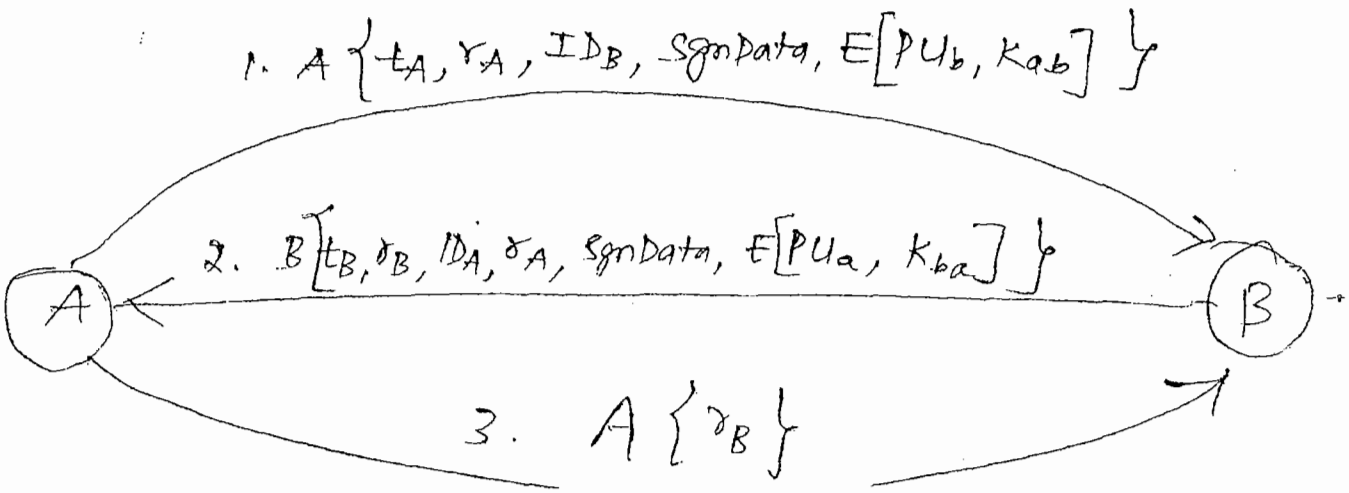
* X.509 includes three alternative authentication procedures that are intented for use across a variety of applications. All these makes use of public key signatures.
* Fig illustrates the three procedures.



fig(a) one-way authentication.



fig(b) two-way authentication.



* It involves a single transfer of information from one user (A) to another (B), and establishes the following.

1. Identity of A and that the message was generated by A.
2. that the message was intended for B.
3. Integrity & originality of the message.

two-way Authentication.

* In addition to above three elements, two-way authentication establishes the following:

4. Identity of B & that the reply message was generated by B.
5. that the message was intended for A.
6. The integrity & originality of the reply.

three way Authentication.

* Here final msg from A to B is included, which contains B's signed copy of the nonce r_B.

notations

t_A = time stamp.
r_A = nonce
signData = information
PU_A = A's public key
PU_B = B's public key.
ID_A = Identity of A
ID_B = Identity of B.

UNIT 6:

ELECTRONIC MAIL SECURITY

Syllabus

* Pretty Good Privacy (PGP)
* S/MIME

— 6 Hours.

PRETTY GOOD PRIVACY (PGP)

* Phil Zimmermann is the creator of PGP.
* PGP provides confidentiality and authentication service that can be used for email and file storage applications.
* He has done the following.

→ Selected the best available cryptographic algo.s

→ Integrated these algo's into a general purpose application, independent of OS & processor

→ Made the package & its documentation, including source code, freely available via Internet

→ Entered into agreement with a company (Network associates) to provide fully compatible, low cost commercial version of PGP.

* Reasons Why PGP has grown explosively & widely used -

→ It is available free on a variety of platforms.

→ Based on well known Algorithms (RSA, DSS, Diffie Hellman, etc)

→ Wide range of applicability.

→ It is now on an Internet standards Track (RFC 3156). Not developed or controlled by governmental or standards organizations

Operational Description:

Function	Algorithms Used	Description
Digital signature	DSS/SHA or RSA/SHA	A hash code of a message is created using SHA-1. This message digest is encrypted using DSS or RSA with the sender's private key and included with the message.
Message encryption	CAST or IDEA or Three-key Triple DES with Diffie-Hellman or RSA	A message is encrypted using CAST-128 or IDEA or 3DES with a one-time session key generated by the sender. The session key is encrypted using Diffie-Hellman or RSA with the recipient's public key and included with the message.
Compression	ZIP	A message may be compressed, for storage or transmission, using ZIP.
Email compatibility	Radix 64 conversion	To provide transparency for email applications an encrypted message may be converted to an ASCII string using radix 64 conversion.
Segmentation		To accommodate maximum message size limitations, PGP performs segmentation and reassembly.

A-1

1. Authentication.
2. Confidentiality
3. Compression.
4. Email compatibility.
5. Segmentation.

~~Authentication~~: Notations:

K_S = Session key used in symmetric encryption scheme
PR_a = private key of user A, used in public key encryption scheme
PU_a = public ——— "
EP = public key encryption
DP = public key decryption
EC = symmetric encryption.
DC = symmetric decryption.
H = Hash function.
|| = concatenation.
Z = compression using ZIP algorithm.
R64 = conversion to radix 64 ASCII format.

Authentication:

* Fig (a) illustrates the digital signature service provided by PGP.
* The sequence is as follows:

1. Sender creates a message
2. SHA-1 is used to generate a 160-bit hash code of the message
3. The Hash code is encrypted with RSA using sender's private key, & the result is prepended to the message.
4. The receiver uses RSA with sender's public key to decrypt and recover the hash code.
5. The receiver generates a new hash code for the message and compares it with the decrypted hash code. If the two match, the message is Authentic.

Confidentiality:

* Confidentiality is provided by encrypting messages to be transmitted or to be stored locally as files. (refer fig b)

1. Sender generates a message and a random 128 bit number to be used as a session key for this message only.
2. The message is encrypted using CAST 128 (or IDEA or 3DES) with the session key.
3. The session key is encrypted with RSA, using the

4. The receiver uses RSA with its private key to decrypt and recover the session key.
5. The session key is used to decrypt the message.

Confidentiality & Authentication:

* refer fig (c), - both services may be used for the same message.
* First, a signature is generated for the plaintex message and pepended to the message. Then, the plain text ~~and~~ message plus signature is encrypted using CAST-128 (or IDEA or 3DES), and the session key is encrypted using RSA or (ElGamal)

Compression:

Z = compression
Z^{-1} = Decompression.

- the signature is generated before compression for two reasons

(a) It is preferable to sign an uncompressed message so that one can store only the uncompressed message together with the signature for future verification.
If one signed a compressed document, then it would be necessary either to store a compressed version of the message for later verification or to decompress the message, when verification

(b) Even if one were willing to generate dynamically a recompressed message for verification, PGP's compression algorithm presents a difficulty.
The algorithm is not deterministic; Various implementations of the algorithm achieve different tradeoffs in running speed versus compression ratio, and as a result, produces different compressed forms.

2. Message encryption is applied after compression to strengthen cryptographic security.

Email Compatibility.
Segmentation and Reassembly } refer table.

Fig (c) shows relationships among PGP services excluding segmentation & Reassembly.
Fig (d) includes segmentation & Reassembly.

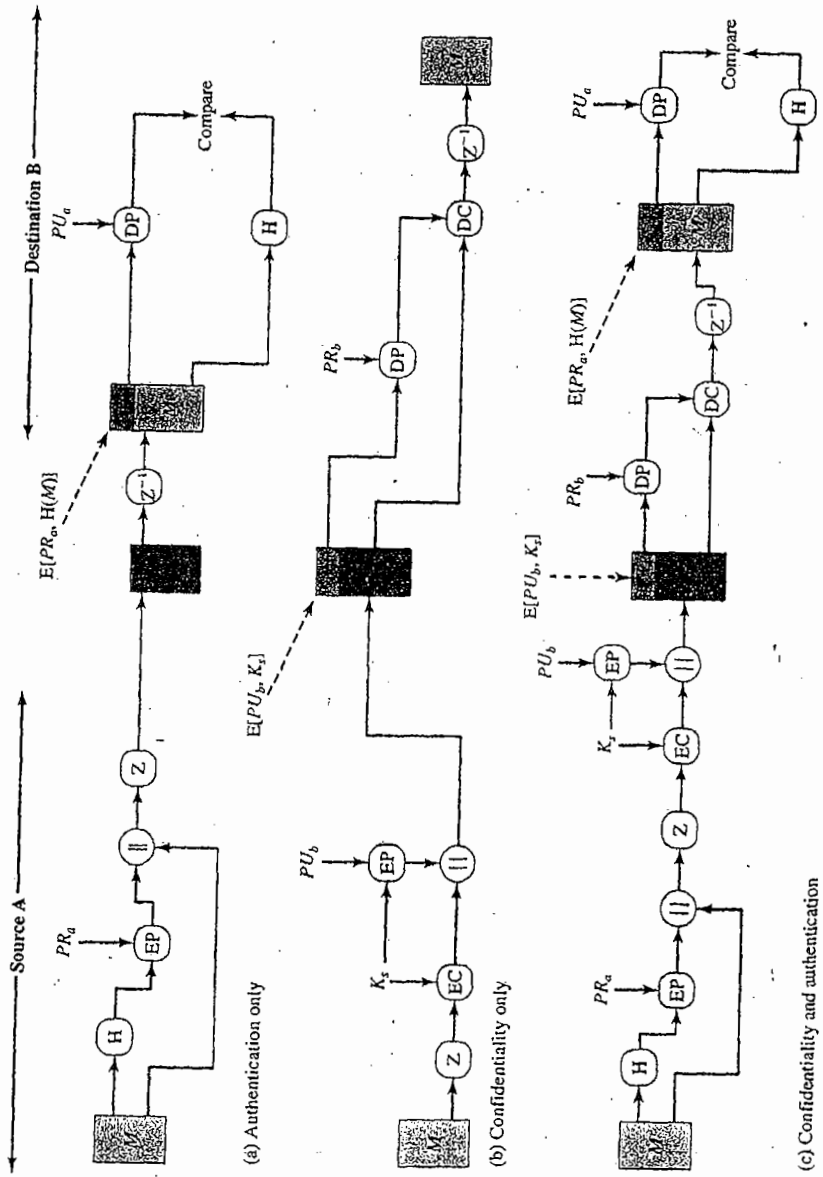
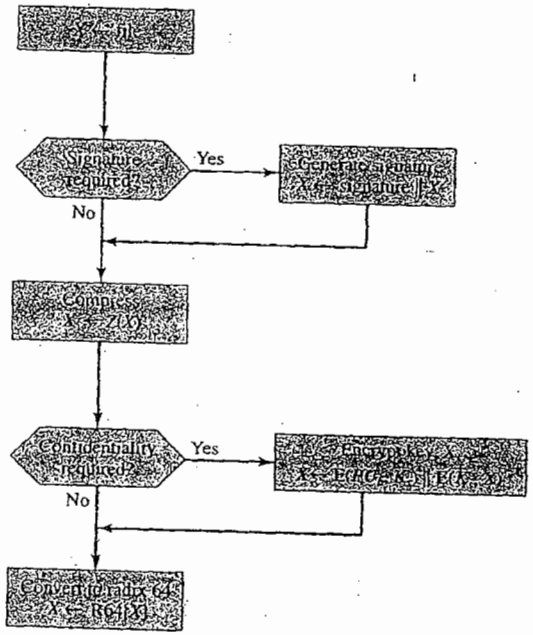
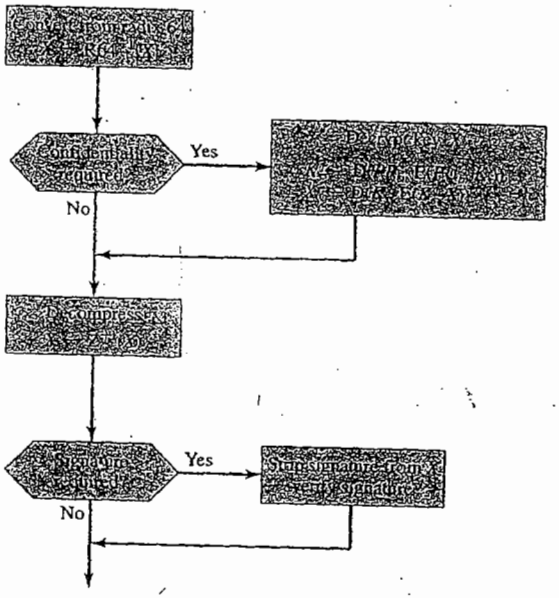


Figure 5.1 PGP Cryptographic Functions



(a) Generic transmission diagram (from A)



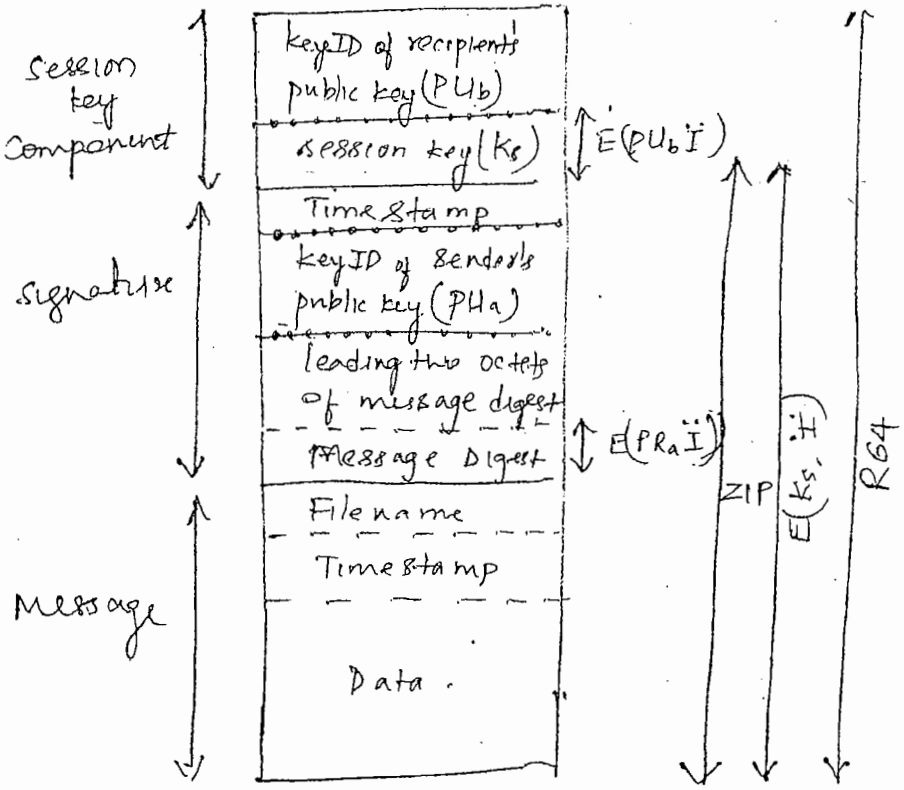
(b) Generic reception diagram (to B)

Figure 5.2 Transmission and Reception of PGP Messages

Cryptographic keys and key Rings.

* PGP makes use of four types of keys:
  - → one-time session symmetric keys.
  - → public keys
  - → private keys
  - → passphrase-based symmetric keys.

General format of PGP message from (A to B)



E(PUb, Ks) = encryption with user b's public key
E(PRa, H) = " " " a's private key.
E(Ks, X) = encryption with session key
ZIP = ZIP compression function
R64 = Radix 64 conversion function

Note 1:

* Session key generation: (using CAST 128)

→ Each session key is associated with a single message and is used only for the purpose of encrypting and decrypting that message.

→ Working:

- Random 128 bit numbers are generated using CAST 128 itself.
- The input to the random number generator consists of a 128 bit key and two 64 bit blocks that are treated as plaintext to be encrypted.
- Using cipher feedback mode, the CAST 128 encrypter produces two 64 bit cipher text blocks, which are concatenated to form the 128 bit session key.

Note 2

* Key Identifiers.

→ WKT any given user may have multiple public/private key pairs.
Then, How does the ~~receiver~~ recipient know which of its public keys was used to encrypt the session key?

Soln 1: Transmit the public key with the message.
Disadv: Waste of space.

Soln 2: Associate an identifier with each public key that is unique atleast within one user.
Disadv: management & overhead problem.

→ The soln adopted by PGP is to assign a key ID to each public key, unique within a user ID. The key ID associated with each public key consists of its least significant 64 bits.

ie the key ID of public key PU_a is $= PU_a \bmod 2^{64}$

→ When the message is received, the recipient verifies that the key ID is for a public key that it knows for that sender and then proceeds to verify the signature.

* A PGP message consists of three components:
  1. the message component
  2. A signature
  3. Session key component

  } optional.

The message component includes the actual data to be stored or transmitted, as well as filename, and a timestamp that specifies the time of creation.

2. The Signature component includes the following:

(a) Time stamp: time at which the signature was made

(b) Message digest:

It is 160-bit SHA-1 digest, encrypted with the sender's private signature key.
The inclusion of signature timestamp in the digest assures against replay types of attacks
The exclusion of the filename and timestamp portions of the message component ensures that detached signatures are exactly the same as attached signatures prefixed to the message.

(c) Leading two octets of message digest:

To enable the recipient to determine if the correct public key was used to decrypt the message digest for authentication, by comparing this plaintext copy of the first two octets with the first two octets of the decrypted digest.
These octets also serves as a 16-bit FCS for the messg.

d) Key ID of sender's public key:

Identifies the public key that should be used to decrypt the message digest, & hence identifies the private key that was used to encrypt the message digest.

3. The session key component includes the session key and the identifier of the recipient's public key that was used by the sender to encrypt the session key.

* the entire block is usually encoded with radix 64 encoding.

Key Rings.

* How the key IDs are stored and organised in PGP ?

⇒ The scheme used in PGP is to provide a pair of data structures at each node, one to store the public/private key pairs owned by that node and one to store the public keys of other users known at this node.

⇒ These data structures are referred to as private key ring and the public key ring respectively.

* Fig shows the general structure of a private-key and public-key ring.

plz refer fig in textbook page 143

* Each row in private key ring consists of:
  - → Time stamp: date/time when this key pair was generated.
  - → keyID: the least significant 64 bits of the public key for this entry.
  - → public key: the public-key portion of the pair.
  - → private key: <encrypted> private-key portion of the pair.
  - → userID: Typically, the user's email address.

* Each row in public key consists many fields.
  lets ignore few fields
  - → Time stamp
  - → keyID
  - → Public keys

  } same as above.

  - → userID: Identifies the owner of this key. May be associated with multiple userIDs.

Explain PGP message Generation and PGP message Reception from user A to user B.

PGP message Generation

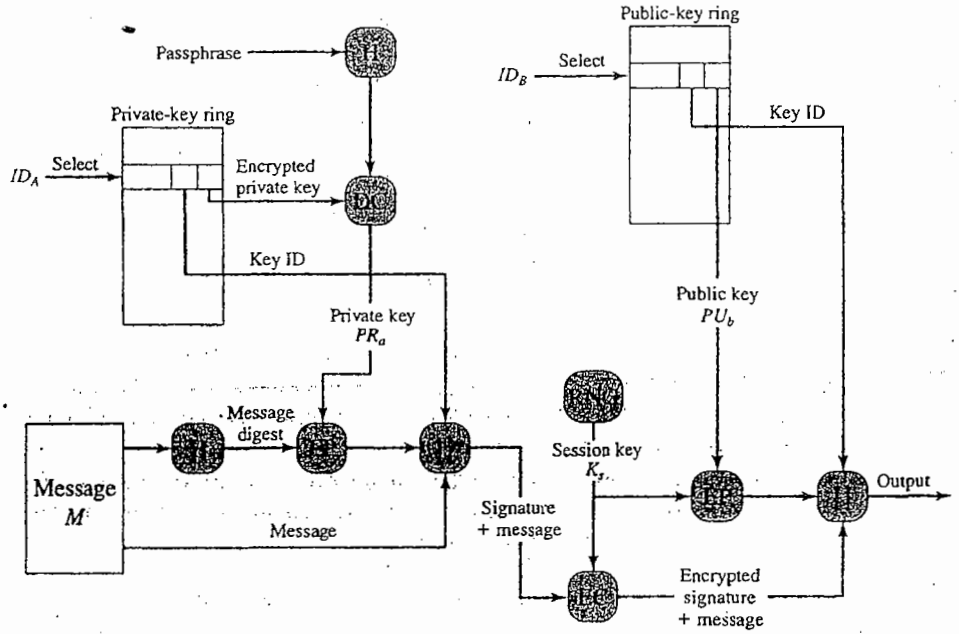


Figure 5.5 PGP Message Generation (from User A to User B; no compression or radix 64 conversion)

* the sending PGP entity performs the following steps:

1. Signing the message:
   a.) PGP retrieves the sender's private key from the private key ring using your_userid as an index. If your_userid is not provided in the command, the first private key on the ring is retrieved.
   b.) PGP prompts the user for the passphrase to recover the unencrypted private key.
   c.) the signature component of the message is constructed.

2. Encrypting the message.

a) PGP generates a session key and encrypts the message.

b.) PGP retrieves the recipient's public key from the public-key ring using her-userid as an index.

c.) The session key component of the message is constructed.

PGP message Reception.

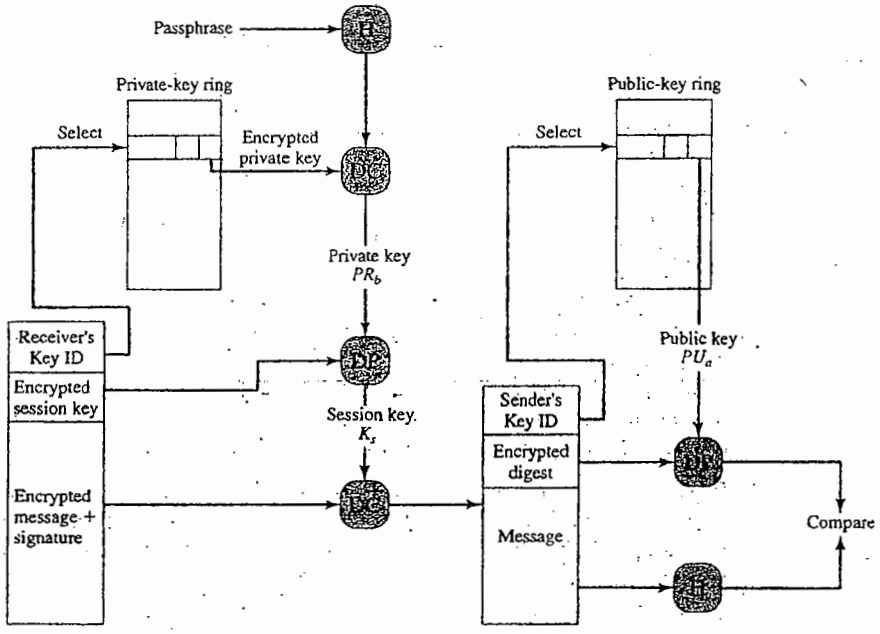


Figure 5.6 PGP Message Reception (from User A to User B; no compression or radix 64 conversion)

* The receiving PGP entity performs the following steps:

1. Decrypting the message

a) PGP retrieves the receiver's private key from private-key ring, using the key ID field in the session key component of the message as index.

b.) PGP prompts the user for the passphrase to recover the unencrypted private key.

c.) PGP then recovers the session key and decrypts the message.

2. Authenticating the message.

a.) PGP retrieves the sender's public key from the public key ring, using the key ID field in the signature key component of the message as index

b.) PGP recovers the transmitted message digest.

c.) PGP computes the message digest for the received message and compares it to the transmitted message digest to authenticate.

public key management

problem: The whole business of protecting public keys from tampering is the single most problem difficult in practical public key applications.
It is the "Achilles heel" of the public key cryptography.

Solution:
there are no. of approaches for minimizing the risk that a user's public key ring contains false public keys.
suppose that A wishes to obtain a reliable public key for B, the following are some approaches that could be used:

1. physically get the key from B. (thru floppy disk and hand it to A.)
2. Verify a key by Telephone, or, B could transmit her key in an email message.
3. obtain B's public key from a mutual trusted individual D. For this purpose, the introducer, D creates a signed certificate, which includes B's public key, time of creation, validity period.
4. obtain B's public key from an trusted certifying authority. Again, a public key certificate is created and signed by the authority.

For cases 3 and 4, it is upto A to assign a level of trust to anyone who is to act as an introducer.

The Use of trust

* A key legitimacy field is associated with public key certificate entry in public key ring. This field indicates the extent to which PGP will trust that this is a valid public key for this user.
* A signature trust field is associated with each signature, that indicates the degree to which this PGP user trusts the signer to certify public keys.
* Finally, each entry (in public key ring) defines a public key associated with a particular owner, and an owner trust field is included that indicates the degree to which this public key is trusted to sign other public-key certificates; this level of trust is assigned by the user.
* These three fields ~~mentioned~~ are each contained in a structure referred to as a trust flag byte. The content of this trust flag for each of these three uses is shown in below table.

Table 5.2 Contents of Trust Flag Byte

(a) Trust Assigned to Public Key Owner (appears after key packet; user defined)	(b) Trust Assigned to Public Key/User ID Pair (appears after User ID packet; computed by PGP)	(c) Trust Assigned to Signature (appears after signature packet; cached copy of OWNERTRUST for this signator)
OWNERTRUST Field —undefined trust —unknown user —usually not trusted to sign other keys —usually trusted to sign other keys —always trusted to sign other keys —this key is present in secret key ring (ultimate trust) BUCKSTOP bit —set if this key appears in secret key ring	KEYLEGIT Field —unknown or undefined trust —key ownership not trusted —marginal trust in key ownership —complete trust in key ownership WARNONLY bit —set if user wants only to be warned when key that is not fully validated is used for encryption	SIGTRUST Field —undefined trust —unknown user —usually not trusted to sign other keys —usually trusted to sign other keys —always trusted to sign other keys —this key is present in secret key ring (ultimate trust) CONTIG bit —set if signature leads up a contiguous trusted certification path back to the ultimately trusted key ring owner

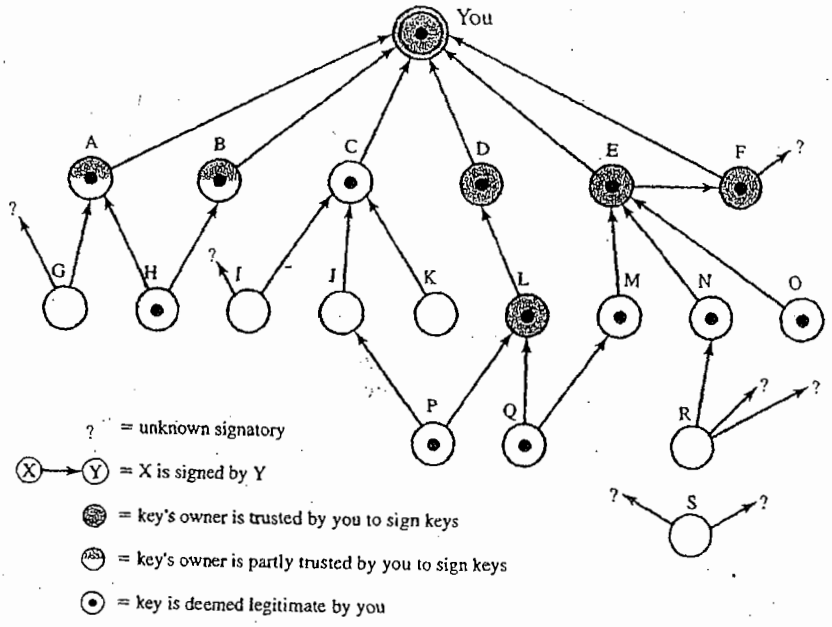


Figure 5.7 PGP Trust Model Example

Revoking public keys:

* A user may wish to revoke his current public key either because compromise is suspected or simply to avoid the use of same key for an extended period.

* Steps:

1. Owner issues a key revocation certificate, which is signed.
   - this certificate has the same form as normal certificate but includes an indicator that the purpose of this certificate is to revoke the use of this public key.
2. Corresponding private key is used to sign a certificate that revokes a public key.
3. The owner then attempts to disseminate this certificate as widely and as quickly as possible to enable potential correspondents to update their public key rings.

S/MIME

[Secure/Multipurpose Internet Mail Extension]

* S/MIME is a security enhancement to the MIME Internet email format standard, based on technology from RSA data security.
  → S/MIME will emerge as the industry standard for commercial and organizational use.
  → PGP will remain the choice for personal e-mail security for many users.

* To understand S/MIME, first we have to understand MIME.

~~Before~~

note: RFC 822 is the traditional email format.

ex:-

```
Date: Tue, 16 Jan 1998 10:37:17 (EST)
From: "William Stallings" <ws@shore.net>
Subject: The Syntax in RFC 822
To: Smith@Other-host.com
Cc: Jones@Yet-Another-Host.com

Hello. This section begins the actual
message body, which is delimited from the
message heading by a blank line.
```

Multipurpose Internet Mail Extensions (MIME)

* MIME is an extension to the RFC 822 that is intended to address some of the problems and limitations of the use of SMTP (simple mail transfer protocol)

* Some limitations of SMTP/822 scheme are:

→ SMTP cannot transmit executable files,
or other binary objects (like JPEG image)

→ SMTP cannot transfer text data that includes national language characters (non-ASCII)

→ SMTP servers may reject email message over a certain size.

→ SMTP gateways that translate b/w ASCII and the character code EBCDIC do not use consistent set of mappings, resulting in translation problems.

SMTP gateways to X.400 email n/ws cannot handle nontextual data included in X.400 messages

→ SMTP implementations do not adhere completely to the SMTP standards defined in RFC 821. Common problems includes:

- Deletion, addition, or reordering of carriage return/line fe
- Truncating lines longer than 76 characters.
- Removal of trailing white space.
- Padding of lines in a message to same length

Overview of MIME.

* the MIME specification includes the following elements

1. Five new message header fields are defined, which provide info. about the body of the message
2. A number of content formats are defined, thus standardizing representations that support multimedia e-mail.
3. Transfer encodings are defined that enable the conversion of any content format into a form that is protected from alteration by the mail System.

Message Header Fields.

* The Five header fields defined in MIME are as follows:

→ MIME version: Indicates that the message conforms to RFCs 2045 & 2046. It must have the parameter value 1.0.

→ Content type: Describes the data contained in the body with sufficient detail that the receiving user agent can pick an appropriate agent or mechanism to represent the data to the user or otherwise deal with the data in an appropriate manner.

→ Content transfer encoding: indicates the type of transformation that has been used to represent the body of the message in a way that is acceptable for e-mail transport.

→ Content-ID: Used to identify MIME entities uniquely in multiple contexts.

→ Content-Description: A text description of the object with the body; this is useful when the object is not readable (eg: audio data.

MIME Content types.

* A) content type ~~describes~~ declares the general type of data, and the subtype specifies a particular format for that type of data.

MIME Transfer Encodings

* It is a definition of transfer encodings for message bodies.
The objective is to provide reliable delivery across the largest range of environments.
* MIME std defines two methods of encoding data.
* The content-transfer-Encoding field can actually take on six values, as listed below.

Table 5.4 MIME Transfer Encodings

7bit	The data are all represented by short lines of ASCII characters.
8bit	The lines are short, but there may be non-ASCII characters (octets with the high-order bit set).
binary	Not only may non-ASCII characters be present, but the lines are not necessarily short enough for SMTP transport.
quoted-printable	Encodes the data in such a way that if the data being encoded are mostly ASCII text, the encoded form of the data remains largely recognizable by humans.
base64	Encodes data by mapping 6-bit blocks of input to 8-bit blocks of output, all of which are printable ASCII characters.
x-token	A named nonstandard encoding.

* Below table shows difference b/w native and canonical forms.

Table 5.5 Native and Canonical Form

Native Form	The body to be transmitted is created in the system's native format. The native character set is used and, where appropriate, local end-of-line conventions are used as well. The body may be a UNIX-style text file, or a Sun raster image, or a VMS indexed file, or audio data in a system-dependent format stored only in memory, or anything else that corresponds to the local model for the representation of some form of information. Fundamentally, the data is created in the "native" form that corresponds to the type specified by the media type.
Canonical Form	The entire body, including "out-of-band" information such as record lengths and possibly file attribute information, is converted to a universal canonical form. The specific media type of the body as well as its associated attributes dictate the nature of the canonical form that is used. Conversion to the proper canonical form may involve character set conversion, transformation of audio data, compression, or various other operations specific to the various media types. If character set conversion is involved, however, care must be taken to understand the semantics of the media type, which may have strong implications for any character set conversion (e.g., with regard to syntactically meaningful characters in a text subtype other than "plain").

S/MIME Functionality.

* In terms of general functionality, S/MIME is very similar to PGP. Both offer the ability to sign and/or encrypt messages.

* S/MIME provides the following functions.

→ Enveloped Data: Consists of encrypted contents and encrypted session keys for recipients.

→ Signed Data: A digital signature is formed by taking the message digest of the content to be signed and then encrypting that with the private key of the signer.

→ clear-signed Data: Signed, but not encrypted.

→ Signed and Enveloped Data: Signed-only and encrypted-only entities may be nested, so that encrypted data may be signed and signed data or clear-signed data may be encrypted.

Cryptographic Algorithms

* Table below summarizes the cryptographic algorithms used in S/MIME.

Table 5.6 Cryptographic Algorithms Used in S/MIME

Function	Requirement
Create a message digest to be used in forming a digital signature.	MUST support SHA-1. Receiver SHOULD support MD5 for backward compatibility.
Encrypt message digest to form digital signature.	Sending and receiving agents MUST support DSS. Sending agents SHOULD support RSA encryption. Receiving agents SHOULD support verification of RSA signatures with key sizes 512 bits to 1024 bits.
Encrypt session key for transmission with message.	Sending and receiving agents SHOULD support Diffie-Hellman. Sending and receiving agents MUST support RSA encryption with key sizes 512 bits to 1024 bits.
Encrypt message for transmission with one-time session key.	Sending and receiving agents MUST support encryption with triple DES. Sending agents SHOULD support encryption with AES. Sending agents SHOULD support encryption with RC2/40.
Create a message authentication code.	Receiving agents MUST support HMAC with SHA-1. Receiving agents SHOULD support HMAC with SHA-1.

* S/MIME uses the following terminology to specify the requirement level:

→ **must**: The definition is an absolute requirement of the specification. An implementation must include this feature or function to be in conformance with the specification.

→ **should**: There may exists valid reasons in particular circumstances to ignore this feature/function, but it is recommended that an implementation include the feature/function.

S/MIME messages.

* S/MIME makes use of a number of new MIME content types, which are shown below:

Table 5.7 S/MIME Content Types

Type	Subtype	smime Parameter	Description
Multipart	Signed		A clear signed message in two parts: one is the message and the other is the signature.
Application	pkcs7-mime	signedData	A signed S/MIME entity.
	pkcs7-mime	envelopedData	An encrypted S/MIME entity.
	pkcs7-mime	degenerate signedData	An entity containing only public-key certificates.
	pkcs7-mime	CompressedData	A compressed S/MIME entity.
	pkcs7-signature	signedData	The content type of the signature subpart of a multipart/signed message.

Securing a MIME entity:

S/MIME secures a MIME entity with a signature, encryption, or both.

The MIME entity is prepared according to the normal rules for MIME message preparation.

Then, the MIME entity plus some security-related data are processed by S/MIME to produce what is known as a PKCS object.

A PKCS object is then treated as a message content & wrapped in MIME

Enveloped Data:

* An application/pkcs7-mime subtype is used for one of four categories of S/MIME processing, each with a unique smime-type parameter. In all cases, the resulting entity referred to as an object., is represented in a form known as

* Steps for preparing an envelopedData MIME entity are as follows:

1. Generate a pseudo random session key for a particular symmetric encryption algorithm (RC2/40 or triple DES)
2. For each recipient, encrypt the session key with the recipient's public RSA key.
3. For each recipient, prepare a block known as RecipientInfo that contains an identifier of the recipient's public-key certificate, an identifier of the algorithm used to encrypt the session key, and the encrypted session key.
4. Encrypt the message content with the session key.

SignedData:

* The steps for preparing a signedData MIME entity are as follows:

1. Select a message digest Algorithm (SHA or MD5)
2. Compute the message digest, or hash function, of the content to to be signed.
3. Encrypt the message digest with the signer's private key.
4. Prepare a block known as SignerInfo that contains the signer's public key certificate, an Identifier of the message digest Algorithm, an identifier of the algorithm used to encrypt the message digest,

Examples:

→ Sample message for Enveloped Data:

Content-Type: application/pkcs7-mime; smime-type=enveloped-data; name=smime.p7m

Content-Transfer-Encoding: base64

Content-Disposition: attachment; filename=smime.p7m

→ Sample message for SignedData:

Content-Type: application/pkcs7-mime; smime-type=signed-data; name=smime.p7m

Content-Transfer-Encoding: base64

Content-Disposition: attachment; filename=smime.p7m

Clear signing:

* clear signing is achieved using the multipart content type with a signed subtype.

Registration Request:

* Typically, an application or user will apply to a certification authority for a public key certificate. The application/pkcs10 S/MIME entity is used to transfer a certification request.

Certificates-only messages:

* A message containing only certificates or a certificate revocation List (CRL) can be sent in response to a registration request.

S/MIME Certificate Processing.

* S/MIME managers and/or users must configure each client with a list of trusted keys and with certificate revocation lists.
that is, the responsibility is local for maintaining the certificates needed to verify incoming signatures and to encrypt outgoing messages.
On the other hand, the certificates are signed by Certification authorities.

User Agent Role.

- An S/MIME user has several key-management functions to perform:

› Key generation: The user ~~must~~ MUST be capable of generating separate Diffie-Hellman and DSS key pairs and SHOULD be capable of generating RSA key pairs.

→ Registration: A user's public key must be registered with a certification authority in order to receive an X.509 public key certificate

→ Certificate storage and retrieval: A user requires access to a local list of certificates in order to verify incoming signatures and to encrypt outgoing messages.

(certification authority)

Verisign Certificates:

* There are several companies that provide CA services.
* Verisign provides a CA service that is intended to be compatible with S/MIME & a variety of other applications.
* Verisign provides three levels or classes of security for public-key certificates as summarized below:

	Summary of Confirmation of Identity	IA Private Key Protection	Certificate Applicant and Subscriber Private Key Protection	Applications implemented or contemplated by Users
Class 1	Automated unambiguous name and e-mail address search	PCA: trustworthy hardware CA: trustworthy software or trustworthy hardware	Encryption software (PIN protected) recommended but not required	Web browsing and certain e-mail usage
Class 2	Same as Class 1 plus automated enrollment information check plus automated address check	PCA and CA: trustworthy hardware	Encryption software (PIN protected) required	Individual and intra- and inter-company E-mail, online subscriptions, password replacement, and software validation
Class 3	Same as Class 1 plus personal presence and ID document plus Class 2 automated ID check for individuals; business record (or filings) for organizations	PCA and CA: trustworthy hardware	Encryption software (PIN protected) required; hardware token recommended but not required	E-banking, corporate database access, personal banking, membership-based online services, content integrity services, e-commerce server software validation, authentication of LRAAs, and strong encryption for certain servers

IA Issuing Authority
CA Certification Authority

* A user requests a certificate online at verisign's Web site or other participating website.

→ class 1: Buyer's email address confirmed by emailing vital info.

→ class 2: postal address is confirmed as well, and data checked against directories.

→ class 3: Buyer must appear in person, or send notarized documents.

Enhanced Security Services:

1. Signed Receipts: A signed receipt may be requested in a Signed Data object. (ie proof of delivery to the originator of the message)

2. Security Labels: may be included in the authenticate ~~at~~ attributes of a Signed Data object.
A security label is a set of security information regarding the sensitivity of the content that is protected by S/MIME encapsulation.

3. Secure mailing lists: when a user sends a message to multiple recipients, a certain amount of per-recipient processing is required, including the use of each recipient's public key.
The user can be ~~relieve~~ relieved of this work by employing the services of an S/MIME Mail List Agent (MLA).

UNIT 7: IP SECURITY

Syllabus

* IP security overview
* IP security Architecture
* Authentication Header
* Encapsulating security payload
* combining security Associations
* key management

— 6 Hours.

* IP security (IPSec) is a capability that can be added to either current version of the internet protocol (IPV4 or IPV6) by means of additional headers.

* IPSec encompasses three functional areas:
  - Authentication
  - Confidentiality
  - key management

→ Authentication makes use of the HMAC message authentication code.
Authentication can be applied to the entire original IP packet (tunnel mode) or to all of the packet except for the IP header (transport mode)

→ Confidentiality is provided by an encryption format known as Encapsulating Security payload. Both tunnel and transport mode can be accomodated.

→ IPsec defines a number of techniques for key management.

note: Why do we need IP level security ?
⇒ By implementing security at the IP level, an organization can ensure secure networking not only for applications that have security mechanisms but also for the many security-ignorant applications.

IP SECURITY OVERVIEW

* IPsec is not a single protocol. Instead, IPsec provides a set of security algorithms plus a general framework that allows a pair of communicating entities to use whichever algorithms provide security appropriate for the communication.

Applications of IPsec

* IPsec provides the capability to secure communication across a LAN, across private and public WANs, and across the internet.
* Examples of use of IPsec
  - → Secure branch office connectivity over the internet
    - A company can build a secure virtual private network over the internet or over public WAN.
  - → Secure remote access over the internet
    - An end user whose system is equipped with IPsec protocols can make a local call to an ISP and gain secure access to a company n/w.
  - → Establishing extranet and intranet connectivity with partners
    - IPsec can be used to secure communication with other organisation, ensuring authentication & confidentiality & providing a key exchange mechanism
  - → Enhancing Electronic commerce security

Fig below is a typical scenario of IPsec usage.

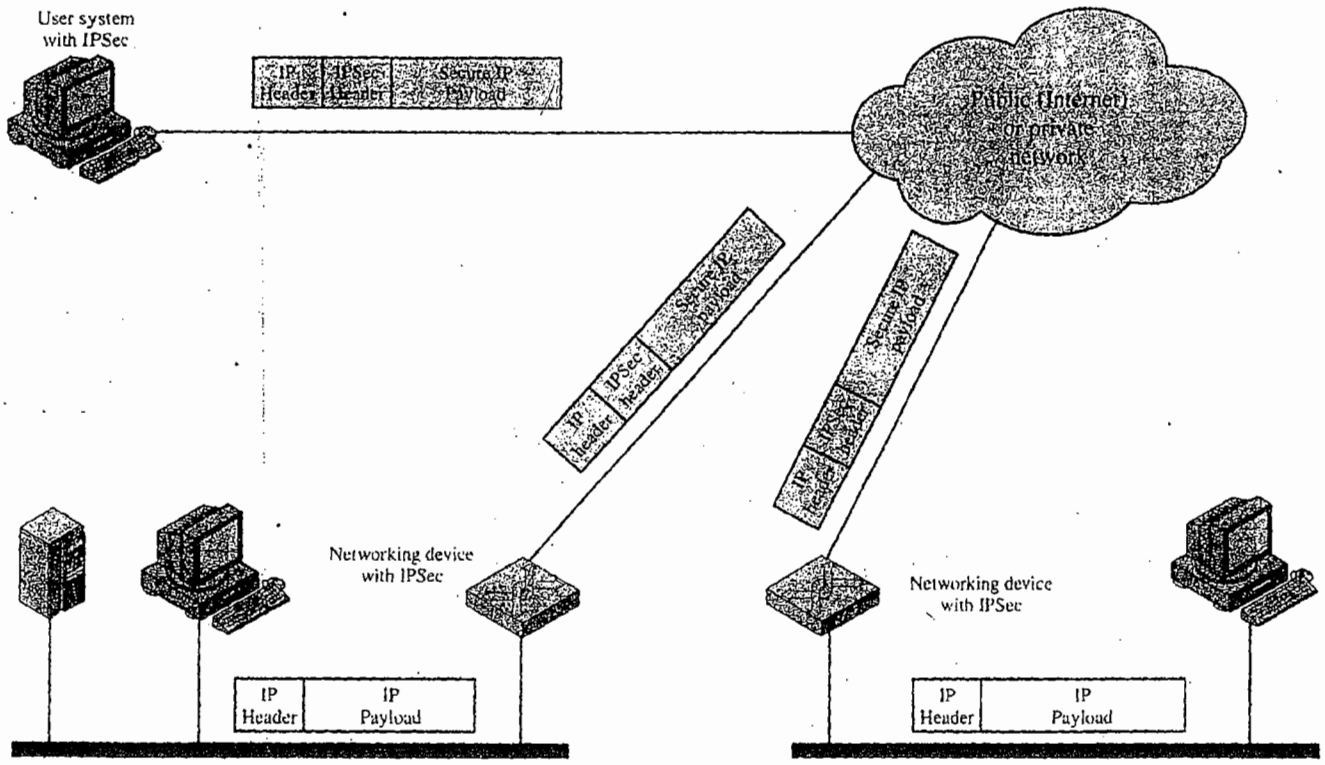


Figure 6.1 An IP Security Scenario

Benefits of IPsec

1. When IPsec is implemented in a firewall or router, it provides strong security that can be applied to all traffic crossing the perimeter.
2. IPsec in a firewall is resistant to bypass if all traffic from the outside must use IP, and the firewall is the only means of entrance from the internet into the organisation.
3. IPsec is transparant to applications. ie it is below the transport layer (TCP, UDP). There is no need to change software on a user or server system when IPsec is implemented in the firewall or router.
4. IPsec can be transparant to end users. There is no need to train users on security mechanisms etc
5. IPsec provides security for individual users if needed

Routing Applications

IPsec can assure that:

1. A router advertisement (a new router advertises its presence) comes from an authorised router.
2. A neighbor advertisement comes from an authorized router.
3. A redirect message comes from the router to which the initial packet was sent.
4. A routing update is not forged.

IP SECURITY ARCHITECTURE

IPSec Documents

* IPSec specification consists of numerous documents

→ RFC 2401 : An overview of security Architecture

→ RFC 2402 : Description of a packet authentication extension to IPV4 and IPV6.

→ RFC 2406 : Description of a packet encryption extension to IPV4 and IPV6.

→ RFC 2408 : Specification of key mgmt capabilities

* The documents are divided into 7 groups as shown :

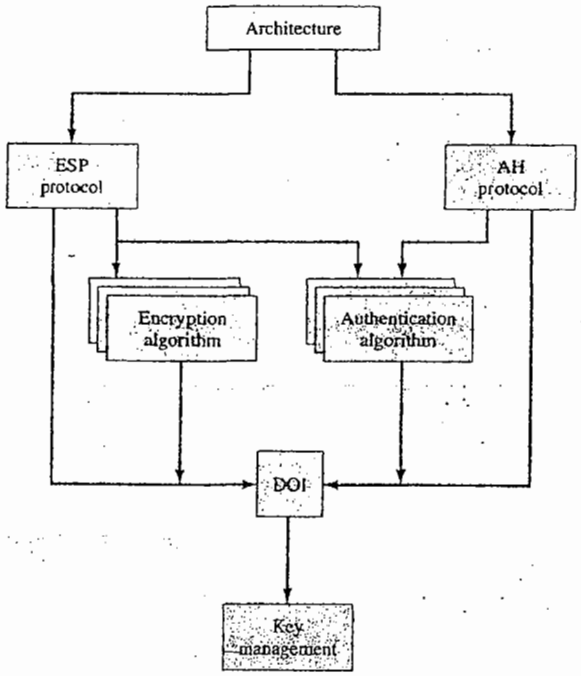


Figure 6.2 IPSec Document Overview

→ Architecture:
Covers the general concepts, security requirements, definitions, and mechanisms defining IPSec technology.

→ Encapsulating Security payload (ESP):
Covers the packet format and general issues related to the use of the ESP for packet encryption and, optionally, authentication.

→ Authentication Header (AH):
Covers the packet format and general issues related to the use of AH for packet authentication

→ Encryption Algorithm:
A set of documents that describe how various encryption algorithms are used for ESP.

→ Authentication Algorithm:
A set of documents that describe how various authentication algorithms are used for AH and for the authentication option of ESP.

→ Key management:
Documents that describe key mgmt schemes.

→ Domain of Interpretation (DOI):
Contains values needed for the other documents to relate to each other. These include identifiers for approved encryptions and authentication

IPSec Services.

* Below table shows which services are provided by AH and ESP protocols. For ESP, there are two cases; with and without the authentication option.

Table 6.1 IPSec Services

	AH	ESP (encryption only)	ESP (encryption plus authentication)
Access control	✓	✓	✓
Connectionless integrity	✓		✓
Data origin authentication	✓		✓
Rejection of replayed packets	✓	✓	✓
Confidentiality		✓	✓
Limited traffic flow confidentiality		✓	✓

Security Associations (SA)

* An association is a one way relationship between a sender and a receiver that affords security services to the traffic carried on it.
* SA is identified by three parameters.

1. Security parameters Index (SPI):
   A bit string, having local significance only. SPI is carried in AH and ESP headers to enable to receiving system to select the SA under which a received packet will be processed.

2. IP destination Address:
This is the address of the destn endpoint of the SA.

3. Security protocol Identifier:
Indicates whether the association is an AH or ESP security association.

SA parameters.

1. Sequence no. counter : 32 bit value used to generate the sequence no. field in AH or ESP headers.

2. sequence counter overflow: Flag indicating whether overflow of the sequence no. counter should generate an auditable event and prevent further transmission of packets on this SA.

3. Anti-replay window: Used to determine whether an inbound AH or ESP packet is a replay.

4. AH information: Authentication algo, keys, keys lifetimes, & related parameters being used with AH.

5. ESP information: Encryption and authentication algo, keys, initialization values, key lifetimes & related parameters being used with ESP.

6. Lifetime of this SA: A time interval or byte count after which an SA must be replaced with a new SA (& new SPI) or terminated, plus an indication of which of these actions should occur.

7. IPSec protocol mode: Tunnel, transport, or wildcard.

8. Path MTU: Any observed path maximum transmission unit and aging variables.

SA selectors:

* the means by which IP traffic is related to specific SAs is the nominal Security Policy Database (SPD)
  In its simplest form, an SPD contains entries, each of which defines a subset of IP traffic and points to an SA for that traffic.
* Each SPD entry is defined by a set of IP and upper layer protocol field values called selectors.
* the following selectors determine an SPD entry:
  - Destination IP address
  - Source IP address
  - User ID: User ID from OS.
  - Data sensitivity level: used for providing info flow security
  - Transport layer protocol:

Transport & Tunnel modes

* Transport mode
  → provides protection primarily for upper layer protocols.
* Tunnel mode
  → provides protection to the entire IP packet.

Table 6.2 Tunnel Mode and Transport Mode Functionality

	Transport Mode SA	Tunnel Mode SA
AH	Authenticates IP payload and selected portions of IP header and IPv6 extension headers.	Authenticates entire inner IP packet (inner header plus IP payload) plus selected portions of outer IP header and outer IPv6 extension headers.
ESP	Encrypts IP payload and any IPv6 extension headers following the ESP header.	Encrypts entire inner IP packet.
ESP with Authentication	Encrypts IP payload and any IPv6 extension headers following the ESP header. Authenticates IP payload but not IP header.	Encrypts entire inner IP packet. Authenticates inner IP packet.

AUTHENTICATION HEADER (AH)

* provides support for data integrity and Authentication of IP packets.
  → Data integrity ensures that undetected modification to a packet's content in transit is not possible.
  → Authentication enables an end system or N/W device to authenticate the user or application and filter traffic accordingly.
* It also prevents the address spoofing attacks and guards against the replay attack.

* AH consists of following fields.

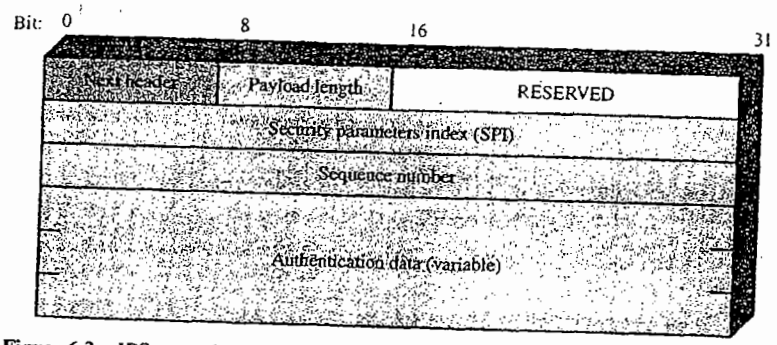


Figure 6.3 IPSec Authentication Header

The Authentication Header consists of the following fields (Figure 6.3):

- **Next Header (8 bits):** Identifies the type of header immediately following this header.
- **Payload Length (8 bits):** Length of Authentication Header in 32-bit words, minus 2. For example, the default length of the authentication data field is 96 bits, or three 32-bit words. With a three-word fixed header, there are a total of six words in the header, and the Payload Length field has a value of 4.
- **Reserved (16 bits):** For future use.
- **Security Parameters Index (32 bits):** Identifies a security association.
- **Sequence Number (32 bits):** A monotonically increasing counter value, discussed later.

Anti Replay service.

* A replay attack is one in which an attacker obtains a copy of an authenticated packet and later transmits it to the intended destination.

* Antireplay mechanism
  - → Generation of sequence number by sender.
  - → processing of this sequence no. by recipient.

* generation of sequence number by sender.

→ When a new SA is established, the sender initializes a sequence no. counter to 0. Each time that a packet is sent on this SA, the sender increments the counter and places the value in the sequence number field.

→ If anti replay is enabled (default), sender must not allow the sequence no. to cycle past $2^{32}-1$ back to zero.

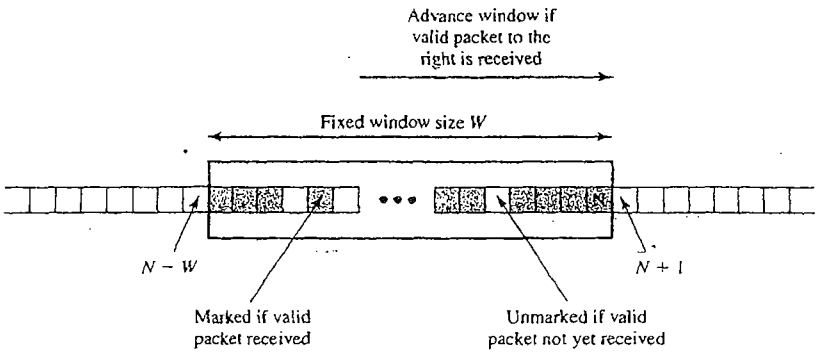


Figure 6.4 Antireplay Mechanism

→ IPSec Authentication document dictates that the receiver should implement a window of size W, with default of W = 64.

→ the right edge of the window represents highest sequence no. N, so far received for a valid packet.

→ For any packet with a sequence no. in the range from N-W+1 to N that has been correctly received (ie authenticated), the corresponding slot in the window is marked.

* When a packet is received, the inbound processing proceeds as follows

1. If the received packet falls within the window and is new, the MAC is checked. If the packet is authenticated, the corresponding slot in the window is marked.

2. If the received packet is to the right of the window and is new, the MAC is checked. If it is authenticated, the window is advanced so that this seq. no. is the right edge of the window, and the corresponding slot in the window is marked.

3. If the received packet is to the left of the window, or if authentication fails, the packet is discarded; this is an auditable even

Transport and Tunnel modes AH

* Fig shows two ways in which the IPSec authentication service can be used.

1. End to End Authentication

→ Here, authentication is provided directly b/w a server and client workstations.

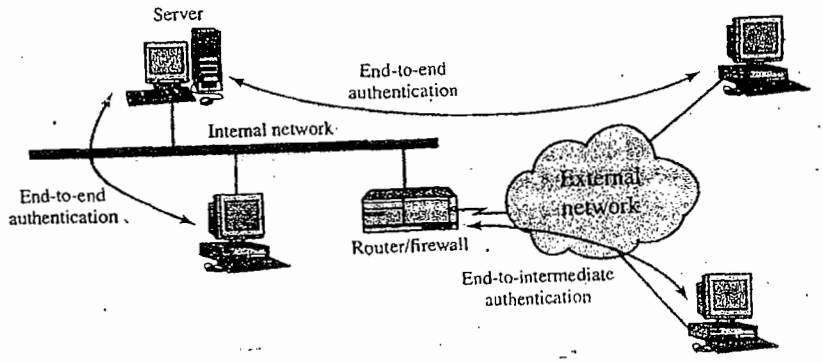
→ As long as the workstation & server share a protected secret key, the authentication process is secure.

→ This uses a transport mode SA

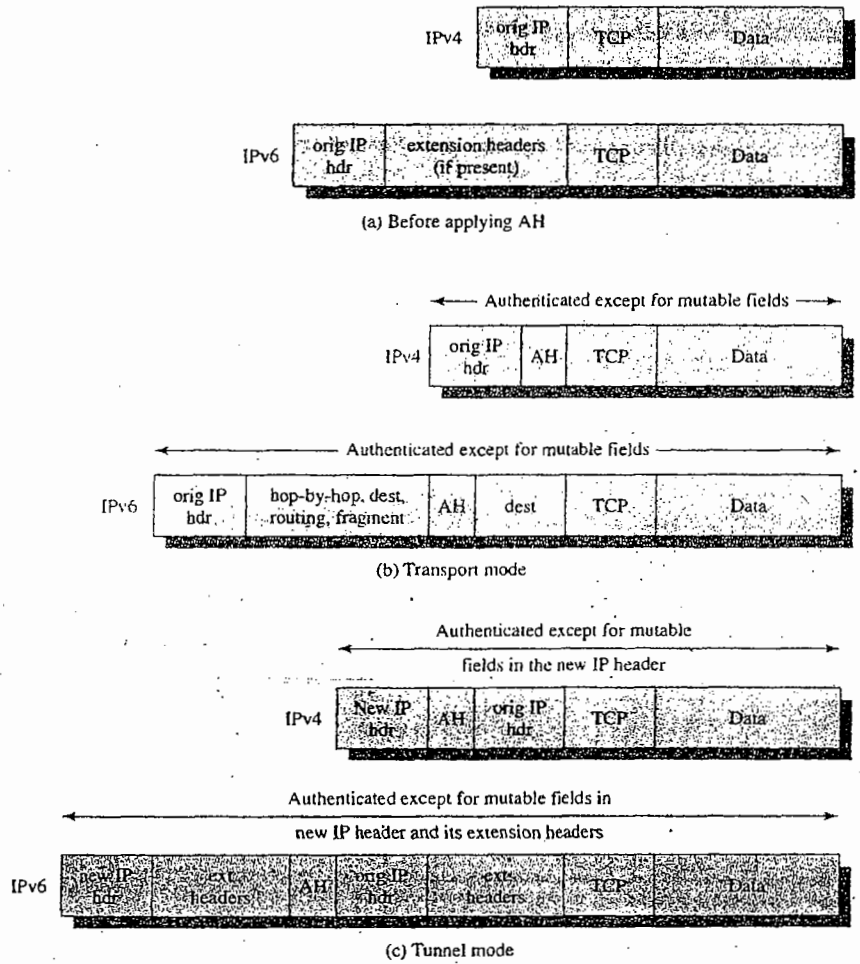
2. End to intermediate Authentication

→ Here, a remote workstation authenticates itself to the corporate firewall, either for access to the entire internal n/w or because the requested server does not support the authentication feature.

→ This uses a tunnel mode SA.



Transport mode AH and Tunnel mode AH

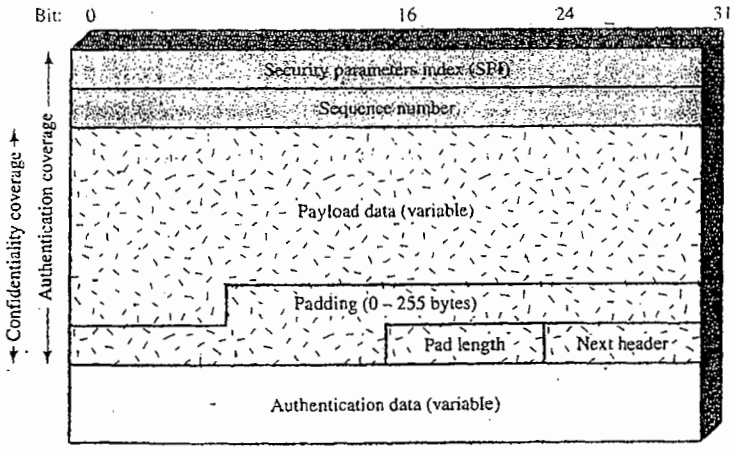


ENCAPSULATING SECURITY PAYLOAD (ESP)

* ESP provides confidentiality services, including confidentiality of message contents and limited traffic flow confidentiality.
Optionally, it can provide authentication service.

ESP Format

- **Security Parameters Index (32 bits):** Identifies a security association.
- **Sequence Number (32 bits):** A monotonically increasing counter value; this provides an anti-replay function, as discussed for AH.
- **Payload Data (variable):** This is a transport-level segment (transport mode) or IP packet (tunnel mode) that is protected by encryption.



- **Padding (0–255 bytes):** The purpose of this field is discussed later.
- **Pad Length (8 bits):** Indicates the number of pad bytes immediately preceding this field.
- **Next Header (8 bits):** Identifies the type of data contained in the payload data field by identifying the first header in that payload (for example, an extension header in IPv6, or an upper-layer protocol such as TCP).
- **Authentication Data (variable):** A variable-length field (must be an integral number of 32-bit words) that contains the Integrity Check Value computed over the ESP packet minus the Authentication Data field.

Encryption Algorithms

- three-key triple DES
- RC5
- IDEA
- ~~Triple~~ three-key triple IDEA
- CAST
- Blowfish

Authentication Algorithms

- HMAC-MD5-96
- HMAC-SHA-1-96

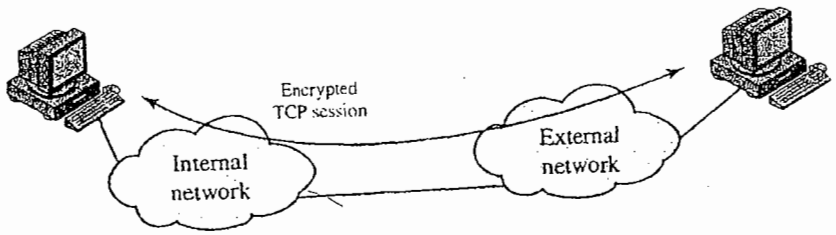
Padding

* Padding field serves several purposes

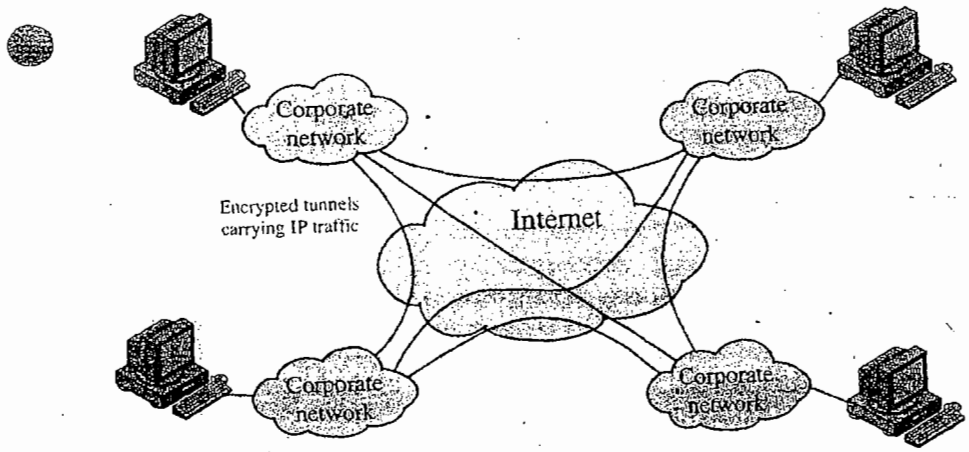
1. Used to expand the plaintext to required length
2. ESP format requires that pad length & next header fields be right aligned within a 32 bit word. And, ciphertext must be an integer ~~of~~ multiple of 32 bits. Padding field is used to assure this alignment.
3. Additional padding may be added to provide partial traffic flow confidentiality by concealing the actual length of payload.

Transport and Tunnel mode ESP.

* Fig shows two ways in which IPSec ESP service can be used.



(a) Transport-level security

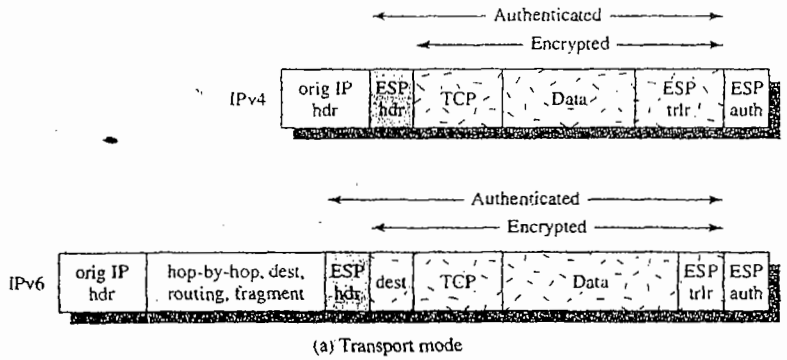


(b) A virtual private network via tunnel mode

Fig. 6.8 Transport Mode vs Tunnel Mode Encryption

Transport mode ESP.

* Encryption is provided directly b/w two hosts.
* Used to encrypt & optionally authenticate the data carried by (IP. eg TCP segment) as shown in below fig (a).



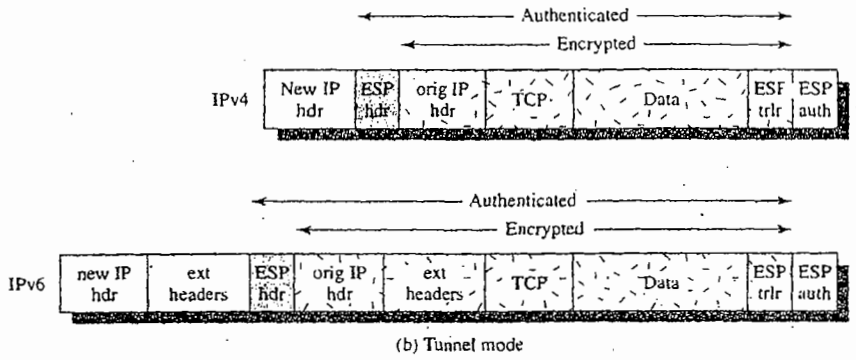
(a) Transport mode

* Transport mode operation

1. At the source, the block of data consisting of ESP trailer plus the entire transport layer segment is encrypted and the plaintext of this block is replaced with its cipher text to form the IP packet for transmission. Authentication is added if this option is selected.
2. The packet is then routed to dest^n. Each intermediat router needs to examine and process the IP header plus any plaintext IP extension headers but does not need to examine the cipher text.
3. The dest^n node examines and processes the IP header plus any plain text IP extension headers. Then, on the basis of the SPI in the ESP header the dest^n node decrypts the remainder of the packet to recover the plaintext transport layer segment.

Tunnel mode ESP

* used to set up VPN.
* used to encrypt an entire IP packet as shown in fig (b)



(b) Tunnel mode

* Here, the ESP header is prefixed to the packet and then the packet plus the ESP trailer is encrypted.
* The following steps occur for transfer of a transport layer segment from external host to the internal host.

1. the source prepares an inner IP packet with a destination address of the target internal host. this packet is prefixed by an ESP header; then the packet and ESP trailer are encrypted and authentication data may be added. The resulting block is encapsulated with a new IP header whose destn addr. is the firewall; this forms the outer IP packet.

2. The outer packet is routed to the destination firewall. Each intermediate router needs to examine and process the outer IP header plus any outer IP extension headers but does not need to examine the ciphertext.

3. The destination firewall examines and processes the outer IP header plus any outer IP extension headers. Then, on the basis of the SPI in the ESP header, the destn node decrypts the remainder of the packet to recover the plaintext inner IP packet. This packet is then transmitted in the internal network.

4. The inner packet is routed through zero or more routers in the internal network to the destination host.

COMBINING SECURITY ASSOCIATIONS

* the term security Association bundle refers to a sequence of SAs thru' which traffic must be processed to provide a desired set of IPSec services.

* Security associations may be combined into bundles in two ways

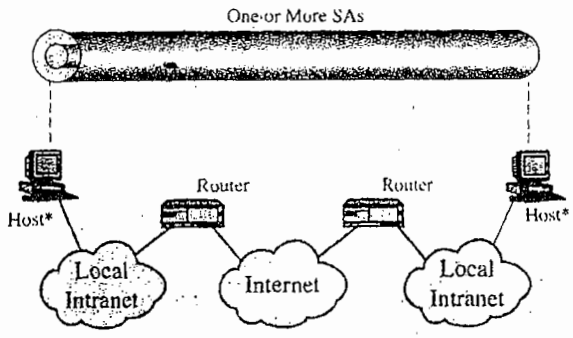
i.) Transport adjascency : Refers to applying more than one security protocol to the same IP packet, without invoking tunneling. This approach to combining AH & ESP allows for only one level of combination; further nesting yields no added benefits since the processing is performed at one IPSec instance; the destination.

ii) Iterated tunneling : Refers to the application of multiple layers of security protocols effected through IP tunneling. This approach allows for multiple levels of nesting, since each tunnel can originate or terminate at a different IPSec site along the path.

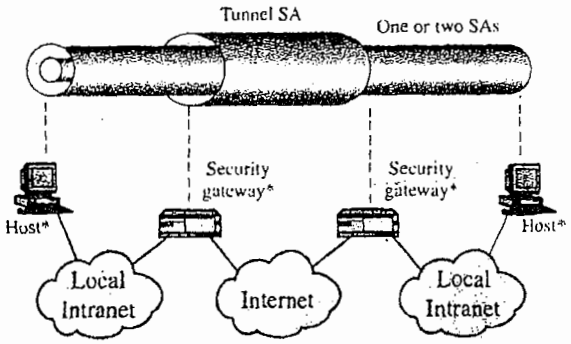
* These two approaches can also be combined.

Basic Combinations of SA

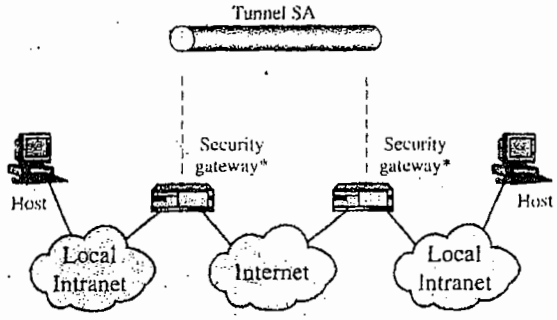
* Fig below shows four basic combinations of security Associations:



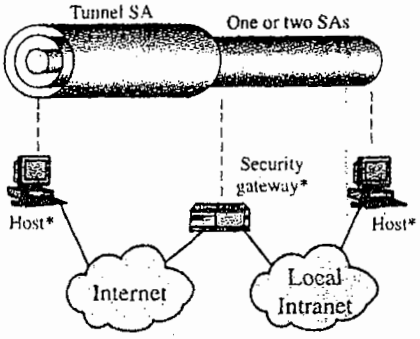
(a) Case 1



(c) Case 3



(b) Case 2



(d) Case 4

* = implements IPSec

Figure 6.10 Nested Tunnels

case (1):

* Here all security is provided b/w end systems that implement IPSec.
* Among the possible combinations:
  a. AH in transport mode
  b. ESP in transport mode
  c. ESP followed by AH in transport mode (an ESP SA inside an AH SA)
  d. Any one of a, b, or c inside an AH or ESP in tunnel mode.

case (2):

* Here, the security is provided only b/w gateways (routers, firewalls, etc) and no hosts implement IPSec.
* this case implement simple VPN support.
* only single tunnel SA is needed in this case.

(tunnel could support AH, ESP, or ESP with authentication option)

case (3):

* It builds on case 2 by adding end-to-end security.
* The same combinations discussed for cases 1 & 2 are allowed here.

case (4):

* ~~Support~~ provides support for a remote host that uses the internet to reach an organizations' firewall and then to gain access to some server or workstation behind the firewall.
* Only tunnel mode is required b/w the remote host

KEY MANAGEMENT

* The key management portion of IPSec involves the determination and distribution of secret keys.
* IPsec Architecture supports for two types of key management:
  1. Manual: A system administrator manually configures each system with its own keys and with the keys of other communicating systems.
  2. Automated: An automated system enables the on-demand creation of keys for SAs and facilitates the use of keys in a large distributed system with an evolving configuration.
* Automated key management protocol includes:
  1. oakley key Determination protocol
  2. Internet Security Association and key mgmt protocol (ISAKMP)

Oakley Key Determination Protocol

* Oakley is a key exchange protocol based on the Diffie-Hellman algorithm but providing added security
* Features of Oakley :
  - → Employs a mechanism known as cookies to thwart clogging attacks.
  - enables the two parties to negotiate a group
  - → It uses nonces to ensure against replay attacks.
  - → Enables the exchange of D-H public key values.
  - → Authenticates the D-H exchange to thwart man-in-the-middle attacks.
* The cookie exchange requires that each side send a pseudo random number, the cookie, in the initial message, which the other side acknowledges.
* oakley supports the use of different groups for the D-H key exchange. Each group includes the definition of the two global parameters and identity of the algorithm.
* Oakley ~~uses~~ employs nonces to ~~exchange~~ ensure against replay attacks. Each nonce is a locally-generated pseudo random number.

* three different Authentication methods can be used with Oakley:

1. Digital signature: the exchange is authenticated by signing a mutually obtainable hash; each party encrypts the hash with its private key. The hash is generated over important parameters, such as user IDs and nonces.

2. Public key encryption: the exchange is authenticated by encrypting parameters such as IDs and nonces with the sender's private key.

3. Symmetric-key Encryption: A key derived by some out-of-band mechanism can be used to authenticate the exchange by symmetric encryption of exchange parameters.

ISAKMP

* ISAKMP provides a framework for Internet key management and provides the specific protocol support, including formats, for negotiation of security attributes.

* ISAKMP defines procedures and packet formats to establish, negotiate, modify, and delete SA.

ISAKMP header format:

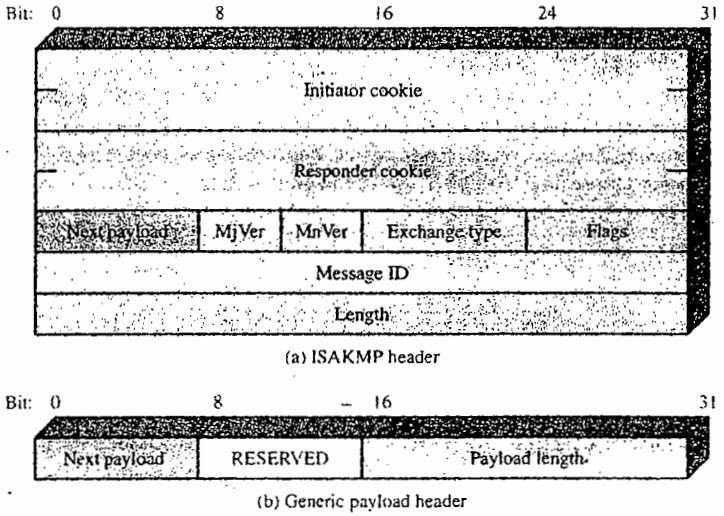


Figure 6.12 ISAKMP Formats

Figure 6.12a shows the header format for an ISAKMP message. It consists of the following fields:

- **Initiator Cookie (64 bits):** Cookie of entity that initiated SA establishment, SA notification, or SA deletion.
- **Responder Cookie (64 bits):** Cookie of responding entity; null in first message from initiator.
- **Next Payload (8 bits):** Indicates the type of the first payload in the message; payloads are discussed in the next subsection.
- **Major Version (4 bits):** Indicates major version of ISAKMP in use.
- **Minor Version (4 bits):** Indicates minor version in use.
- **Exchange Type (8 bits):** Indicates the type of exchange; these are discussed later in this section.
- **Flags (8 bits):** Indicates specific options set for this ISAKMP exchange. Two bits so far defined: The Encryption bit is set if all payloads following the header are encrypted using the encryption algorithm for this SA. The Commit bit is used to ensure that encrypted material is not received prior to completion of SA establishment.
- **Message ID (32 bits):** Unique ID for this message.
- **Length (32 bits):** Length of total message (header plus all payloads) in octets.

* ISAKMP defines payloads for exchanging key generation and authentication data.

ISAKMP payload types

* Fig (b) in prev page shows generic payload header
* Below fig shows ISAKMP payload types

Type	Parameters	Description
Security Association (SA)	Domain of Interpretation, Situation	Used to negotiate security attributes and indicate the DOI and Situation under which negotiation is taking place.
Proposal (P)	Proposal #, Protocol-ID, SPI Size, # of Transforms, SPI	Used during SA negotiation; indicates protocol to be used and number of transforms.
Transform (T)	Transform #, Transform-ID, SA Attributes	Used during SA negotiation; indicates transform and related SA attributes.
Key Exchange (KE)	Key Exchange Data	Supports a variety of key exchange techniques.
Identification (ID)	ID Type, ID Data	Used to exchange identification information.
Certificate (CERT)	Cert Encoding, Certificate Data	Used to transport certificates and other certificate-related information.
Certificate Request (CR)	# Cert Types, Certificate Types, # Cert Auths, Certificate Authorities	Used to request certificates; indicates the types of certificates requested and the acceptable certificate authorities.
Hash (HASH)	Hash Data	Contains data generated by a hash function.
Signature (SIG)	Signature Data	Contains data generated by a digital signature function.
Nonce (NONCE)	Nonce Data	Contains a nonce.
Notification (N)	DOI, Protocol-ID, SPI Size, Notify Message Type, SPI, Notification Data	Used to transmit notification data, such as an error condition.
Delete (D)	DOI, Protocol-ID, SPI Size, # of SPIs, SPI (one or more)	Indicates an SA that is no longer valid.

ISAKMP Exchanges.

+ ISAKMP provides a framework for message exchange, with the payload types serving as building blocks.

Exchange	Note
(a) Base Exchange	
(1) I → R: SA; NONCE	Begin ISAKMP-SA negotiation
(2) R → I: SA; NONCE	Basic SA agreed upon
(3) I → R: KE; ID_I; AUTH	Key generated; Initiator identity verified by responder
(4) R → I: KE; ID_R; AUTH	Responder identity verified by initiator; Key generated; SA established
(b) Identity Protection Exchange	
(1) I → R: SA	Begin ISAKMP-SA negotiation
(2) R → I: SA	Basic SA agreed upon
(3) I → R: KE; NONCE	Key generated
(4) R → I: KE; NONCE	Key generated
(5)*I → R: ID_I; AUTH	Initiator identity verified by responder
(6)*R → I: ID_R; AUTH	Responder identity verified by initiator; SA established
(c) Authentication Only Exchange	
(1) I → R: SA; NONCE	Begin ISAKMP-SA negotiation
(2) R → I: SA; NONCE; ID_R; AUTH	Basic SA agreed upon; Responder identity verified by initiator
(3) I → R: ID_I; AUTH	Initiator identity verified by responder; SA established
(d) Aggressive Exchange	
(1) I → R: SA; KE; NONCE; ID_I	Begin ISAKMP-SA negotiation and key exchange
(2) R → I: SA; KE; NONCE; ID_R; AUTH	Initiator identity verified by responder; Key generated; Basic SA agreed upon
(3)*I → R: AUTH	Responder identity verified by initiator; SA established
(e) Informational Exchange	
(1)*I → R: N/D	Error or status notification, or deletion

Notation:

I = initiator
R = responder
* = signifies payload encryption after the ISAKMP header
AUTH = authentication mechanism used