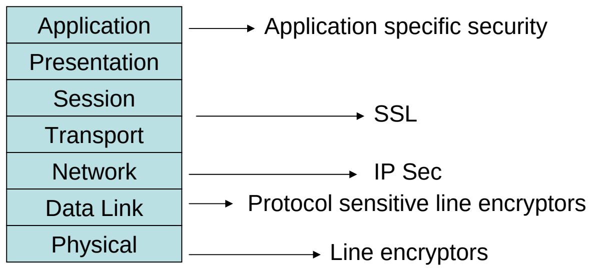
Communications Security in Real-Time

OSI security layer, Perfect Forward Secrecy, Denial of Service, Endpoint Identifier Hiding, negotiating Crypto Parameters

Security in OSI Reference Model



Application → Application specific security

Presentation

Session

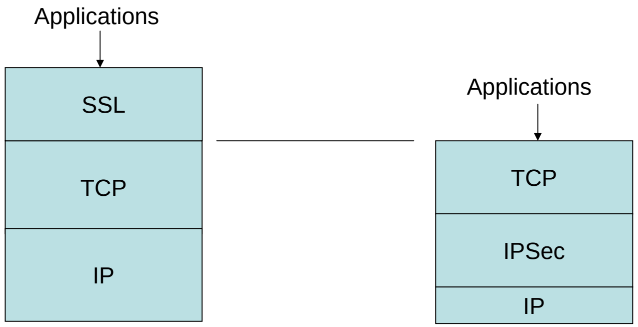
Transport → SSL

Network → IP Sec

Data Link → Protocol sensitive line encryptors

Physical → Line encryptors

IPSec and Secure Sockets Layer (SSL)



Applications → SSL / TCP / IP

Applications → TCP / IPSec / IP

Security above Layer 4 (SSL)

- Operating system (up to level 4) requires no modification
- Applications require modification to talk to SSL instead of TCP
- Inserted or modified packets (below layer 4) may cause connection to be broken
- May distinguish different users from same network address

Security at Layer 3 (IPSec)

- Applications require no modification in respect of connection to TCP but may require modification if full security functionality is to be achieved
- Can be implemented as outboard device if undesirable to modify operating system
- Suitable for address based firewall type security
- Can authenticate network address

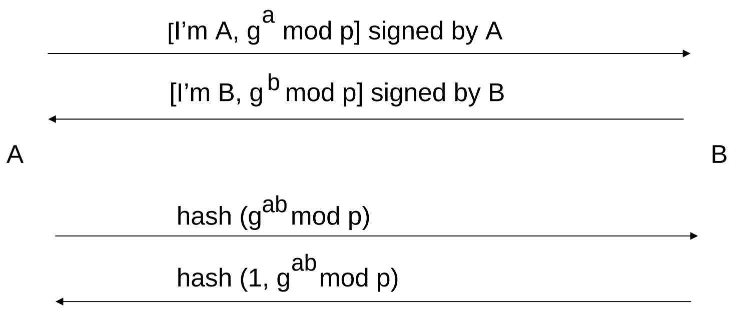
TCP Session Hijack

- Wait until initial security handshake is completed
- Impersonate source address and introduce high TCP sequence numbers
- Protect by establishing session key
- Prevent replay by changing session key whenever TCP sequence numbers repeat

Perfect Forward Secrecy (I)

- Attacker records all communications
- Breaks into one or both endpoints to obtain long-term secrets (master keys or private keys in public key system)
- Recorded communications then decrypted
- Protect by using systems which have PFS
- Also provides escrow foilage

Perfect Forward Secrecy (II)



[I’m A, g^a mod p] signed by A

[I’m B, g^b mod p] signed by B

A

B

hash (g^{ab} mod p)

hash (1, g^{ab} mod p)

After the session each party “forgets” g^{ab} mod p and a, b

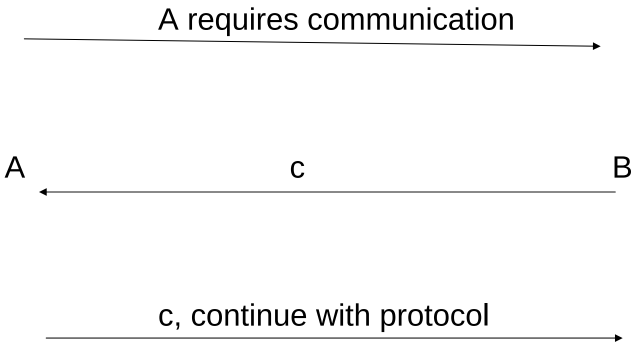
Perfect Forward Secrecy (III) Systems without PFS

- Normal public key encryption
- Kerberos (reliance on master key)
- Any system in which session keys are encrypted under public keys

Denial of Service/Clogging Protection (I)

- Attack with forged IP addresses in large number of packets
- Server resources exhausted in authentication attempts
- Protect by “cookies” or “puzzles”

Denial of Service/Clogging Protection (II)



A requires communication

A

c

B

c, continue with protocol

If A has forged a network address he will have difficulty returning c

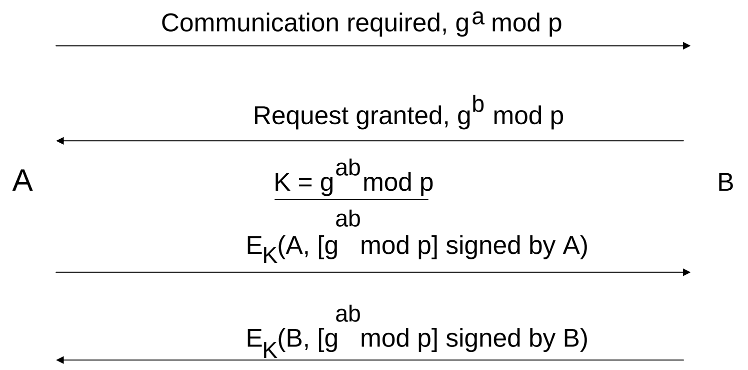
Denial of Service/Clogging Protection (III)

- “Puzzles” may be set by servers – long time to solve but quick to check (e.g. find message for a particular MD)
- Validity of “puzzle” depends upon reasonably uniform workstation power
- “Cookies” and “puzzles” should be “stateless” - e.g. a function of network address
- Very difficult to defend against distributed attack caused by viruses

Endpoint Identifier Hiding (I)

- Anonymous Diffie – Hellman exchange
- Divulge identities encrypted under new key from Diffie – Hellman
- Authenticate using original keys (master keys or private keys in public key system)

Endpoint Identifier Hiding (II)



Communication required, $g^a \bmod p$

Request granted, $g^b \bmod p$

A

$K = g^{ab} \bmod p$

B

E_K(A, [$g^{ab} \bmod p$] signed by A)

E_K(B, [$g^{ab} \bmod p$] signed by B)

Negotiating Crypto Parameters (I)

- A and B may wish to negotiate
  - algorithm for encryption
  - hash function
  - prime (p) in Diffie – Hellman exchange
- Allows migration from broken algorithms
- Allows migration to stronger system as workstation power increases
- Important to negotiate only after initial handshake when a shared secret is established