CS 550:
Advanced Operating Systems

Security

Ioan Raicu
Computer Science Department
Illinois Institute of Technology

CS 550
Advanced Operating Systems
March 24th, 2011

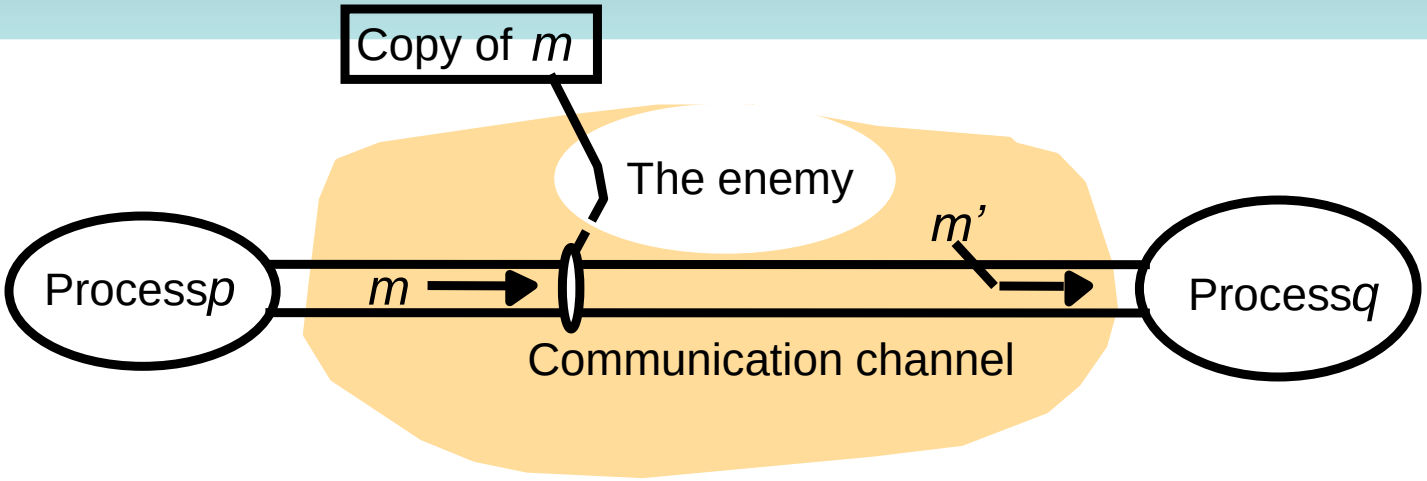
Outline

- Security issues:
  - Threats
  - Methods of attack
- Encryption algorithms
  - Secret-key
  - Public-key
  - Hybrid protocols

Historical context

	1965-75	*1975-89*	*1990-99*	*Current*
Platforms	Multi-user timesharing computers	Distributed systems based on local networks	The Internet, wide-area services	The Internet + mobile devices
Shared resources	Memory, files	Local services (e.g. NFS), local networks	Email, web sites, Internet commerce	Distributed objects, mobile code
Security requirements	User identification and authentication	Protection of services	Strong security for commercial transactions	Access control for individual objects, secure mobile code
Security management environment	Single authority, single authorization database (e.g. /etc/ passwd)	Single authority, delegation, repli-cated authorization databases (e.g. NIS)	Many authorities, no network-wide authorities	Per-activity authorities, groups with shared responsibilities

Security Problems

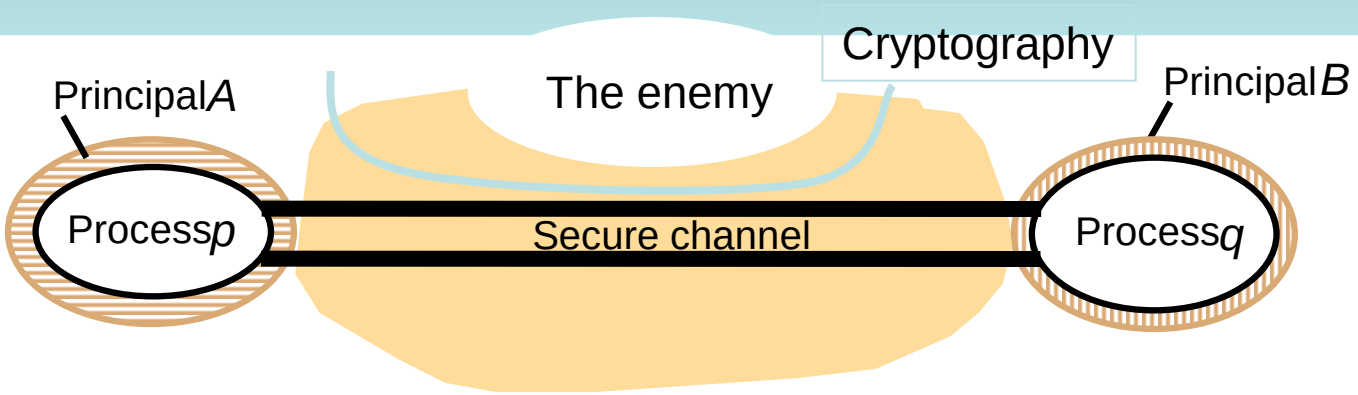


- Attacks
  - On applications that handle financial transactions or other information whose secrecy or integrity is crucial
- Enemy (or adversary)
- Threats
  - To processes, to communication channels, denial of service

Threats/Methods of Attacks

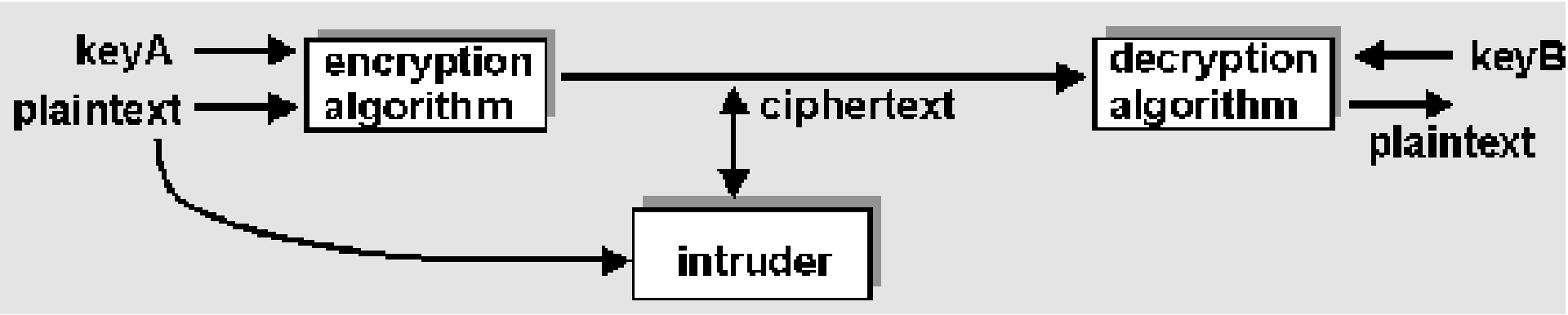
- Eavesdropping:
  - Obtain private or secret information
- Masquerading
  - Assume the identity of another user
- Message tampering
  - Alter the content of messages in transit
    - Man-in-the-middle attack
- Replaying
  - Store secure msgs and send them at a later data
- Denial of service
  - Flood a channel or other resources, denying access to others

Secure channels



- Properties:
  - Each proc is sure of the identity of the other
  - Data is private and protected against tampering
  - Protection against repetition and reordering of data
- Important issues:
  - Cryptography
  - Authentication

Encryption



plaintext: unencrypted message

ciphertext: encrypted form of message

Intruder may

- intercept ciphertext transmission
- intercept plaintext/ciphertext pairs
- obtain encryption decryption algorithms

A simple encryption algorithm

Substitution cipher:

```
abcdefghijklmnopqrstuvwxyz

poiuytrewqasdfghjklmnbvczx
```

- replace each plaintext character in message with matching ciphertext character:

plaintext: `Charlotte, my love`

ciphertext: `iepksgmmy, dz sgby`

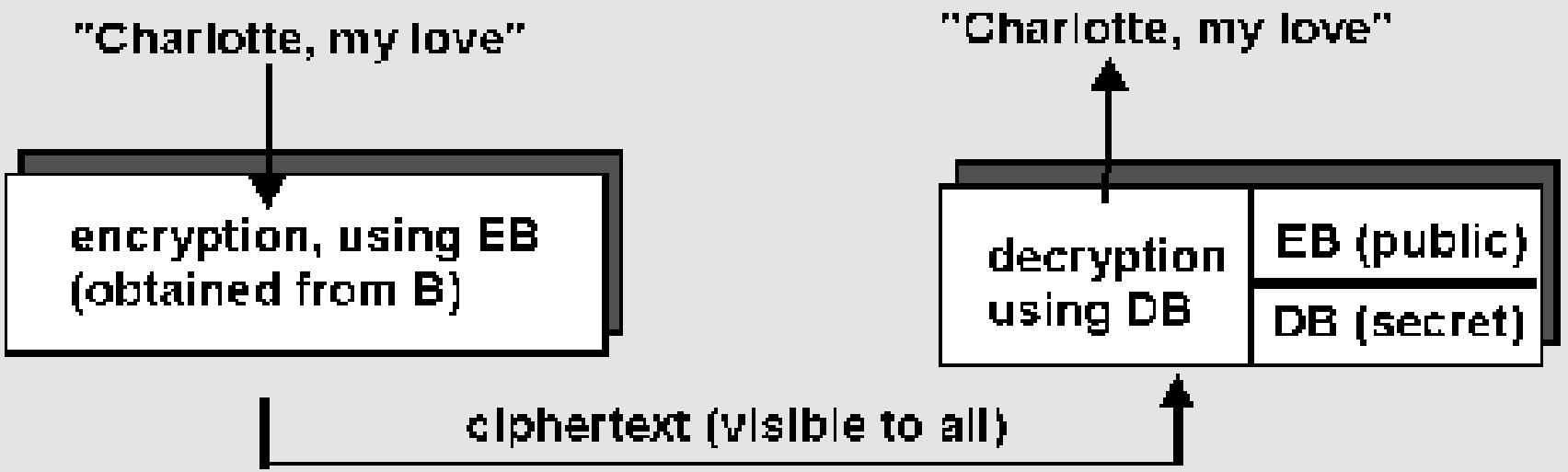
A simple encryption Alg (cont.)

- key is pairing between plaintext characters and ciphertext characters
- 26! (approx 10^26) different possible keys: unlikely to be broken by random trials
- substitution cipher subject to decryption using observed frequency of letters
  - 'e' most common letter, 'the' most common word

Public Key Cryptography

- Separate encryption/decryption keys
  - Receiver makes *known* (!) its encryption key
  - Receiver keeps its decryption key secret
- To send to receiver B:
- To decrypt:

Public Key Cryptography



- Knowing encryption key does not help with decryption; decryption is a non-trivial inverse of encryption
- Only receiver can decrypt message

Question: good encryption/decryption algorithms

RSA: public key encryption/decryption

RSA: a public key algorithm for encrypting/decrypting

Entity wanting to receive encrypted messages:

to break RSA:

- need to know *p, q*, given *pq=n*, *n* known
- factoring 200 digit *n* into primes takes 4 billion years using known methods

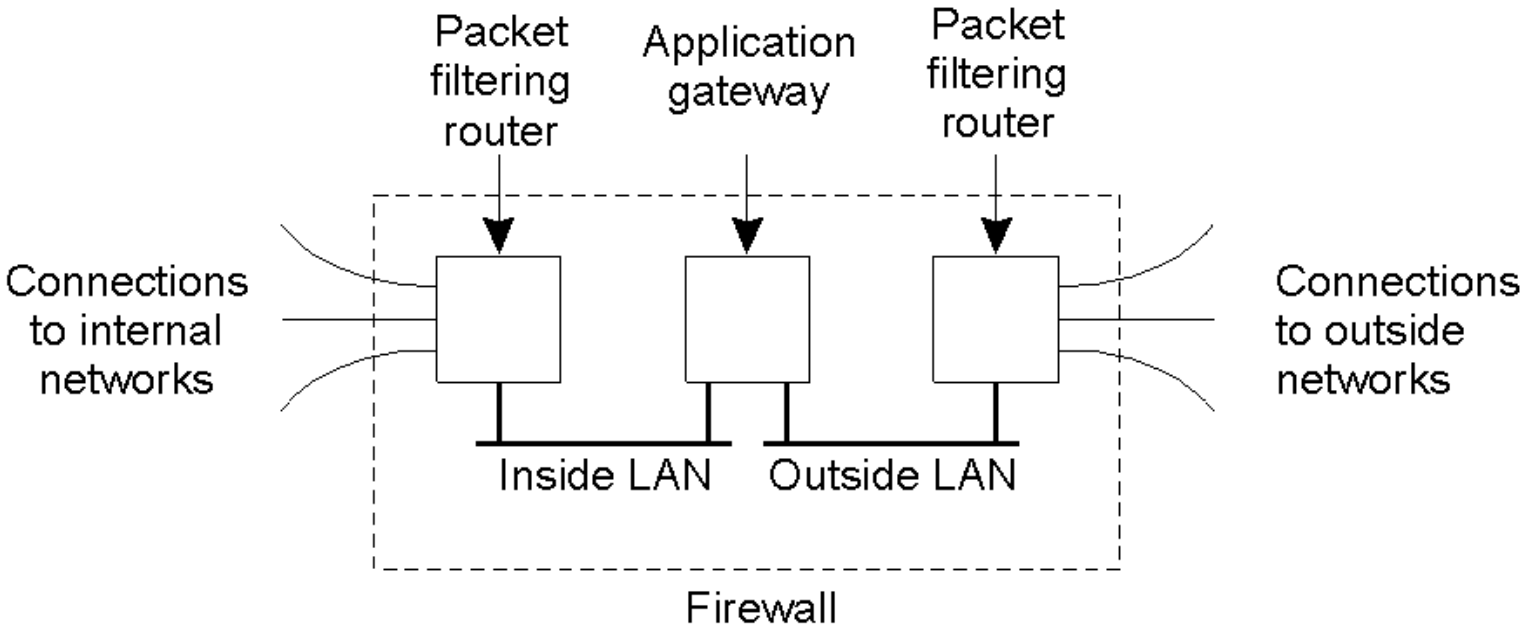
Authentication

- **Question:** how does a receiver know that remote communicating entity is who it is claimed to be?

Authentication Protocol (ap)

- Ap 1.0
  - Alice to Bob: “I am Alice”
  - Problem: ?
- Ap 2.0
  - Authenticate source IP address is from Alice’s machine
  - Problem: ?
- Ap 3.0: use a secret password
  - Alice to Bob: “I am Alice, here is my password” (e.g., telnet)
  - Problem: ?

Protection Against Intruders: Firewalls



Firewalls

Firewall: network components (host/router+software) sitting between inside ("us") and outside ("them)

Packet filtering firewalls: drop packets on basis of source or destination address (i.e., IP address, port)

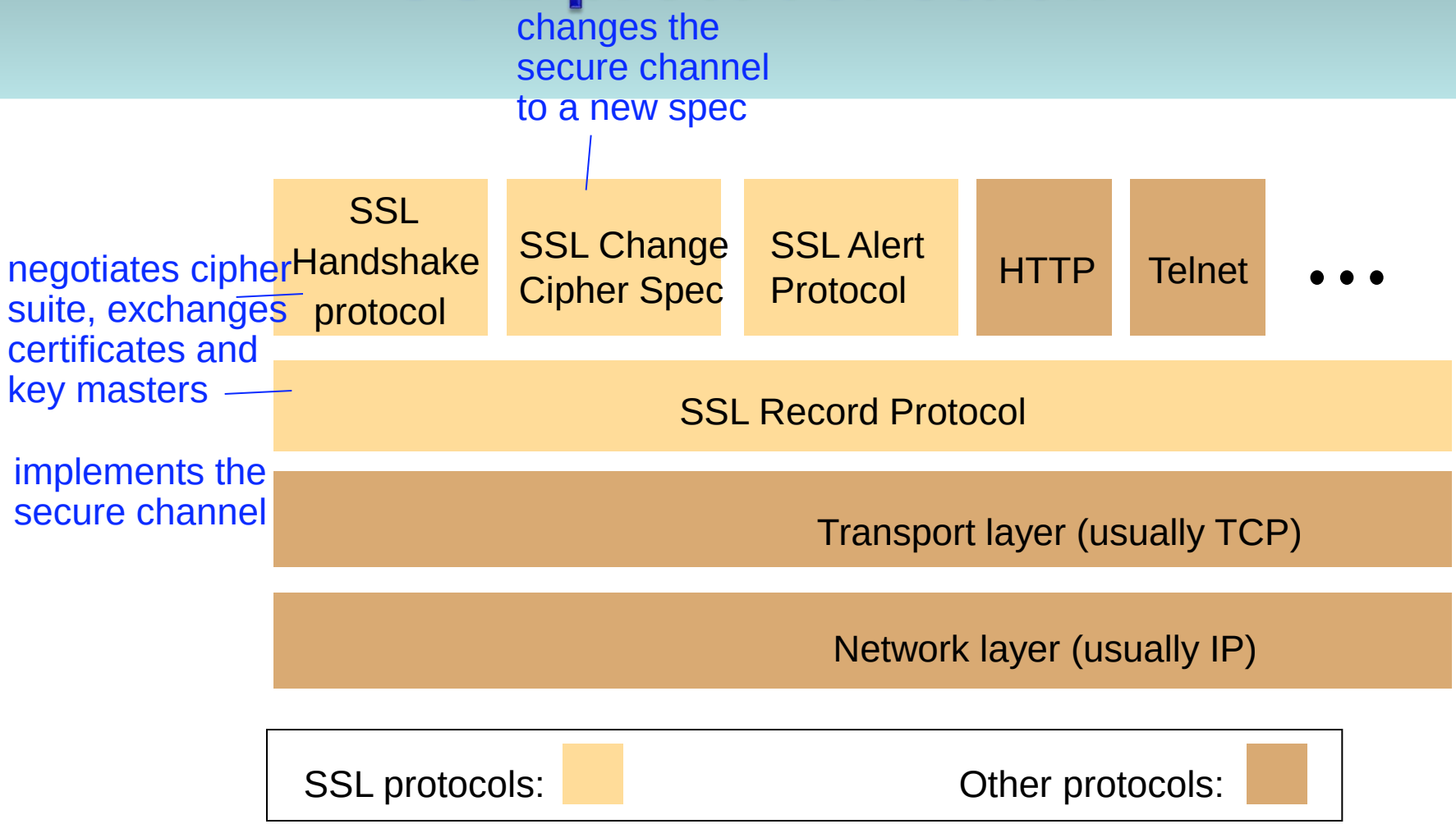
Application gateways: application specific code intercepts, processes and/or relays application specific packets

- e.g., email of telnet gateways
- application gateway code can be security hardened
- can log all activity

Secure Sockets Layer (SSL)

- SSL: Developed by Netscape
  - Provides data encryption and authentication between web server and client
  - SSL lies above the transport layer
  - Features:
    - SSL server authentication
    - Encrypted SSL session
    - SSL client authentication

SSL protocol stack



Secure Socket Layer

- Protocol: https instead of http
  - Steps?
  - Browser -> Server: B's SSL version and preferences
  - S->B: S's SSL version, preferences, and *certificate*
    - Certificate: server's RSA public key encrypted by CA's private key
  - B: uses its list of CAs and public keys to *decrypt S's public key*
  - B->S: generate K, encrypt K with with E_S
  - B->S: "future messages will be encrypted", and K(m)
  - S->B: "future messages will be encrypted", and K(m)
  - SSL session begins…

Security: conclusion

key concerns:

- encryption
- authentication
- key exchange

also:

- increasingly an important area as network connectivity increases
- digital signatures, digital cash, authentication, increasingly important
- an important social concern
- further reading:
  - Crypto Policy Perspectives: S. Landau et al., Aug 1994 CACM
  - Internet Security, R. Oppliger, CACM May 1997
  - www.eff.org

Questions