Securing Email with Pretty Good Privacy

CSC 249 April 17, 2018

Overview

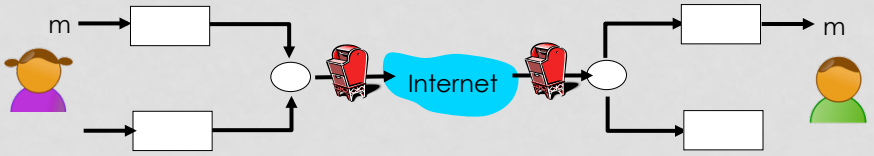
Applying security measure to the Internet

- Securing email – Pretty Good Privacy
- Secure sockets layer, SSL
- Firewalls and Intrusion Detection Systems

2

EMAIL: CONFIDENTIAL STEP 1

- Alice wants to send confidential e-mail, m, to Bob.

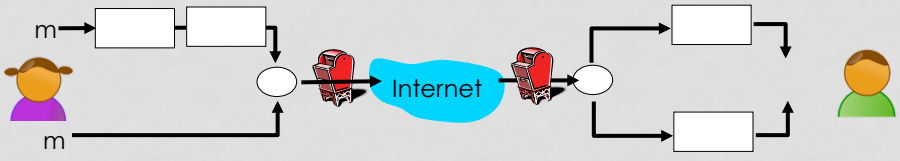


Alice:

1) Generate random *symmetric* private key, K_S
2)
3)
4)

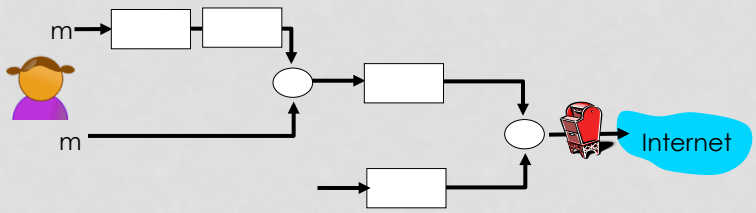
EMAIL: MESSAGE INTEGRITY & AUTHENTICATION

- Alice wants to provide sender authentication message integrity. *...How?*



Email: Fully Secure

- Alice wants to provide secrecy, sender authentication & message integrity. *...How?*



Pretty Good Privacy (PGP)

- To Activity (to act out PGP)…
- Internet e-mail encryption scheme, de-facto standard.
- Uses
  - Symmetric key cryptography
  - Public key cryptography
  - Hash function
  - Digital signature
- Provides
  - Secrecy
  - Sender authentication
  - Integrity

SSL: Secure Sockets Layer

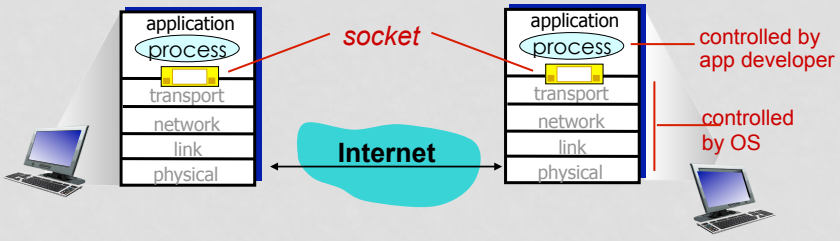
- Provides
  - Confidentiality
  - Integrity
  - Authentication
- Original goals:
  - Encryption (especially credit-card numbers)
  - Web-server authentication
  - Optional client authentication
  - Minimum effort doing business with new merchant
- Available to all TCP applications
  - Secure socket interface

Recall: Socket Programming

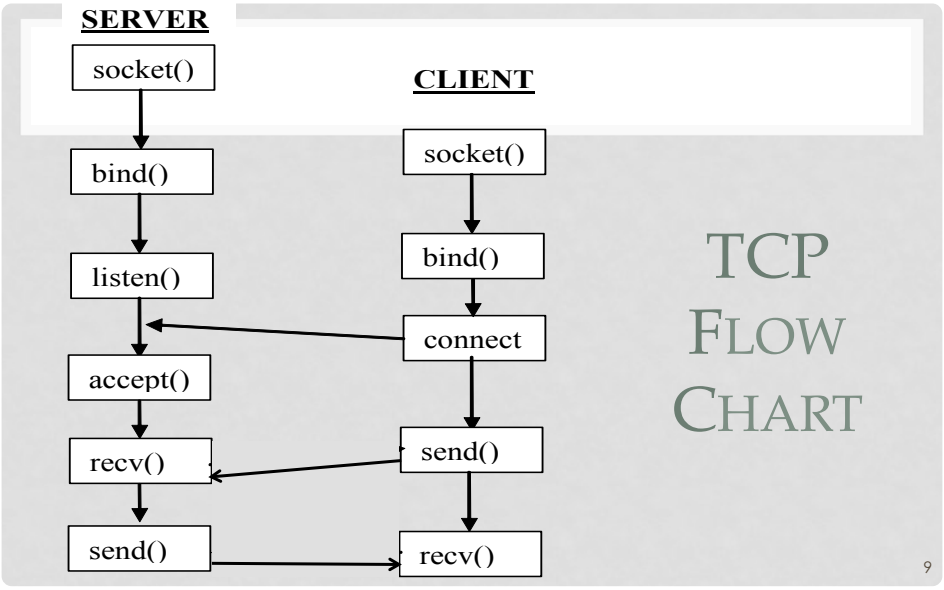
Application layer communication via the transport layer

goal: build client/server applications that communicate using sockets

socket: door between application process and transport protocol

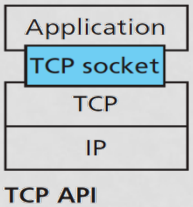


8

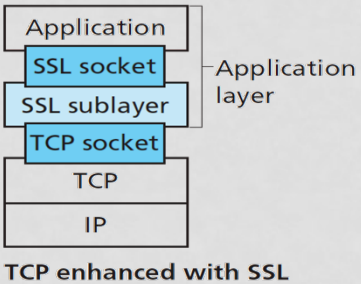


SSL: Secure Sockets Layer

- Provides transport layer security to any TCP-based application using SSL services.
- Security services:
  - Server authentication
  - Data encryption
  - Client authentication

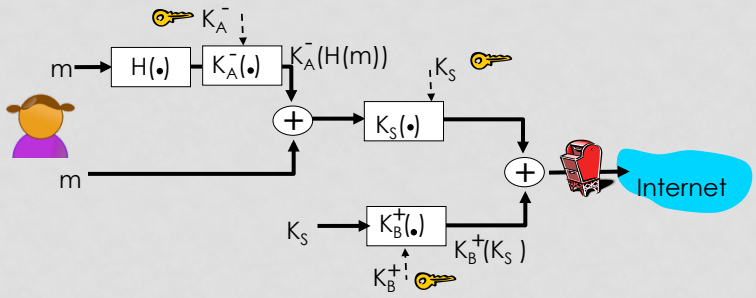


TCP API



TCP enhanced with SSL

SSL: Could Be Based on PGP



- But want to send byte streams
- Want certificate exchange to be part of protocol handshake phase

Basic SSL: A Simple Secure Channel

1. Handshake: Alice and Bob use their certificates and private keys to authenticate each other and exchange shared secret
2. Key Derivation: Alice and Bob use shared secret to derive set of keys – master key
3. Data Transfer: Data to be transferred is broken up into a series of records
4. Connection Closure: Special messages to securely close connection

→ Section 8.5.2 for more details

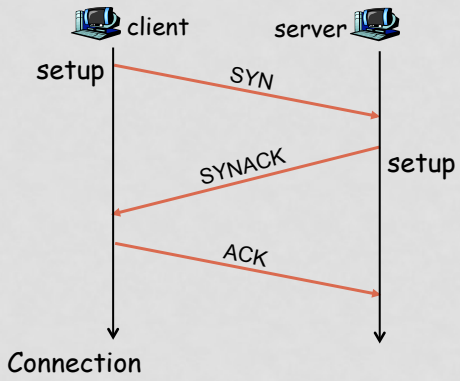
RECALL: TCP CONNECTION MANAGEMENT

Connection Set Up:

Step 1: client sends TCP SYN segment

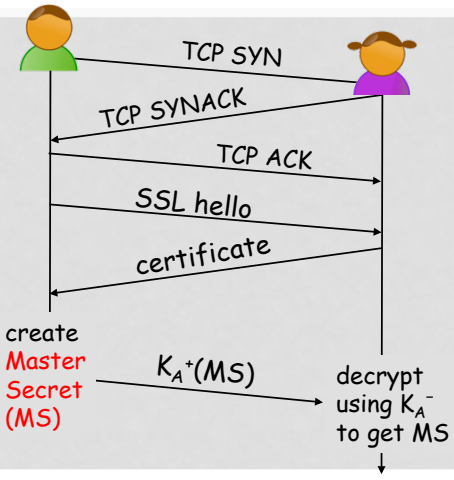
Step 2: server receives SYN and replies with SYNACK

Step 3: client receives SYNACK and replies with ACK



(1) SSL: HANDSHAKE

- Bob establishes TCP connection to Alice
- Authenticates Alice via CA signed certificate
- Creates, encrypts (using Alice's public key), & sends master secret key to Alice
  - nonce exchange not shown



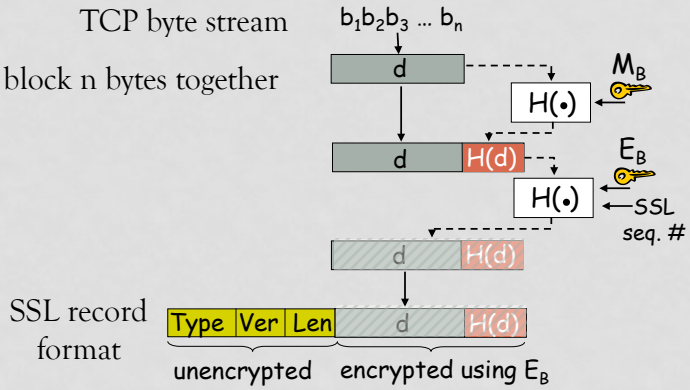
(2) SSL: Key Derivation

- Alice, Bob use shared secret (MS) to generate four keys:
  - E_B: Bob → Alice data encryption key
  - E_A: Alice → Bob data encryption key
  - M_B: Bob → Alice MAC key (the secret 'bit pattern')
  - M_A: Alice → Bob MAC key
- Encryption and MAC algorithms negotiable between hosts
- Why 4 keys?

(3) SSL: Data Records

- Encrypt data in a constant stream as we write it to TCP? … does not work because -
  - Where would we put the MAC?
- Instead, break stream into series of records
  - Each record carries a MAC
  - Receiver can act on each record as it arrives

(3) SSL: DATA TRANSFER



CHAPTER 8 TOPICS

8.1 What is network security?
8.2 Principles of cryptography
8.3 Message integrity
8.4 Securing e-mail
8.5 Securing TCP connections: SSL
8.6 Network layer security: IPsec
8.7 Securing wireless LANs
8.8 Operational security: firewalls and IDS

NETWORK SECURITY (SUMMARY)

Security Objectives……

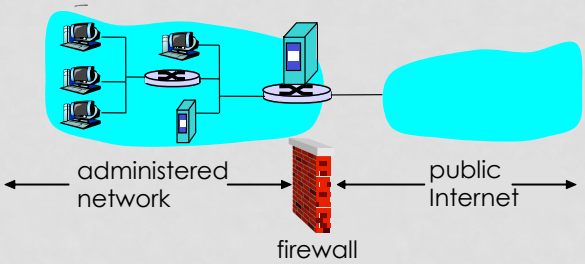
- cryptography (symmetric and public)
- message integrity
- end-point authentication

Used for numerous security scenarios

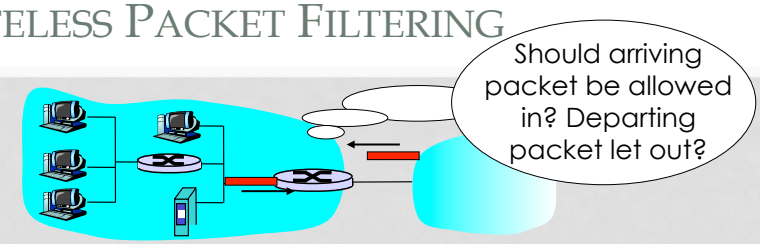
- secure email (PGP)
- secure transport (SSL)
- Operational Security: firewalls and IDS

FIREWALLS

- Isolate an organization's internal network from Internet, allowing some packets to pass, blocking others.
- **Which attacks are prevented?**



STATELESS PACKET FILTERING



- Internal network connected to Internet via router firewall
- Router filters packet-by-packet, decision to forward/drop packet based on:
  - Source IP address, destination IP address
  - TCP/UDP source and destination port numbers
  - ICMP message type
  - TCP SYN and ACK bits

STATELESS PACKET FILTERING: MORE EXAMPLES

- Where is a firewall implemented?

Policy	Firewall Setting
No outside Web access.	Drop all outgoing packets to any IP address, port 80
No incoming TCP connections, except those for institution's public Web server only.	Drop all incoming TCP SYN packets to any IP except 130.207.244.203, port 80
Prevent Web-radios from eating up the available bandwidth.	Drop all incoming UDP packets - except DNS and router broadcasts.
Prevent your network from being used for a smurf DoS attack.	Drop all ICMP packets going to a "broadcast" address (eg 130.207.255.255).
Prevent your network from being tracerouted	Drop all outgoing ICMP TTL expired traffic

Limitations of Firewalls

- IP spoofing: router can't know if data "really" comes from claimed source
- Filters often use all or nothing policy for UDP.
- Tradeoff: degree of communication with outside world, level of security
- Many highly protected sites still suffer from attacks.

Intrusion Detection Systems

- Deep packet inspection: look at packet contents (e.g., check character strings in packet against database of known virus, attack strings)
- Examine correlation among multiple packets
  - port scanning
  - network mapping
  - DoS attack

Network Security (Summary)

Basic techniques…

- cryptography (symmetric and public)
- message integrity
- end-point authentication

… used in many different security scenarios

- secure email
- secure transport (SSL)

Operational Security: firewalls and IDS