UNIT 3

Notes

Network Security

18CSE354T

NETWORK SECURITY

Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List Exploders
- Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing types
- Anomalies
- Object Format
- S/MIME

Reference :

Charlie Kaufman, Radia Perlman, Mike Speciner, Network Security, Prentice Hall of India, 2002.

Chapter 20 : Electronic Mail Security , Chapter 22 : PGP

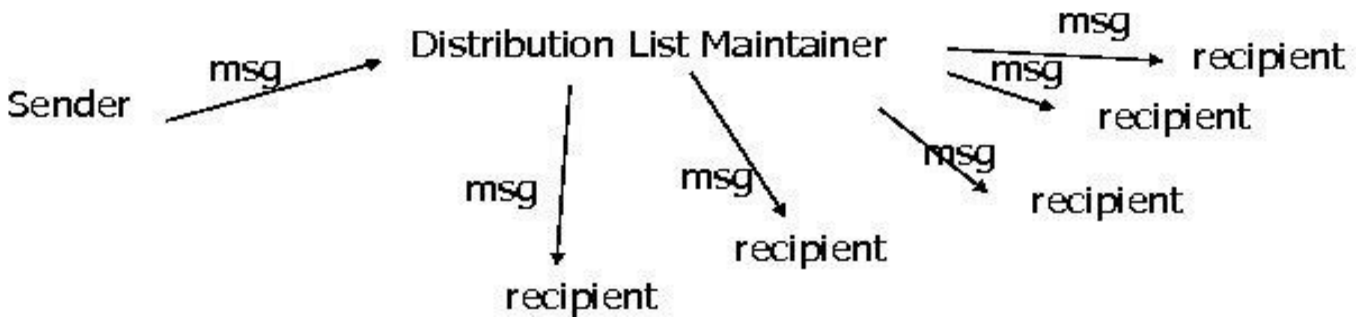
ELECTRONIC MAIL SECURITY

Basic Concepts : DISTRIBUTION LISTS

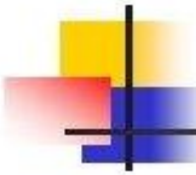
REMOTE EXPLODER

Email: Distribution List

- Simplest:
  - Single recipient per email message.
- Distribution List
  - Send mail to a **set** of recipients.
  - Remote Exploder Model



Basic Concepts : DISTRIBUTION LISTS REMOTE EXPLODER



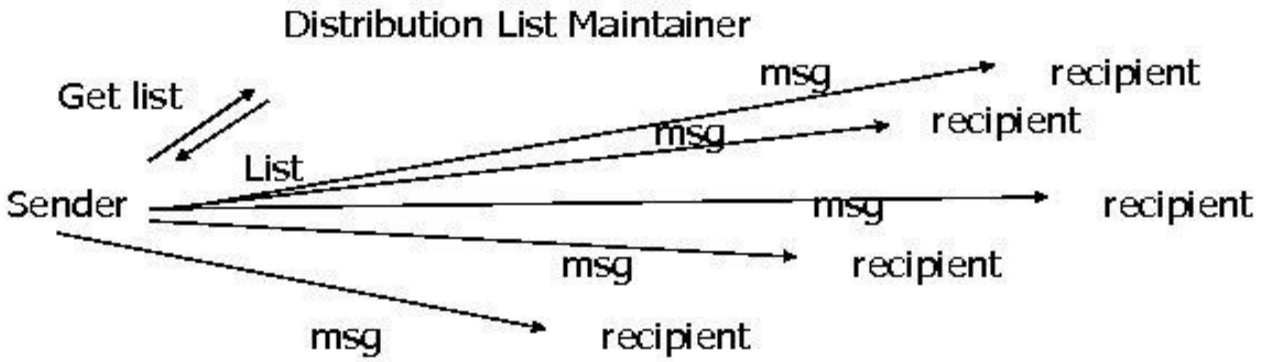
Email: Distribution List

- Remote Exploder
  - Allows the membership to be kept secret from sender.
  - Can be cheaper if recipients are geographically clustered around the list maintaining site.
  - More efficient if list size is bigger than message size.
  - Faster when distribution lists are contained in distribution lists.

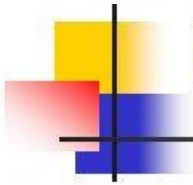
Basic Concepts : DISTRIBUTION LISTS
LOCAL EXPLODER

Email: Distribution List

- Distribution List
  - Send mail to a **set** of recipients.
  - Remote Exploder Model
  - Local Exploder Model



Basic Concepts : DISTRIBUTION LISTS
Remote Exploder Vs Local Exploder



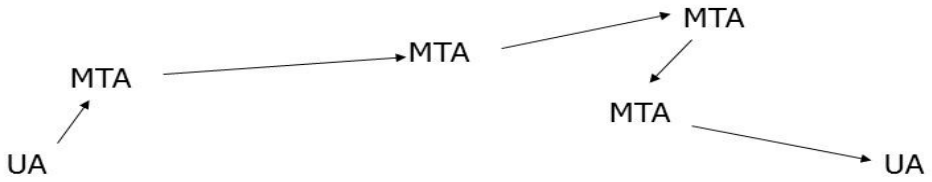
Privacy

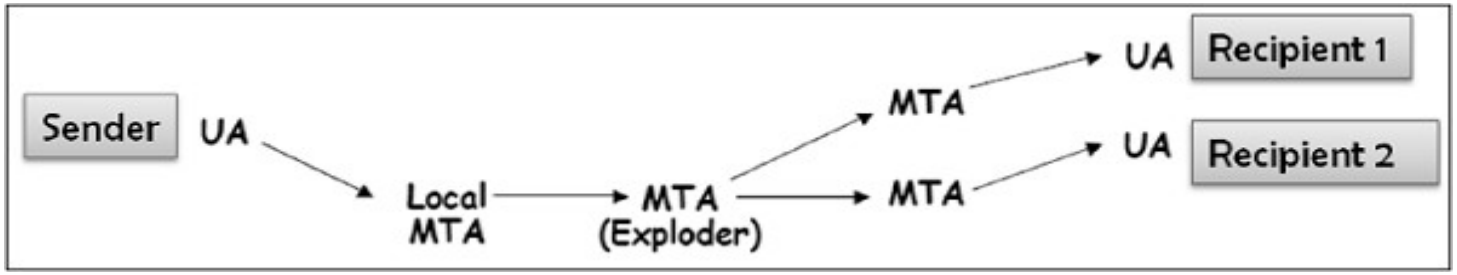
- With Distribution List Exploders
  - Remote exploding:
    - Alice chooses a secret key S and encodes her message.
    - Alice attaches S encrypted to all recipients.
    - Distribution list exploder decodes S and attaches it encrypted to all recipients.
  - Local exploding:
    - Alice needs to exchange keys with all people on the list.

Basic Concepts : STORE AND FORWARD(Mail Forwarding)

Mail Infrastructure

- Mail infrastructure consists of a mesh of mail forwarders.
  - Called Message Transfer Agents (MTA)
  - Processing at source and destination done by User Agent (UA)





SECURITY SERVICES FOR ELECTRONIC MAIL

- privacy
- authentication
- integrity
- non-repudiation
- proof of submission
- proof of delivery
- message flow confidentiality (did Alice sent Bob a message?)
- anonymity
- containment (leakage)
- audit

SECURITY SERVICES FOR ELECTRONIC MAIL(contd)

- accounting
- self destruct
- message sequence integrity

SECURITY SERVICES FOR ELECTRONIC MAIL(contd)

- accounting
- self destruct
- message sequence integrity

Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List
- Exploders Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing
- types Anomalies
- Object Format
- S/MIME

Establishing keys

- Establishing Public Keys
- Establishing Secret Keys

Establishing keys- (Establishing Public Keys)

Ex - Alice wants to send an encrypted message to Bob

- Receive public key through some secure out-of-band mechanism , and install it on workstation .
- Obtain through a PKI (e.g, looking it up in a directory)
  (Public Key Infrastructure)
- The email system could allow piggybacking of certificates (and perhaps CRLs) on email messages.

Establishing keys- (Establishing Secret Keys)

Ex – How can Alice and Bob establish a shared secret key for email ?

- Private communication
- Alice to obtain a ticket for Bob from a KDC Mediated Authentication (with KDC) , and include that ticket with her first message to Bob .

Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List
- Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing
- types Anomalies
- Object Format
- S/MIME

PRIVACY

Breach to Email Privacy

- Eavesdropper listening to the message during transmission
  on the wire if the network doesn't provide link encryption.
- Relay nodes (routers or mail forwarders) having software to store messages and divulge them to people other than the intended recipients.
- Conflicting Security Needs

PRIVACY

- End – to – End Privacy
- Privacy with Distribution List Exploders

- multiple recipients ➠ repeated encryption of long message
- ➠ only encrypt session key for each recipient
- list exploder: get session key, re-encrypt for each recipient
- local list: need key for each recipient

PRIVACY

End – to – End Privacy

- *Scenario : Alice might want to send a message to Bob in such a way that only Bob can read it .*
- *Extend the scenario (Alice to multiple recipients…)*

Issues

- If Long message to send multiple recipients , the long message would have to be encrypted once for each recipient , producing a different version to be send to each recipient.
- Public key Encryption less efficient than secret key encryption.

PRIVACY

End – to – End Privacy

Solution

- Alice chooses a random secret key S to be used only for encrypting that one message.
  - Encrypt the message with S
  - Encrypts S with Bob's Key
- Transmit both quantities to Bob .
- If Multiple recipients …. Still only encrypts the message once , with key S.
- Encrypts S once for each recipient , with appropriate key , and includes each encrypted S with the encrypted message.

PRIVACY

End – to – End

Privacy(contd)

Ex – Alice sending message to Bob , Carol and Ted.

The mail message Alice will send includes :

- **Bob's name; $K_{Bob}\{S\}$**
- **Carol's name; $K_{Carol}\{S\}$**
- **Ted's name; $K_{Ted}\{S\}$**
- **$S\{m\}$**

Privacy with Distribution List Exploders

Scenario : Alice sending a message to distribution list which will be remotely exploded and Bob is one of the recipients.

- Alice will choose a random per-message secret key S, and encrypt the message with S.
- The distribution list exploder will decrypt S(but it does not need to decrypt the message!), and re-encrypt S with the key for each recipient to whom it is forwarding the message.

Local exploding requires different mechanisms.

- Alice has to trust the maintainer of the mailing list , since a bad guy could insert extra names into the distribution list .
- Alice will not be able to send secure message ro a name without establishing a key for that individual.

Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List Exploders
- Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing
- types Anomalies
- Object Format
- S/MIME

Authentication of the source

- Source Authentication Based on Public Key Technology
- Source Authentication Based on Secret Keys
- Source Authentication with Distribution Lists

Authentication of the source

Source Authentication Based on Public Key Technology

Alice -> Bob Exchange

- Alice can digitally sign the message , using her private key , which will assure Bob that Alice wrote the message.

How Message Signing Done ?

- *Compute hash of the message (using MD5) , & then to sign the message digest.*

- This method extends easily to multiple recipients & distribution lists .

Authentication of the source

Source Authentication Based on Public Key Technology(contd)

Scenario : If Alice doesn't know that Bob Knows her Public Key

- Send message & let Bob fetch the public key (together with finding a chain of certificates to certify Alice's key)

OR

- Include the public key in the header of the message , together with a chain of certificates.

Authentication of the source

Source Authentication Based on Secret Keys

Case 1: If Alice has shared key with Bob , then she can reassure Bob that she is Alice by proving she knows the shared secret key . How this is done ?

- Using **MIC (Message Integrity Code) & MAC (Message Authentication Code)**

Case 2: Multiple recipients …… Alice shares to Bob & with Ted

- **Message Digest Computed once , but the MAC on the message digest independently computed for each recipient .**

Authentication of the source

Source Authentication with Distribution Lists

- When using mail exploders and secret key technology , it is vital that the mail exploder verify the source before forwarding the message , and include the name of the sender in the body of the message that the mail exploder cryptographically protects.

Message Integrity

- authentication always with message integrity
- integrity without authentication: ransom note ➠ no system exists

Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List Exploders
- Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing
- types Anomalies
- Object Format
- S/MIME

Non-repudiation

Non-Repudiation

- Alice cannot deny having sent message to Bob
- may want plausible deniability

public key: non-repudiable source authentication easy

secret key: repudiable source authentication easy

Plausible Deniability Based on Public Key Technology

Plausible Deniability with Public Keys

- Bob knows message m from Alice
- Bob can't prove it to anyone else

1. Alice: picks secret S just for m
2. $\{S\}_{\text{Bob}}$
3. $[\{S\}_{\text{Bob}}]_{\text{Alice}}$
4. use S to compute MIC of m: DES CBC residue
5. Alice $\rightarrow$ Bob: MIC(S), $[\{S\}_{\text{Bob}}]_{\text{Alice}}$, m (separately ...)

➠ Bob knows that message was from Alice (MIC)
Bob can construct any message he likes

Non-Repudiation with Secret Keys

Non-Repudiation with Secret Keys

- Bob prove to judge that Alice sent message
- need notary N with secret S_N, trusted by Bob, judge
- N authenticates Alice
- N: MIC with S_N ➠ *seal* MD("Alice", m or MD, S_N)
- sent m, seal to Bob
- Bob verify message: share key with N or ask N
- judge asks N if seal is valid

Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List Exploders
- Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing
- types Anomalies
- Object Format
- S/MIME

Pretty Good Privacy(PGP)-Overview

- Email security protocol invented by Phil Zimmermann.

- PGP was designed to provide all four aspects of security, i.e., privacy, integrity, authentication, and non-repudiation in the sending of email.

- **PGP uses a digital signature** (a combination of hashing and public key encryption) to *provide integrity, authentication, and non-repudiation.*

- PGP **uses a combination of secret key encryption and public key encryption** to provide *privacy*. Therefore, we can say that the digital signature uses one hash function, one secret key, and two private-public key pairs.

Pretty Good Privacy- Overview (contd)

- PGP is an open source and freely available software package for email security.
- PGP provides authentication through the use of Digital Signature.
- It provides confidentiality through the use of symmetric block encryption.
- It provides compression by using the ZIP algorithm, and EMAIL compatibility using the radix-64 encoding scheme.

Algorithms Supported by PGP

For asymmetric key encryption

- RSA
- DSS
- Diffie-Hellman Key

For Symmetric Key Encryption

- CAST-128
- IDEA
- DES-3

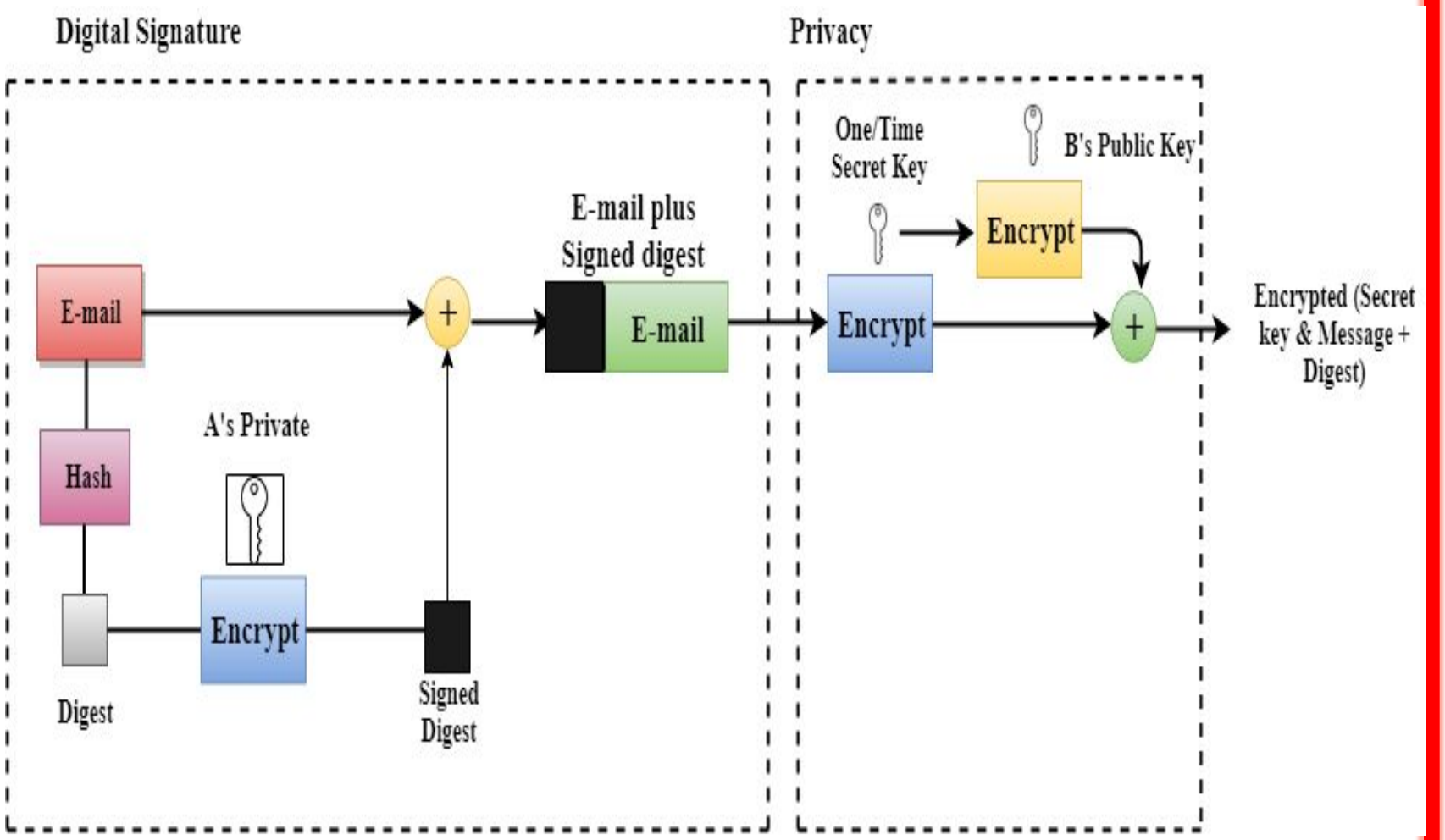
For Authentication

- SHA-1

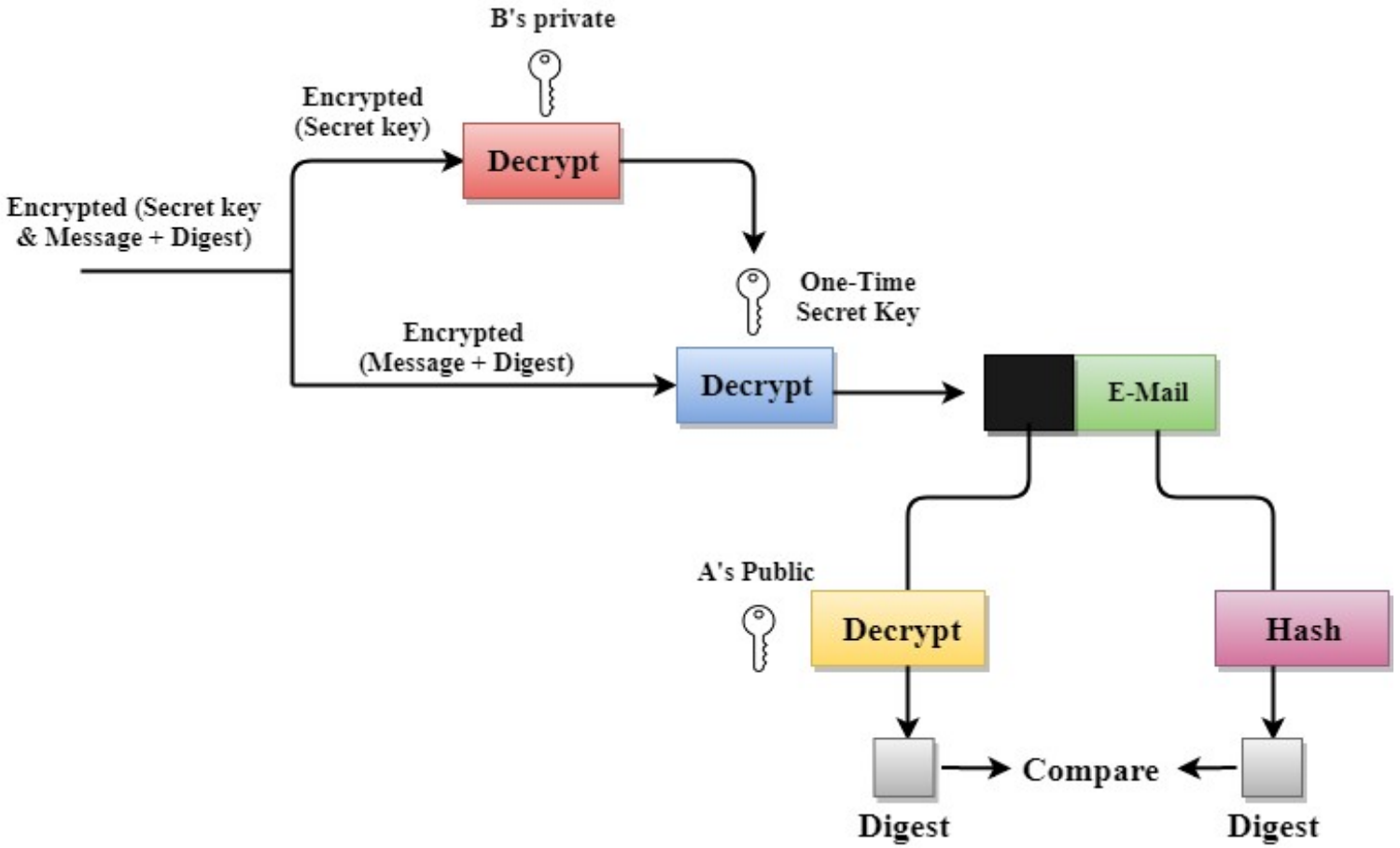
Steps taken by PGP to create secure e- mail at the sender site

- The e-mail message is hashed by using a hashing function to create a digest.
- The digest is then encrypted to form a signed digest by using the sender's private key, and then signed digest is added to the original email message.
- The original message and signed digest are encrypted by using a one-time secret key created by the sender.
- The secret key is encrypted by using a receiver's public key.
- Both the encrypted secret key and the encrypted combination of message and digest are sent together.

PGP at the Sender site (A)



PGP at the Receiver site (B)



Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List Exploders
- Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing
- types Anomalies
- Object Format
- S/MIME

Efficient Encoding

- PGP allows you to specify , when handing a file to PGP to process , whether the file is text or binary.
- If you tell PGP the file is binary , then PGP will not canonicalize it , and PGP will mark the PGP-processed message as binary so that the PGP at the receiver will know not to perform reverse canonicalization.
- By merely marking the file this way , PGP manages to avoid ever needing two encodings on a single message.

Efficient Encoding

- To conserve bits PGP compresses a file before sending it , whether it is binary or not.
- PGP uses the utlity ZIP for compression.
- 50% compression achieved ; PGP need to do one expansion encoding after encrypting a file , what PGP actually has to transmit will often be smaller than the original file.

Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List Exploders
- Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing types
- Anomalies
- Object Format
- S/MIME

Certificate and Key Revocation

- Certificate can be revoked by whoever signed the certificate.
- Current custom in PGP is to issue non – expiring certificates (by omitting the *VALIDITY PERIOD* field).

Certificate and Key Revocation

- Key can be revoked too. Revoked only by the owner .
- Owner will issue a key revocation only if felt someone has discovered owners private key / stolen owner's private key .

Certificate and Key Revocation

- Key and certificate revocations are distributed informally, just as are public keys and certificates.
- There are public servers where you can post and search for certificates.

Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List Exploders
- Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing types
- Anomalies
- Object Format
- S/MIME

Signature Types

- For each signed quantity , PGP indicates whether what is being signed is a message or a certificate .
- It's a good practice whenever a key can be used for multiple things to explicitly say what kind of thing is being processed . This avoids any possibility of aliasing .

YOUR PRIVATE KEY

- To verify signatures on signed messages , you don't need a private key .
- To sign your own messages , or if you want to receive encrypted PGP mail , you need a private key.

- PGP will generate private key for you and you can specify the size of the key .

PGP – Key Rings

- A key ring is a data structure you keep that contains some public keys , some information about people , and some certificates .
- PGP users Sharing key ring information provides a quick method of building up your database of public keys.

PGP – Key Rings

- PGP allows you to assert how much trust you place on different people.
- There are 3 levels of trust : **None , Partial or Incomplete .**
- PGP computes the trust that should be placed on certificates and public keys in your key ring based on the trust information you asserted on the people.
- A certificate signed by someone you indicated you don't trust at all will be ignored.
- Whom or what you trust is a very private decision

Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List Exploders
- Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing types
- Anomalies
- Object Format
- S/MIME

Anomalies

Anomalies associated with

- **File Name**
- **People Names**

Anomalies

- PGP recommends that the user identification string contains a little more information than just the name of the user , for instance , the user's email address .
- PGP recommends including Internet email addresses in the name.

Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List Exploders
- Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing types
- Anomalies
- Object Format
- S/MIME

Object Formats

- PGP reads and writes a variety of objects .

- Objects may contain a variety of fields , and some objects contain other objects.

Object Formats

Each object can be represented in two formats .

- **Compact**
- **SMTP mailable**

Object Formats

The SMTP mailable format is computed from the compact format by performing *encoding and adding a human-readable header and trailer* so that a human will know the cyber crud between the header and trailer needs to be processed by PGP.

Object Formats

The human – readable header is

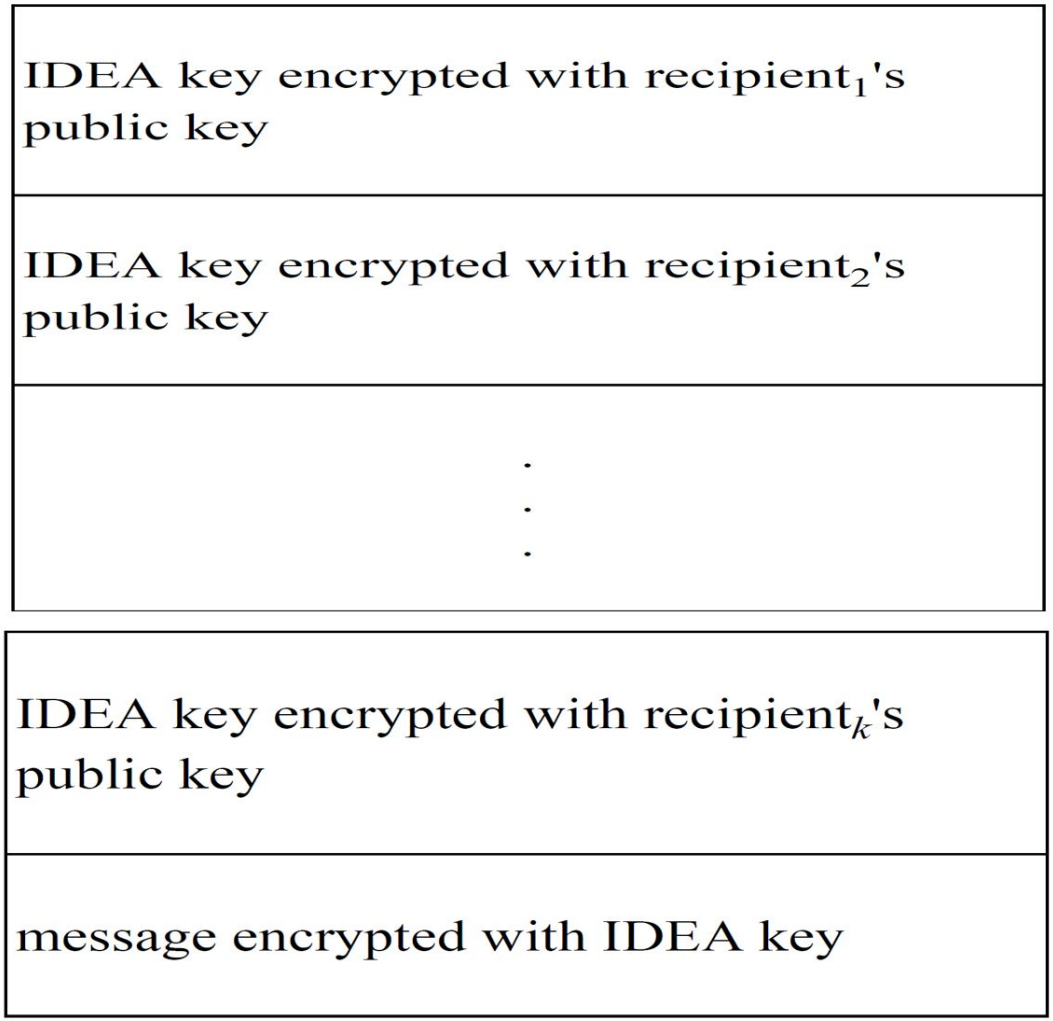
-----BEGIN PGP MESSAGE

Version 2.2

The Human – readable trailer is

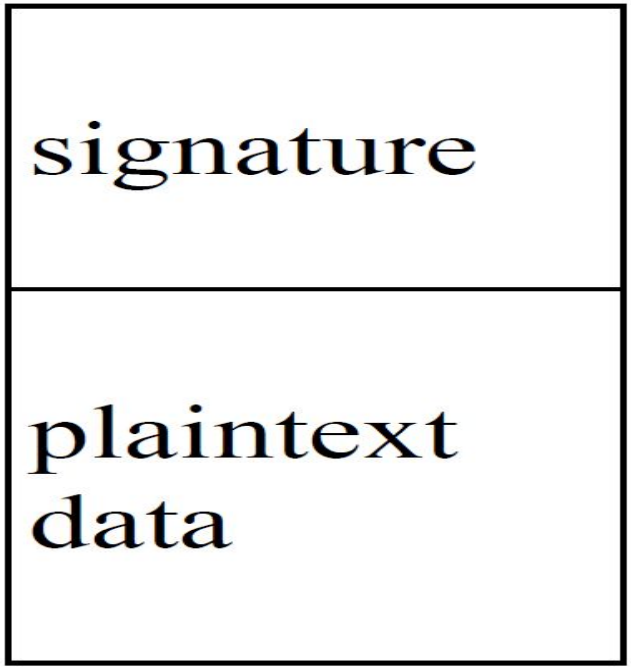
-----END PGP MESSAGE -----

Message Formats – Encrypted Message

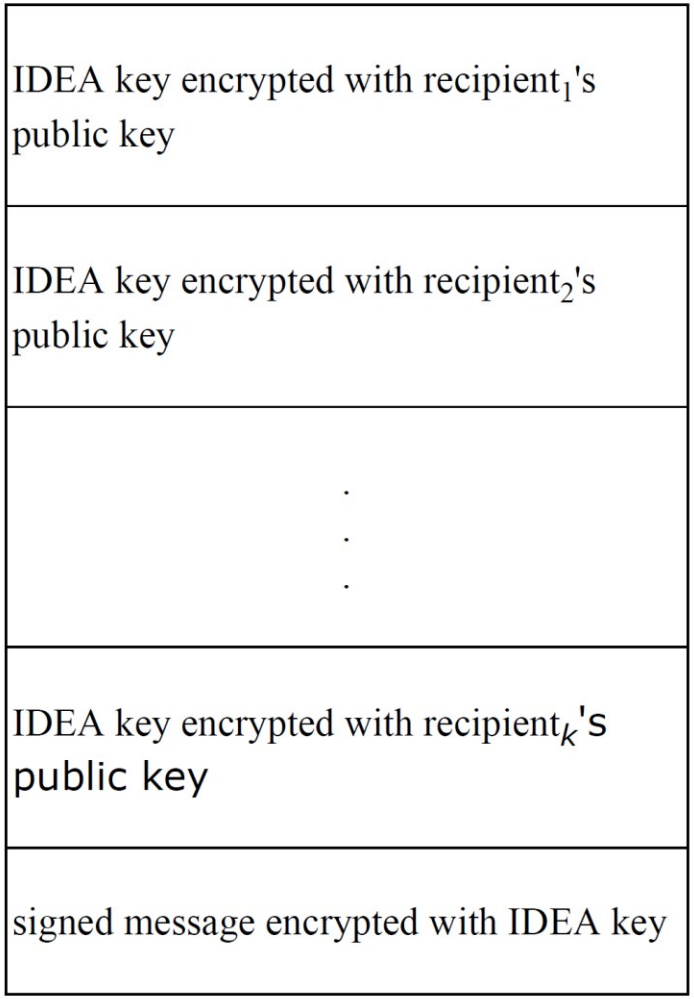


Message Formats – Signed Message

- Signed Message :This is likely to be COMPRESSED DATA . When uncompressed , the result is



Message Formats – Encrypted Signed Message

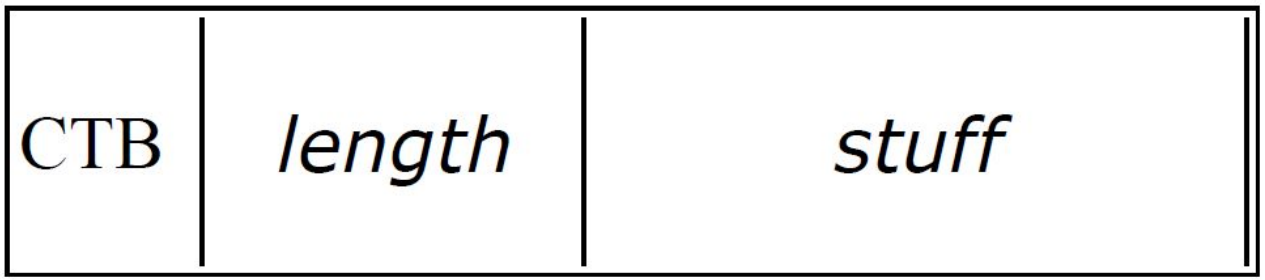


Message Formats – Signed Human- Readable Message

50 -----BEGIN PGP SIGNED MESSAGE-----
cleartext message
50 -----BEGIN PGP SIGNATURE----- Version: 2.2
signature

Primtive Object Formats

Primitive object formats are



- **CTB** stands for **cipher type byte** and specifies the PGP object type and the length of *length.*

- Certain object types contain one or more integers in their *stuff* field.
- Integers are encoded as a length in bits (two octets, most significant first) followed by the binary integer with leading 0 bits to pad to a multiple of 8 bits then packed into octets most significant first

Unit -3 Topics

- Security Services for E-mail
- Establishing keys
- Establishing Public and secret keys
- Privacy
- End-to end Privacy, Privacy with distribution List Exploders
- Authentication of the source
- Based on public key technology and secret keys and with distribution list
- Message Integrity
- Non-repudiation
- Introduction and Overviw of PGP
- Efficient Encoding
- Certificate and key revocation
- Signature types, Private key, Fing types
- Anomalies
- Object Format
- S/MIME

Secure/Multipurpose Internet Mail Extension (S/MIME)

- S/MIME is a security-enhanced version of Multipurpose Internet Mail Extension (MIME).
- In this, public key cryptography is used for digital sign, encrypt or decrypt the email.
- User acquires a public-private key pair with a trusted authority and then makes appropriate use of those keys with email applications.

Difference between PGP and S/MIME

S.NO	PGP	S/MIME
1.	It is designed for processing the plain texts	While it is designed to process email as well as many multimedia files.
2.	PGP is less costly as compared to S/MIME.	While S/MIME is comparatively expensive.
3.	PGP is good for personal as well as office use.	While it is good for industrial use.

Difference between PGP and S/MIME

4.	PGP is less efficient than S/MIME.	While it is more efficient than PGP.
5.	It depends on user key exchange.	Whereas it relies on a hierarchically valid certificate for key exchange.
6.	PGP is comparatively less convenient.	While it is more convenient than PGP due to the secure transformation of all the applications.
7.	PGP contains 4096 public keys.	While it contains only 1024 public keys.

Difference between PGP and S/MIME

8.	PGP is the standard for strong encryption.	While it is also the standard for strong encryption but has some drawbacks.
9.	PGP is also be used in VPNs.	While it is not used in VPNs, it is only used in email services.
10.	PGP uses **Diffie hellman digital signature**.	While it uses **Elgamal digital signature**.