

UNIT - V  
E-Mail, IP & Web-security (1)

E-mail security : - (2)

Security services for e-mail attacks possible through email - (2)

Establishing key & Privacy - (4)

Authentication of the source - (6)

Message Integrity - (8)

Non-repudiation - (9)

Pretty Good Privacy - (10)

S/MIME - (15)

IP Security: overview of IPsec - (17)

IP and IPv6, Authentication Header - (21)

Encapsulation Security Payload ESP - (24)

Internet Key Exchange (Phases of IKE,

ISAKMP / IKE encoding) - (27)

Web security: SSL/TLS Basic Protocol - (31)

Computing the keys - (35)

Client authentication - (36)

PKI as deployed by SSL - (36)

Attacks fixed in v3

possible through email - (2)

(3)

key & Privacy - (4)

Authentication of the source - (6)

Message Integrity - (8)

Non-repudiation - (9)

Pretty Good Privacy - (10)

S/MIME - (15)

IP Security: overview of IPsec - (17)

IP and IPv6, Authentication Header - (21)

Encapsulation Security Payload ESP - (24) (26)

Internet Key Exchange (Phases of IKE, ISAKMP / IKE encoding) - (29) (27)

Web Security: SSL/TLS Basic Protocol - (31)

Computing the keys - (35)

Client authentication - (36)

PKI as deployed by SSL - (36)

Attacks fixed in v3  
Exportability - (38) encoding - (39)  
SET - (40)

E-mail, IP and web-security

E-mail security : Security services for E-mail:-

\*. For security, some features are provided for the electronic mail systems, are as follows;

\*. Privacy

\*. Authentication

\*. Integrity

\*. Non-repudiation

\*. Proof of submission

\*. Proof of delivery

\*. Msg flow of confidentiality

\*. Anonymity

\*. Containment

\*. Audit

\*. Accounting

\*. Self-destruct

\*. Msg sequence integrity

## Attacks possible through E-mail :- ③

\* E-mail hacking is the illegal access to (or) manipulation of an e-mail account

\* SPAM

\* VIRUS

\* PHISHING

### SPAM:-

\* Spam is created by attackers who send bulk e-mail.

\* Spammers attempt to find new ways around the increased legislation and policies governing unsolicited emails.

### VIRUS:-

\* A virus incorporates email as a means of transportation. This type of virus is called a worm - the Sobig virus is an example.

\* This virus creates a spamming framework by taking over unwilling participants.

## PHISHING:-

(4)

\* It is a type of attack that involves e-mails that appear to be from legitimate businesses that the user associated with.

\* Phishing msgs look authentic, with all the corporate logos & formats as to that of official emails.

> An account no

> A Pwd (or)

> A date - of - birth.

## Establishing keys privacy:-

### Establishing keys:-

\* Security services are provided by cryptography requires keys.

### Establishing secret keys:-

\* Two Parties establish a shared secret key for e-mail in some other means of private comm.

\* Electronic mail is private but there are many ways to read that msg.

End-to-End Privacy:-

\* Alice want to send a msg to Bob that only Bob can read it. She can't depend on the n/w keeping the msg secret, but she can ensure that nobody but Bob can read the msg by using cryptography to encrypt the message.

Privacy with Distribution List Explodes:-

\* If Alice is sending a msg to a distribution list which will be remotely exploded, and Bob is only one of the recipients.

> Local exploding requires diff mechanisms.

> Alice has to trust the maintainer of the mailing list, since a user can insert extra names into the distribution list.

## Authentication of the Source (6)

\* In an unsecured mail system, it is possible for Carol to send a msg to Bob where the FROM field says Alice.

\* This cause great harm if Bob takes the msg seriously.

Source Authentication based on public-key technology:-

\* Bob knows Alice's public key. Alice digitally sign the msg, using her private key, which will assure Bob that Alice wrote the msg.

\* The Alice compute a hash of the msg & then to sign the msg digest, since computing a msg digest.

\* Since, computing a msg digest is faster than public key ops and the msg digest is usually a smaller quantity to sign than the msg.

## Source Authentication based on secret keys.

MIC (or) MAC: - [Message Integrity Code] 7

\* The MAC is the CBC residue of the msg computed with the shared secret key.

\* The MAC is the msg digest of the shared secret appended to the msg.

\* The MAC is the encrypted msg digest, where the 128-bit msg digest is encrypted with the shared secret key, for instance in ECB (or) CBC mode.

## Source Authentication with Distribution lists

> Source Authentication is easy with public keys and distribution lists.

> Source Authentication is complicated with secret keys.

> Using mail exchangers & secret-key technology.



## Message Integrity (8)

\* Bob receives a msg from Alice,  
how does bob know that Carol  
did not intercept the msg &  
modify it?

## Message Integrity without Source Authentication

\* To provide msg integrity  
protection without source authentication

is possible?

\* The msg anonymous so there  
is no source verification.

## Integrity Protection :-

\* To prevent someone from  
guessing that the first part of the msg  
was the proof that they were the  
kidnappers & substitute a diff second  
part of the msg, with diff direction  
for dropping off price money.

authentication by humans

this odd combination

such a small market that none is to be developed

Repudiation: Non-Repudiation

\* It is the act of denying that the particular person sent a msg.

Non-Repudiation: -

\* If Alice sends a msg to Bob, Alice can't deny that she sent msg, Alice can prove to a third party that she sent the msg.

\* Non-repudiation based on Public-key Technology.

\* Plausible Deniability Based on Public-key Technology.

\* Non-Repudiation with secret keys.

## PRETTY GOOD PRIVACY [PGP]:-

- \* PGP is a secure mail protocol (10)
- \* PGP performs encryption & integrity protection on files.

Process: - \* one send a secure mail msg could first transform the file to be mailed using PGP & then mail the transformed file using traditional mailer.

### Key distribution:-

- \* PGP uses public-key computation for personal keys.
- \* Public keys certifications and certificate chains verifications are done b/w PGP, PEM, S/MIME.
- \* PGP Assumes APachy.
- \* With PGP, each user decides which keys to trust.

### Efficient encoding:-

\* PEM

\* PGP.

