Networks Lecture 13

Paolo Ciancarini

Innopolis University

February 28, 2022

Source of the material

- This lecture is based on the following resources
  - Chapter 8 of Computer Networking: A Top Down Approach (8th edition) by Jim Kurose and Keith Ross
  - The material is aligned and add/deleted according to the need of the students.

Topic of the lecture

- What is network security?
- Principles of cryptography
- Message integrity, authentication
- Securing e-mail
- Securing TCP connections: SSL
- Network layer security: IPsec
- Securing wireless LANs
- Operational security: firewalls and IDS

Terminologies

Confidentiality: only sender, intended receiver should "understand" message contents

- sender encrypts message
- receiver decrypts message

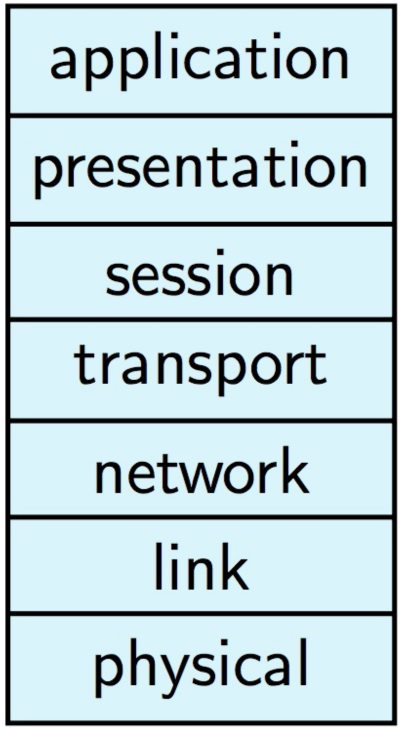
Authentication: sender, receiver want to confirm identity of each other

Message integrity: sender, receiver want to ensure message not altered (in transit, or afterwards) without detection

Access and availability: services must be accessible and available to users

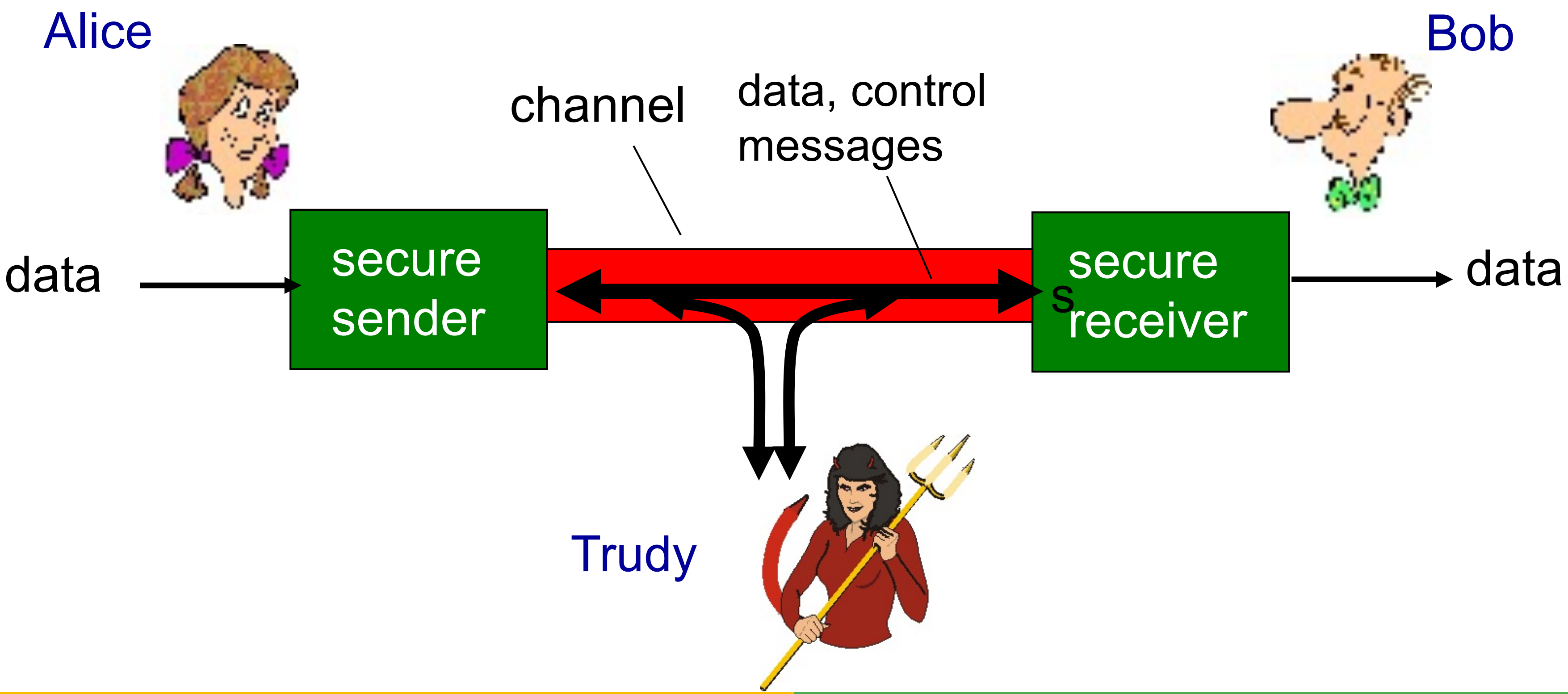
Network Security and ISO-OSI layers

- Security services at the top layers can be tailored for specific applications, but each application then needs a separate service
- Security services at the bottom layers can protect the upper layers transparently, but may not meet all requirements of specific applications



Friends and enemies: Alice, Bob, Trudy

- Alice, Bob, Trudy: well-known in network security world
- Alice & Bob want to communicate "securely" to each other
- Trudy (intruder) may intercept, delete, add messages



Who might Bob, Alice be?

- … well, *real-life* Bobs and Alices!
- Web browser/server for electronic transactions (e.g., on-line purchases)
- on-line banking client/server
- DNS servers
- routers exchanging routing table updates
- Multiplayer games with people trying to cheat
- other examples?

There are bad guys out there!

Q: What can a "bad guy" do?

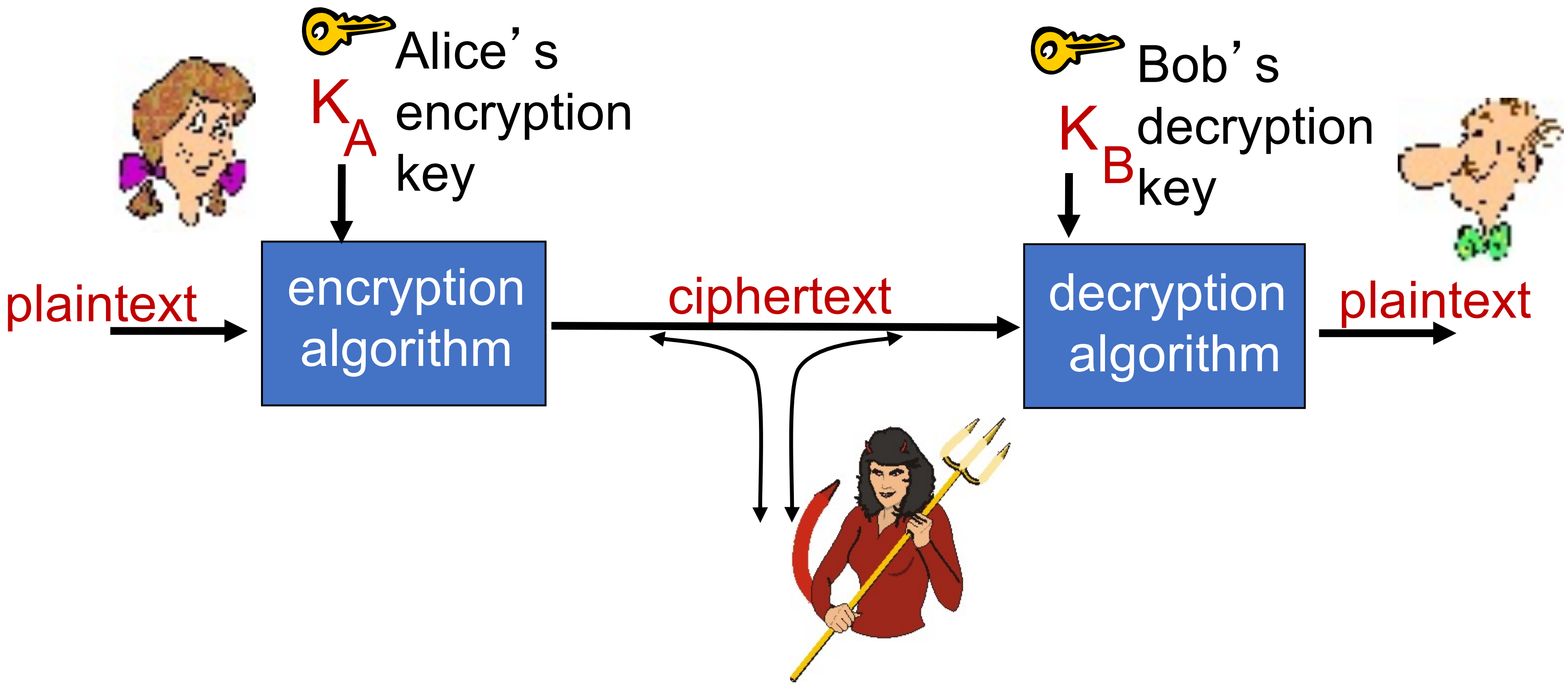
A: A lot!

- *eavesdrop:* intercept messages
- actively *insert* fake messages into connection
- *impersonation:* can fake (spoof) source address in packet (or any field in packet)
- *hijacking:* "take over" ongoing connection by removing sender or receiver, inserting himself in place
- *denial of service*: prevent service from being used by others (e.g., by overloading resources)

Topic of the lecture

- What is network security?
- Principles of cryptography
- Message integrity, authentication
- Securing e-mail
- Securing TCP connections: SSL
- Network layer security: IPsec
- Securing wireless LANs
- Operational security: firewalls and IDS

The language of cryptography



m plaintext message

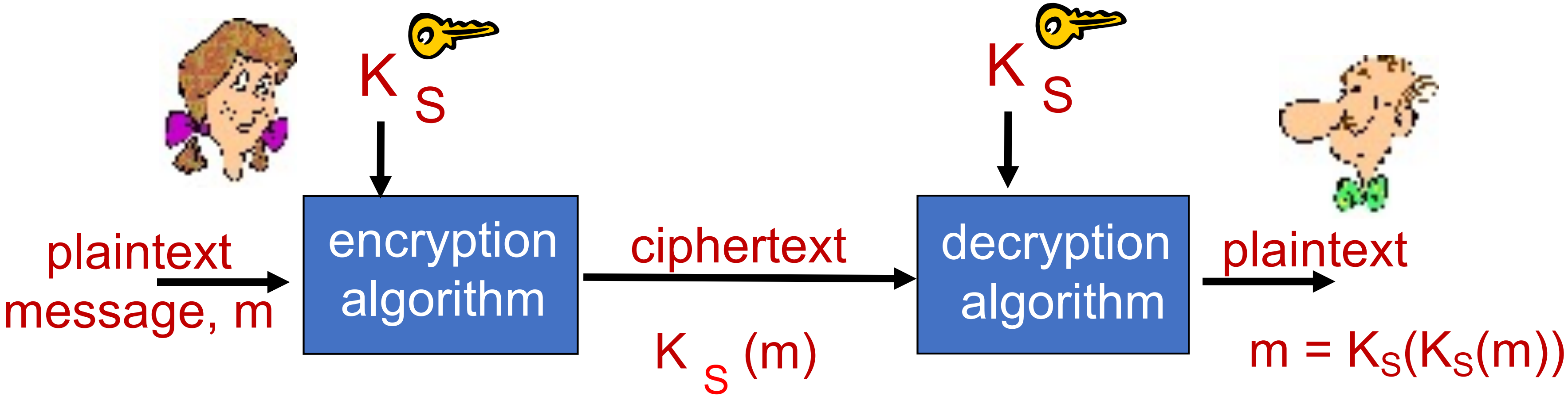
$K_A(m)$ ciphertext, encrypted with key K_A

$m = K_B(K_A(m))$

Breaking an encryption scheme

- Cipher-text only attack: Trudy has ciphertext she can analyze
- Two approaches:
  - Brute force: search through all keys
  - Statistical analysis
- Known-plaintext attack: Trudy has plaintext corresponding to ciphertext
  - e.g., in monoalphabetic cipher, Trudy determines pairings for a,l,i,c,e,b,o,
- Chosen-plaintext attack: Trudy can get ciphertext for chosen plaintext

Symmetric key cryptography



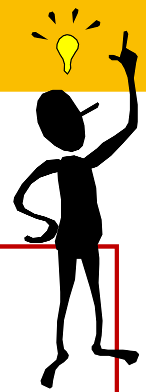
Symmetric key crypto: Bob and Alice share same (symmetric) key: K_S

Advanced Encryption Standard (AES)

- Symmetric-key NIST standard, replaced DES (Nov 2001)
- Processes data in 128 bit blocks
- 128, 192, or 256 bit keys
- Brute force decryption (try each possible key) taking 1 sec on DES, takes 149 trillion years for AES

Data Encryption Standard (DES)
US encryption standard [NIST 1993]

Public Key Cryptography



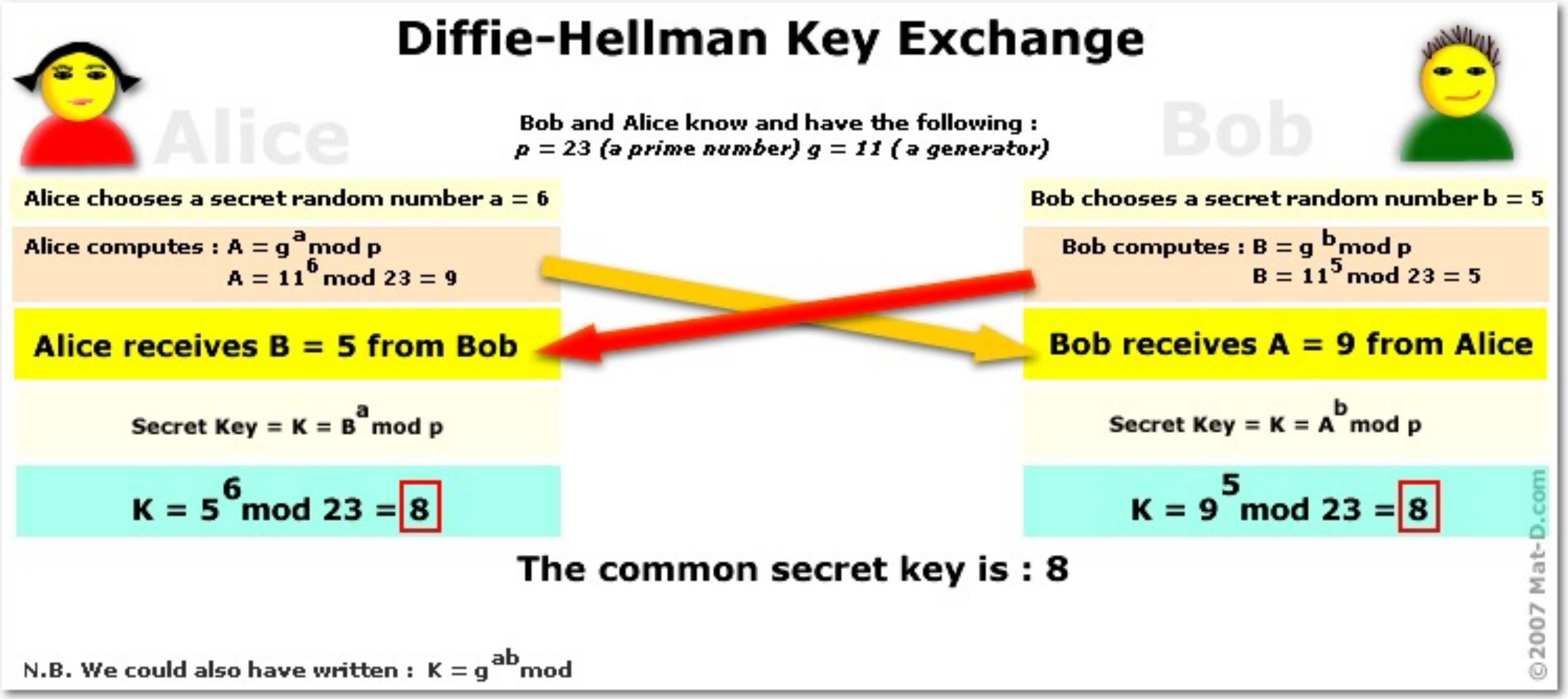
Symmetric key crypto

- Requires sender, receiver know shared secret key
- Q: how to agree on key in first place (particularly if never “met”)?

Public key crypto

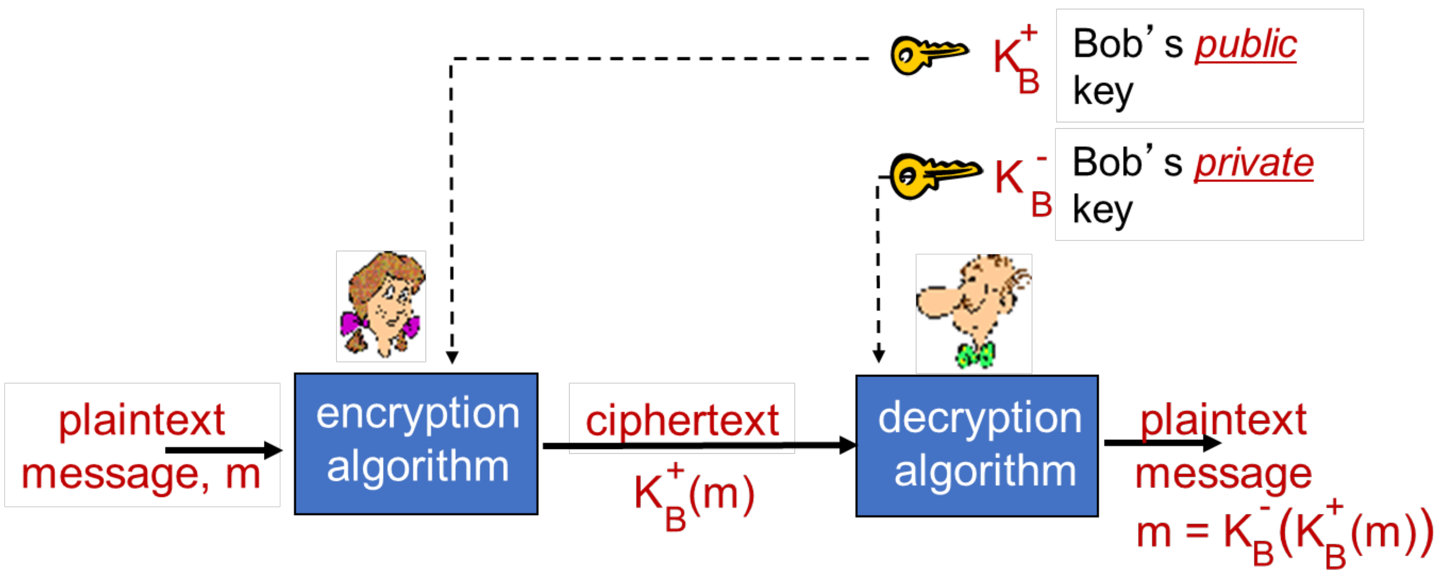
- Radically different approach [Diffie-Hellman76, RSA78]
- Sender, receiver do *not* Share secret key
- *Public* encryption key known to *all*
- *Private* decryption key known only to receiver

Diffie-Hellman Public Key Exchange Algorithm



Source: http://www.mat-d.com/site/rsa-diffie-hellman-explained-in-3-minutes/

Asymmetric Cryptography



- Asymmetric cryptography is often ***referred to as "public key" cryptography***.
- In this process, two different keys are used.
- However, the keys are linked to each other mathematically.
- One is referred to as **public** and the other as **private**.
- The public key can be used by anyone. The private one is a secret.

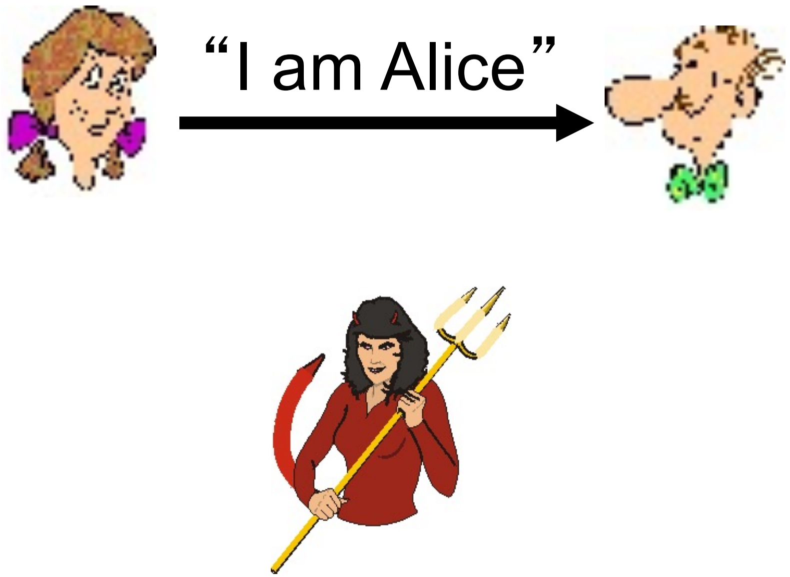
Topic of the lecture

- What is network security?
- Principles of cryptography
- Message integrity, authentication
- Securing e-mail
- Securing TCP connections: SSL
- Network layer security: IPsec
- Securing wireless LANs
- Operational security: firewalls and IDS

Authentication

Goal: Bob wants Alice to "prove" her identity to him

Protocol ap1.0: Alice says "I am Alice"

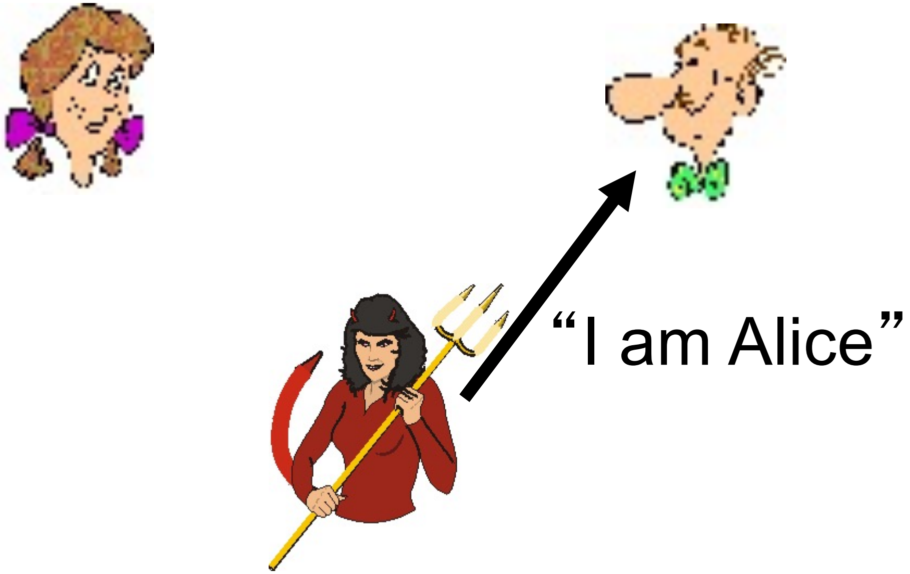


Failure scenario??

Authentication

Goal: Bob wants Alice to "prove" her identity to him

Protocol ap1.0: Alice says "I am Alice"



in a network,
Bob can not "see" Alice,
so Trudy simply declares
herself to be Alice

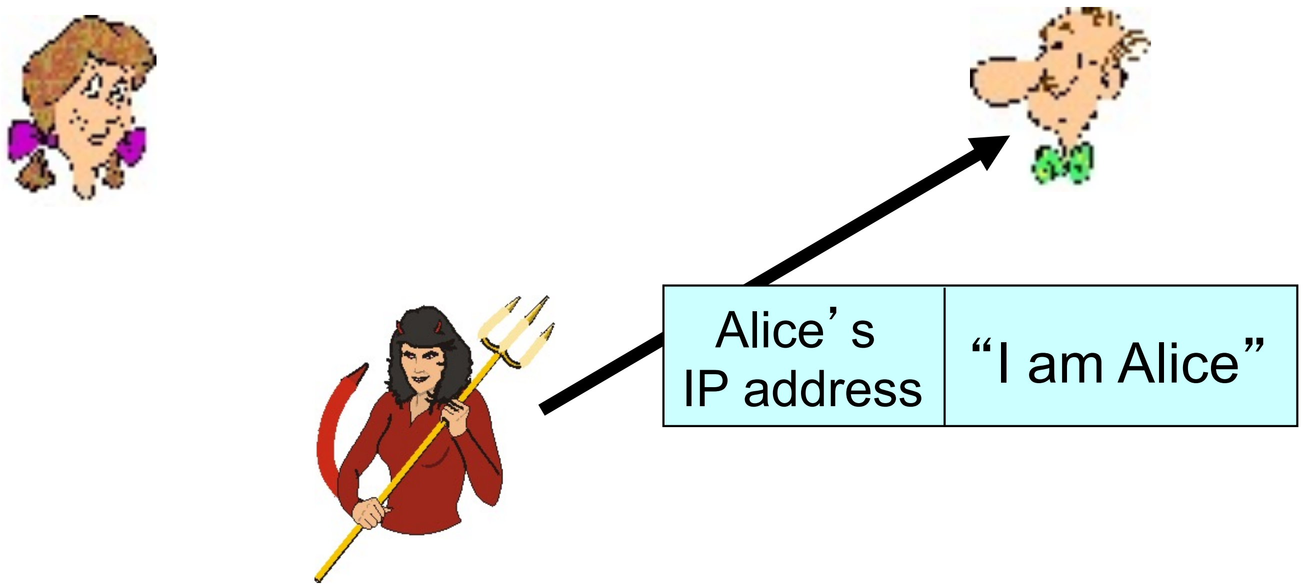
Authentication: another try

Protocol ap2.0: Alice says "I am Alice" in an IP packet containing her source IP address

Failure scenario??

Authentication: another try

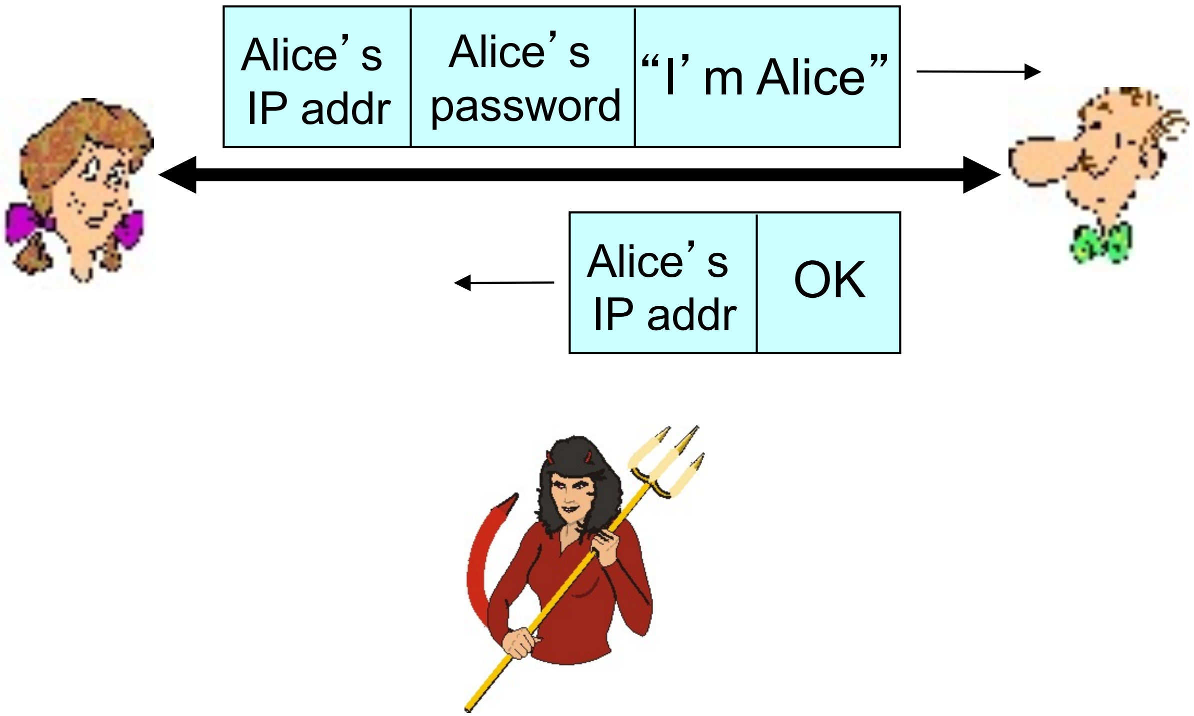
Protocol ap2.0: Alice says "I am Alice" in an IP packet containing her source IP address



Trudy can create a packet "spoofing" Alice's address

Authentication: another try

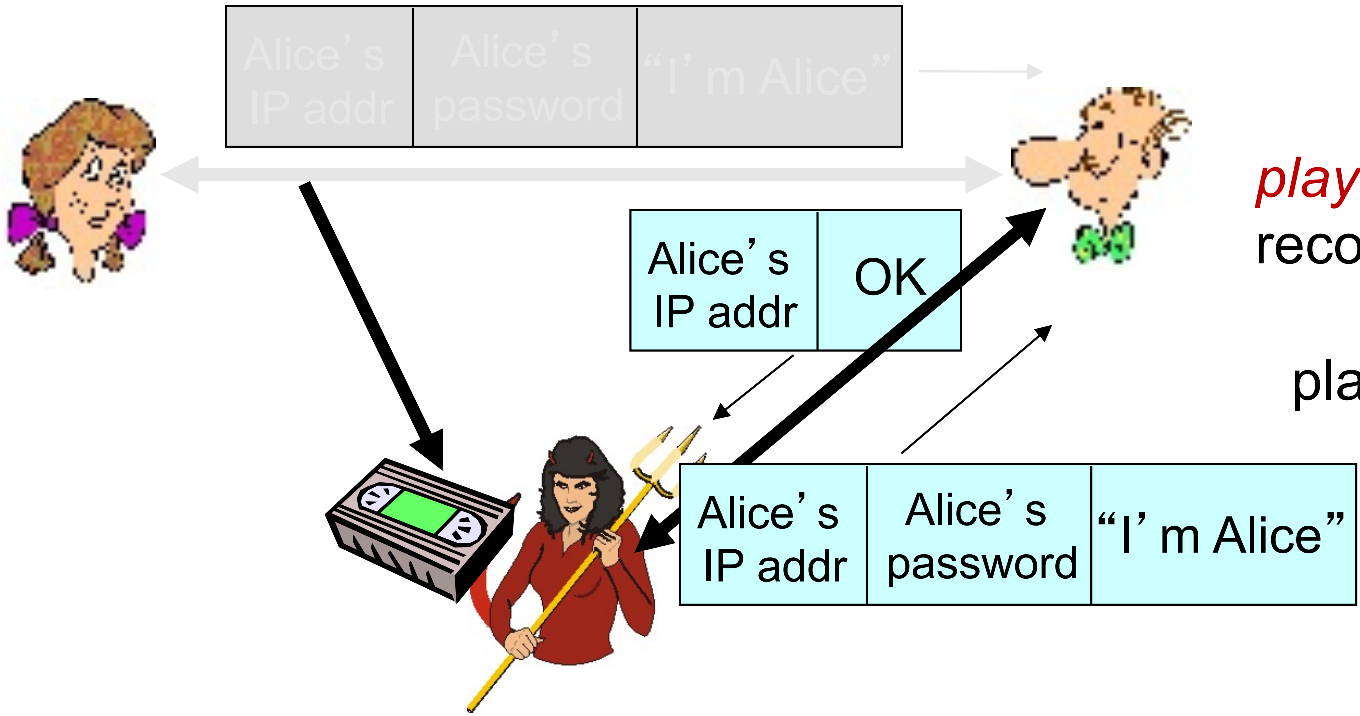
Protocol ap3.0: Alice says "I am Alice" and sends her secret password to "prove" it.



Failure scenario??

Authentication: another try

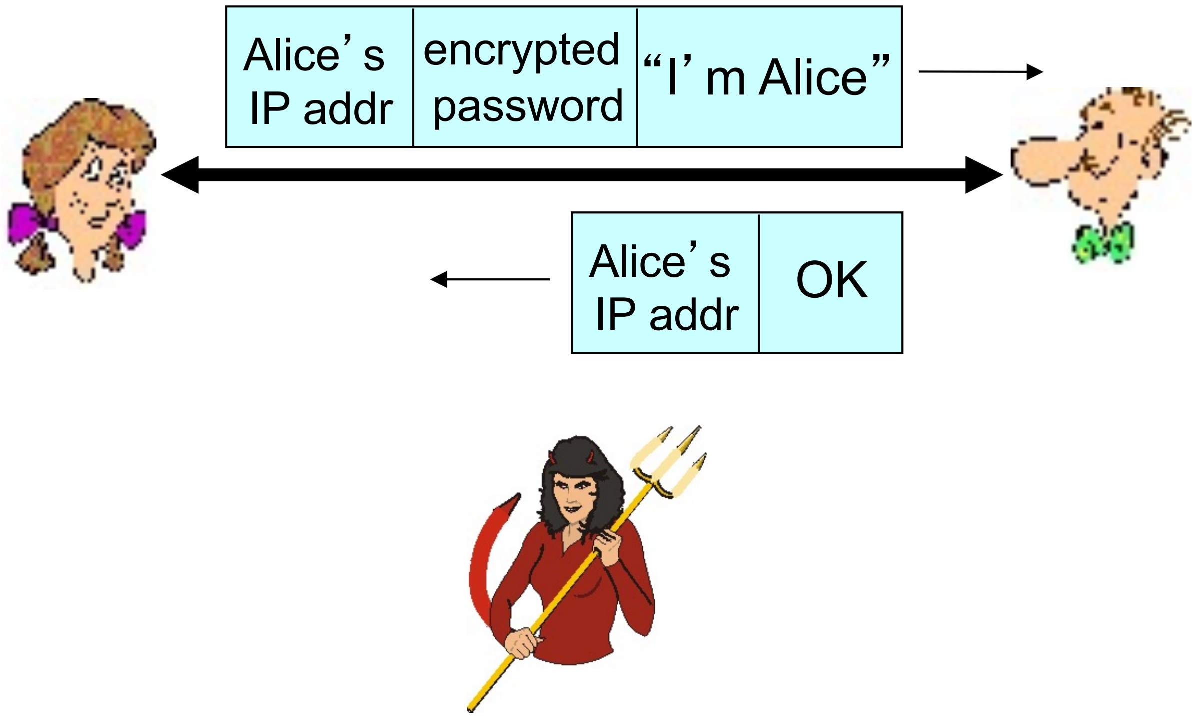
Protocol ap3.0: Alice says "I am Alice" and sends her secret password to "prove" it.



playback attack: Trudy records Alice's packet and later plays it back to Bob

Authentication: yet another try

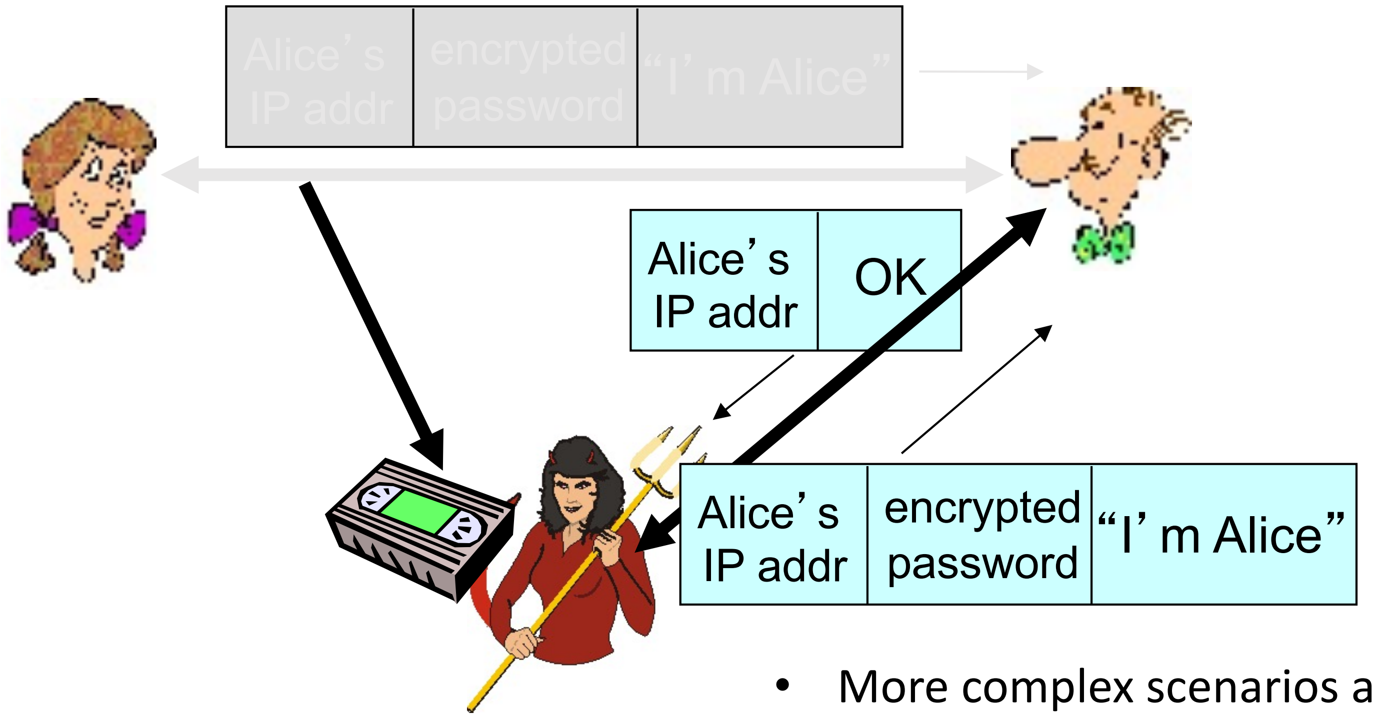
Protocol ap3.1: Alice says "I am Alice" and sends her *encrypted* secret password to "prove" it.



Failure scenario??

Authentication: yet another try

Protocol ap3.1: Alice says "I am Alice" and sends her *encrypted* secret password to "prove" it.



record and playback *still* works!

- More complex scenarios and schemes are possible

Topic of the lecture

- What is network security?
- Principles of cryptography
- Message integrity, authentication
- Securing e-mail
- Securing TCP connections: SSL
- Network layer security: IPsec
- Securing wireless LANs
- Operational security: firewalls and IDS

Digital signatures

Cryptographic technique analogous to hand-written signatures:

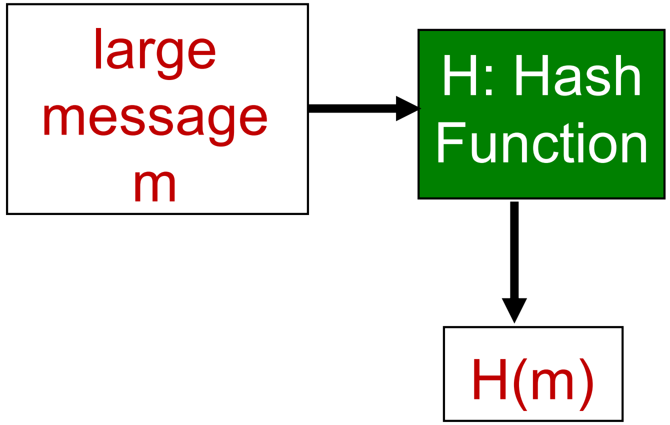
- sender (Bob) digitally signs document, establishing he is document owner/creator.
- *verifiable, nonforgeable:* recipient (Alice) can prove to someone that Bob, and no one else (including Alice), must have signed document

Message digests

computationally expensive to public-key-encrypt long messages

goal: fixed-length, easy- to-compute digital "fingerprint"

- apply hash function H to *m*, get fixed size message digest, *H(m)*.



Hash function properties:

- many-to-1
- produces fixed-size msg digest (fingerprint)
- given message digest x, computationally infeasible to find m such that x = H(m)

Internet checksum: poor crypto hash function

Internet checksum has some properties of hash function:

- produces fixed length digest (16-bit sum) of message
- is many-to-one

But given message with given hash value, it is easy to find another message with same hash value:

message	ASCII format
I O U 1	49 4F 55 31
0 0 . 9	30 30 2E 39
9 B O B	39 42 D2 42
	B2 C1 D2 AC

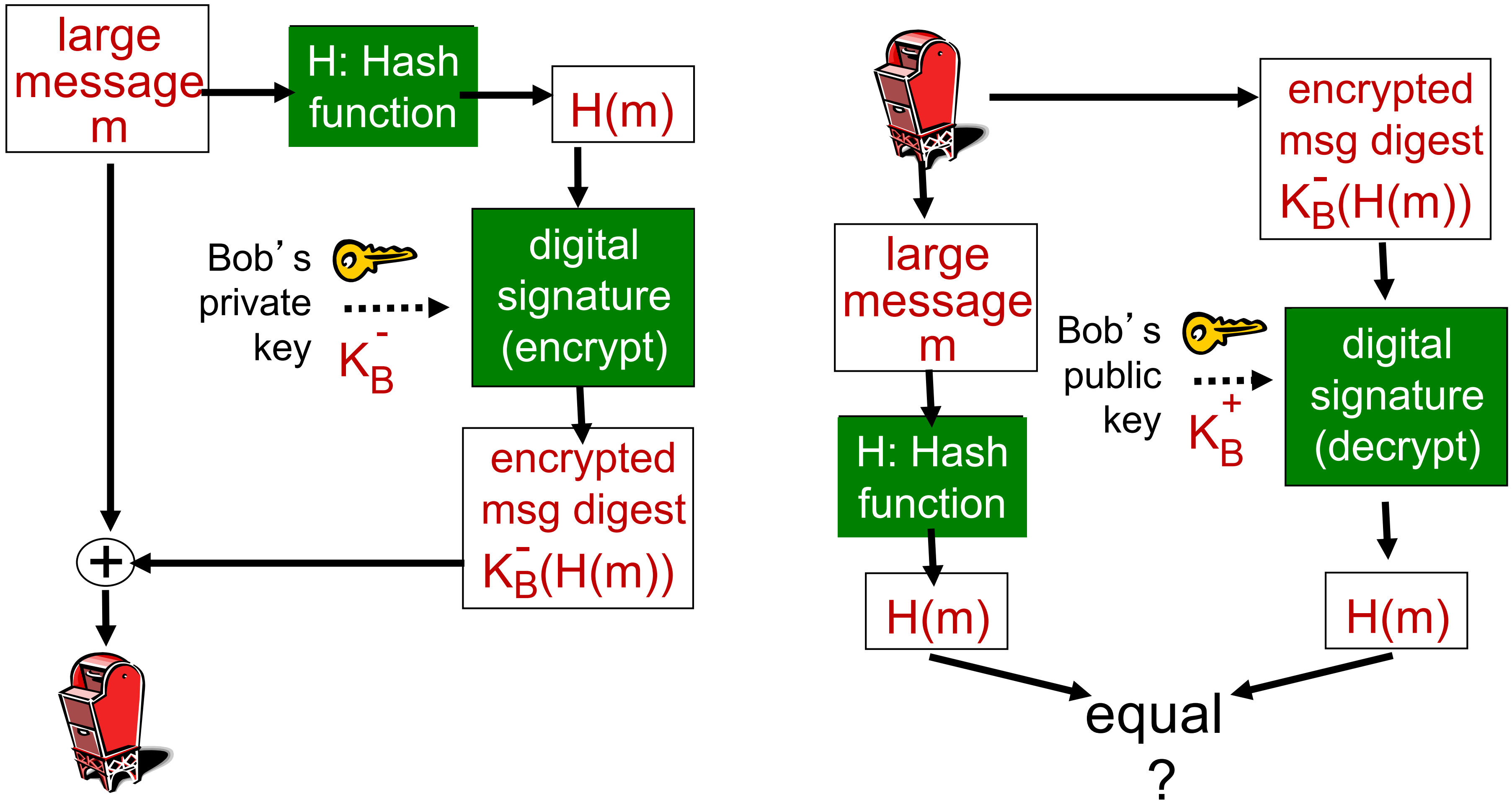
message	ASCII format
I O U <u>9</u>	49 4F 55 <u>39</u>
0 0 . <u>1</u>	30 30 2E <u>31</u>
9 B O B	39 42 D2 42
	B2 C1 D2 AC

different messages but identical checksums!

Digital signature = signed message digest

Bob sends digitally signed message:

Aice verifies signature, integrity of digitally signed message:



Hash function algorithms

- MD5 hash function widely used (RFC 1321)
  - computes 128-bit message digest in 4-step process.
  - arbitrary 128-bit string x, appears difficult to construct msg m whose MD5 hash is equal to x

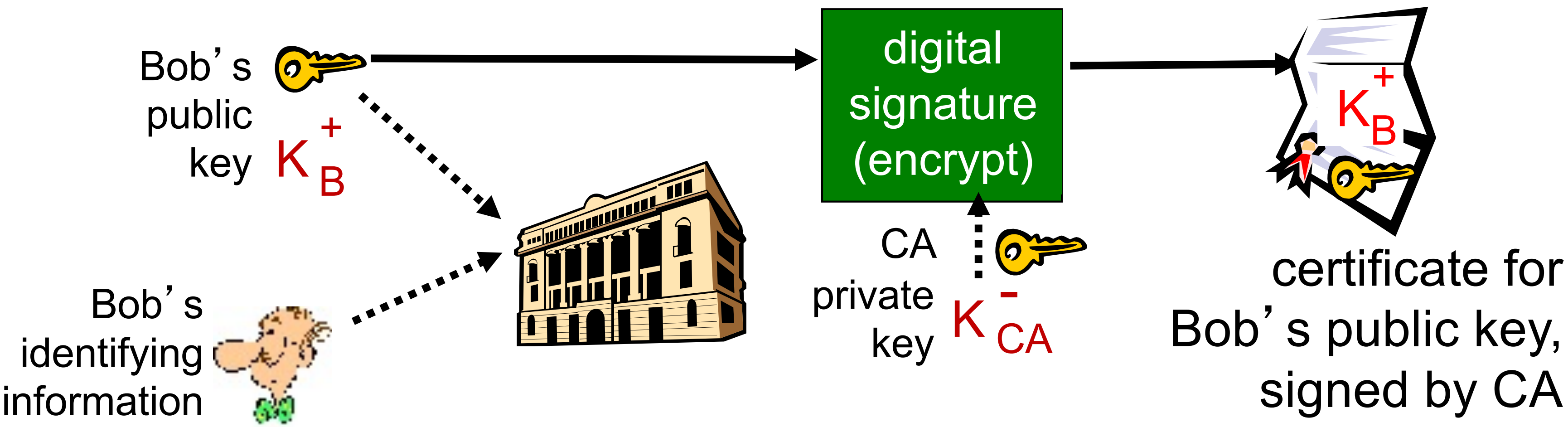
- SHA-1 is also used
  - US standard [NIST, FIPS PUB 180-1]
  - 160-bit message digest

Public-key certification

- Motivation: Trudy plays pizza prank on Bob
  - Trudy creates e-mail order:
    Dear Pizza Store, Please deliver to me four pepperoni pizzas. Thank you, Bob
  - Trudy signs order with her private key
  - Trudy sends order to Pizza Store
  - Trudy sends to Pizza Store her public key, but says it's Bob's public key
  - Pizza Store verifies signature; then delivers four pepperoni pizzas to Bob
  - Bob doesn't even like pepperoni

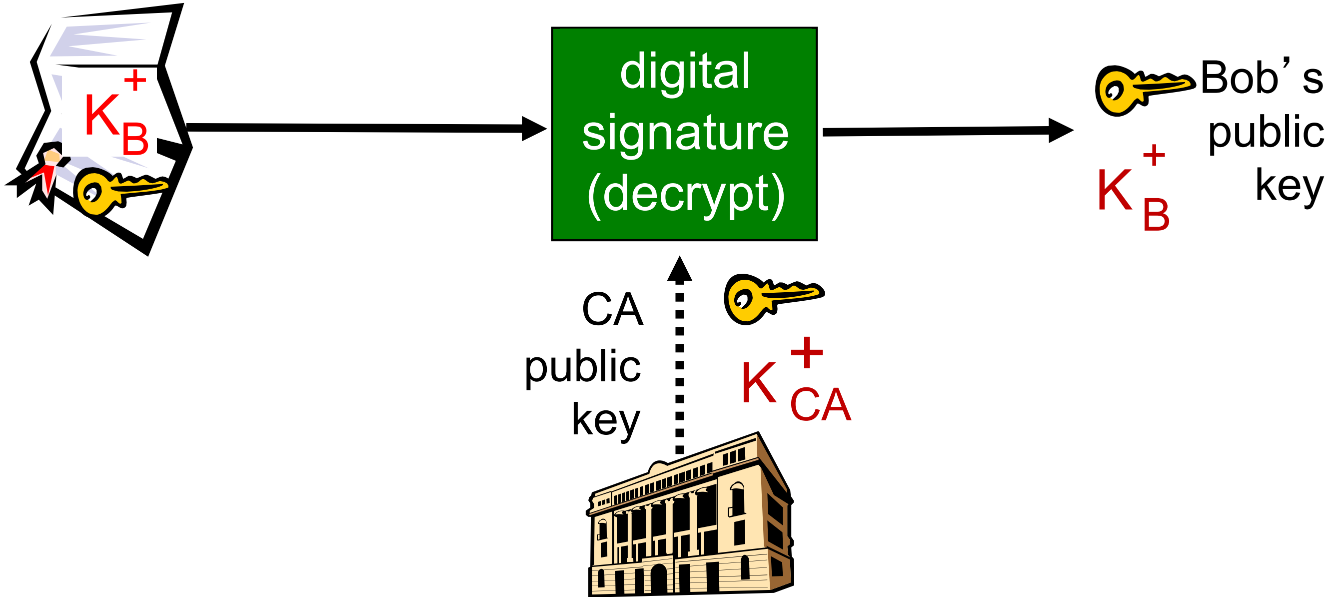
Certification authorities

- *Certification Authority (CA):* binds public key to particular entity, E.
- E (person, router) registers its public key with CA.
  - E provides "proof of identity" to CA.
  - CA creates certificate binding E to its public key.
  - certificate containing E's public key digitally signed by CA – CA says "this is E's public key"



Certification authorities

- when Alice wants Bob's public key:
  - gets Bob's certificate (Bob or elsewhere).
  - apply CA's public key to Bob's certificate, get Bob's public key

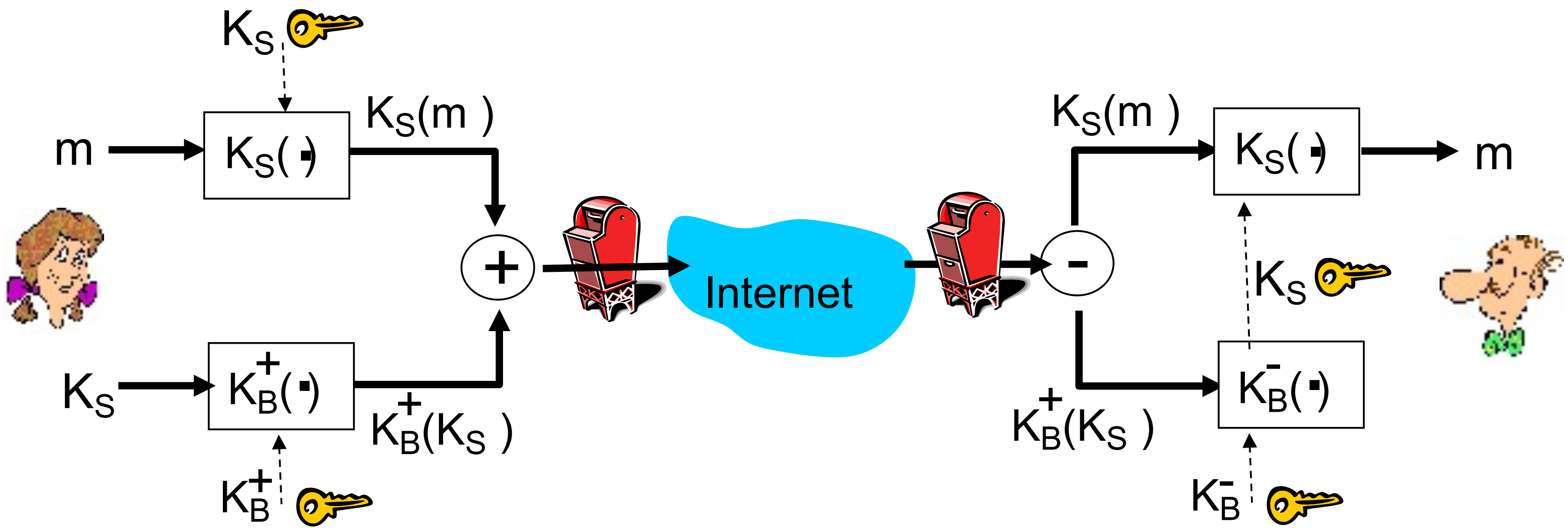


Topic of the lecture

- What is network security?
- Principles of cryptography
- Message integrity, authentication
- Securing e-mail
- Securing TCP connections: SSL
- Network layer security: IPsec
- Securing wireless LANs
- Operational security: firewalls and IDS

Secure e-mail

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

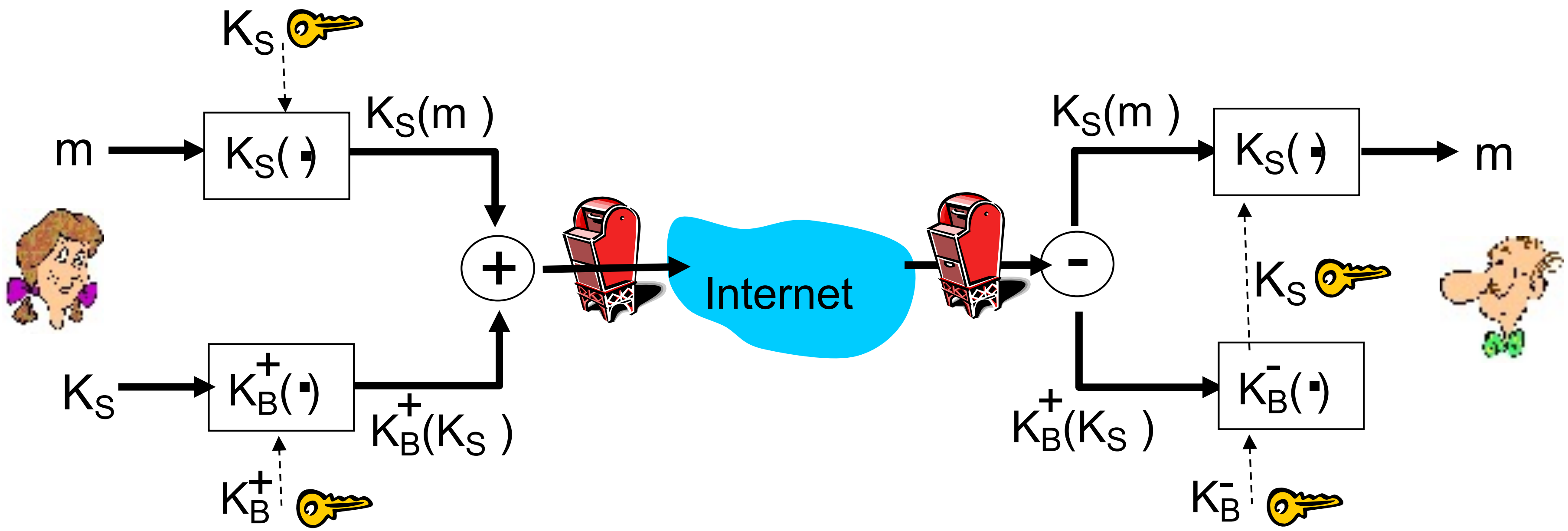


Alice:

- generates random *symmetric* private key, K_S
- encrypts message with K_S (for efficiency)
- also encrypts K_S with Bob's public key
- sends both $K_S(m)$ and $K_B(K_S)$ to Bob

Secure e-mail

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

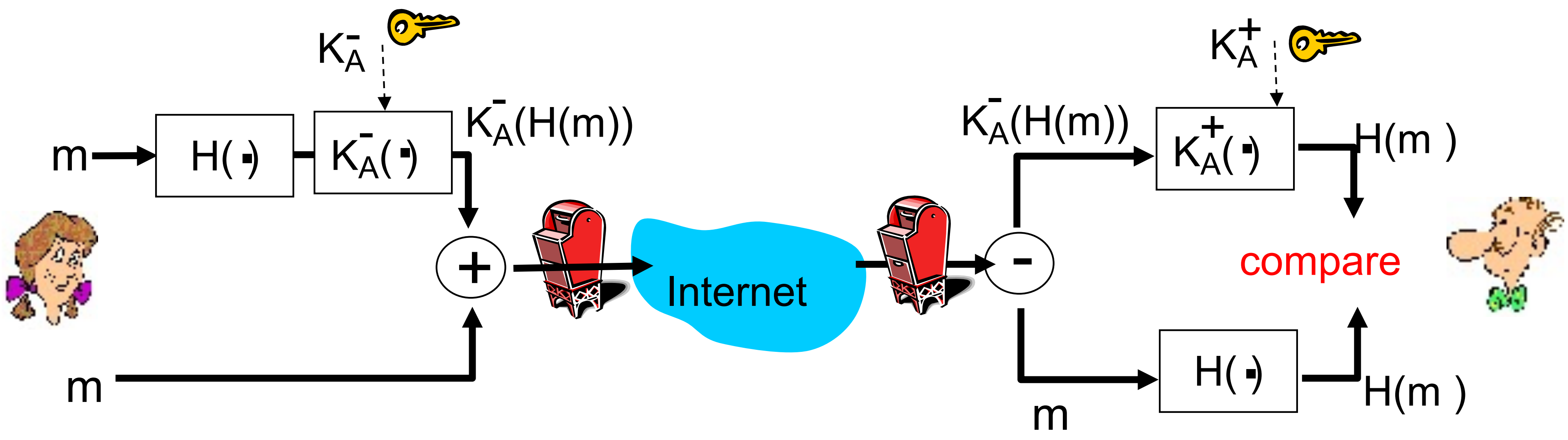


Bob:

- uses his private key to decrypt and recover K_S
- uses K_S to decrypt $K_S(m)$ to recover m

Secure e-mail (continued)

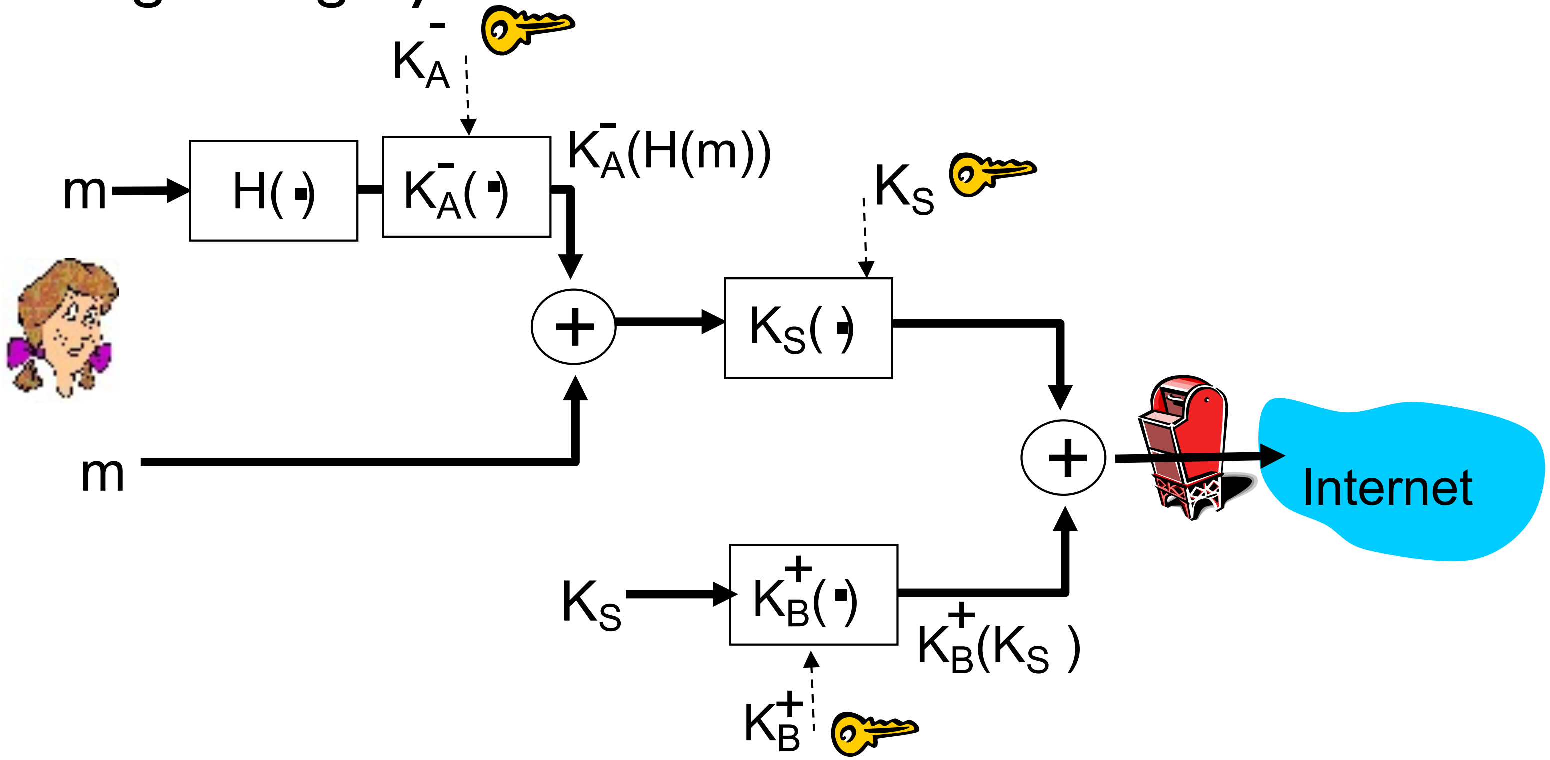
- Alice wants to provide sender authentication message integrity



- Alice digitally signs message
- sends both message (in the clear) and digital signature

Secure e-mail (continued)

- Alice wants to provide secrecy, sender authentication, message integrity.



Alice uses three keys: her private key, Bob's public key, newly created symmetric key

Topic of the lecture

- What is network security?
- Principles of cryptography
- Message integrity, authentication
- Securing e-mail
- Securing TCP connections: SSL
- Network layer security: IPsec
- Securing wireless LANs
- Operational security: firewalls and IDS

SSL: Secure Sockets Layer

- Widely deployed security protocol
  - supported by almost all browsers, web servers
  - https
  - billions $/year over SSL
- Mechanisms: [Woo 1994], implementation: Netscape
- Variation -TLS: transport layer security, RFC 2246
- Provides
  - *confidentiality*
  - *integrity*
  - *authentication*

- Original goals:
  - Web e-commerce transactions
  - encryption (especially credit-card numbers)
  - Web-server authentication
  - optional client authentication
  - minimum hassle in doing business with new merchant
- Available to all TCP applications
  - secure socket interface

SSL cipher suite

- Cipher suite
  - public-key algorithm
  - symmetric encryption algorithm
  - MAC algorithm
- SSL supports several cipher suites
- Negotiation: client, server agree on cipher suite
  - client offers choice
  - server picks one

common SSL symmetric ciphers

- DES – Data Encryption Standard: block
- 3DES – Triple strength: block
- RC2 – Rivest Cipher 2: block
- RC4 – Rivest Cipher 4: stream

SSL Public key encryption

- RSA

Topic of the lecture

- What is network security?
- Principles of cryptography
- Message integrity, authentication
- Securing e-mail
- Securing TCP connections: SSL
- Network layer security: IPsec
- Securing wireless LANs
- Operational security: firewalls and IDS

What is network-layer confidentiality ?

Between two network entities:

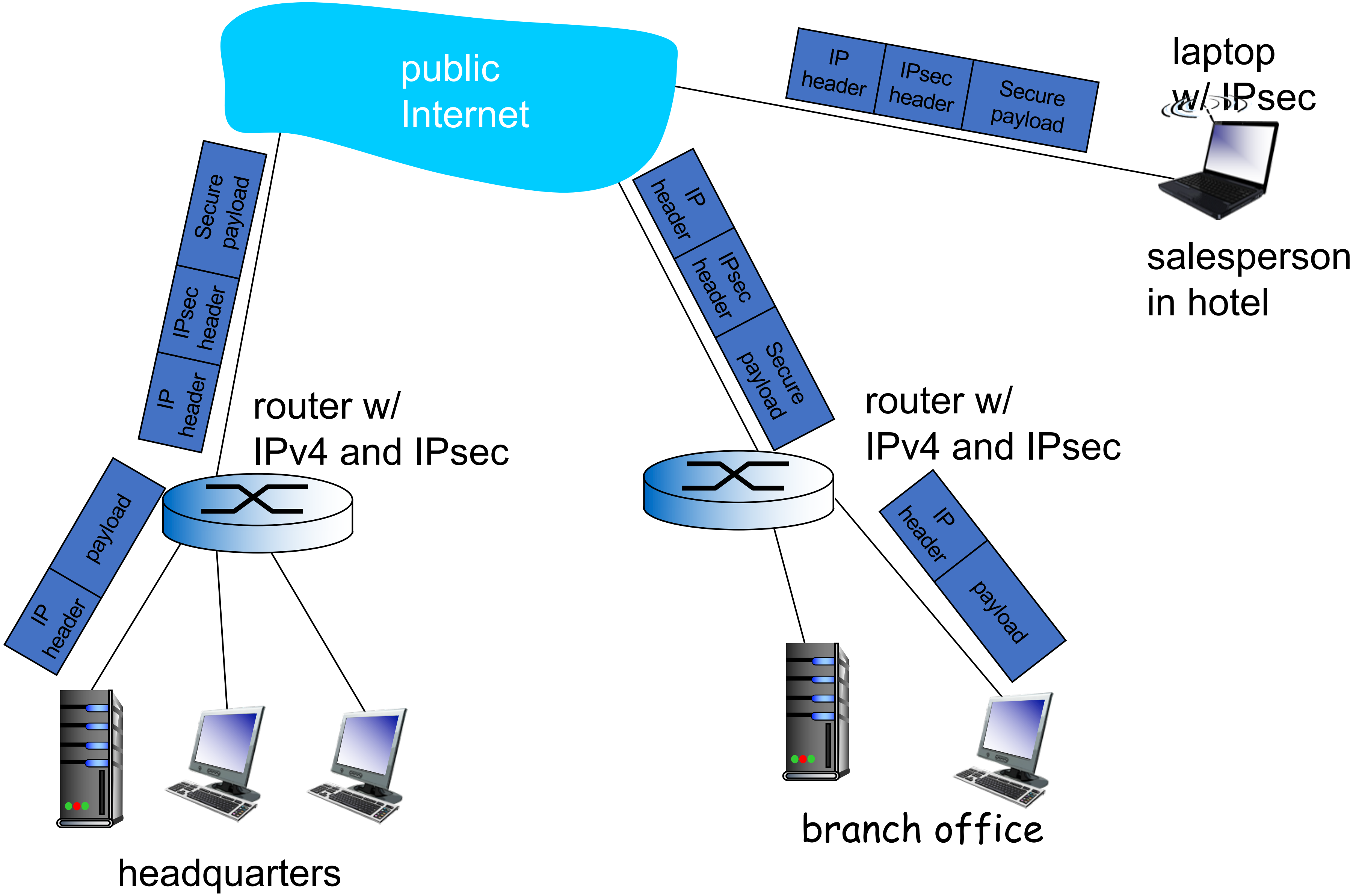
- Sending entity encrypts datagram payload, payload could be:
  - TCP or UDP segment, ICMP message, OSPF message ….
- All data sent from one entity to other would be hidden:
  - web pages, e-mail, P2P file transfers, TCP SYN packets …
- "Blanket coverage"

Virtual Private Networks (VPNs)

Motivation:

- institutions often want private networks for security.
  - costly: separate routers, links, DNS infrastructure.
- VPN: institution's inter-office traffic is sent over public Internet instead
  - encrypted before entering public Internet
  - logically separate from other traffic

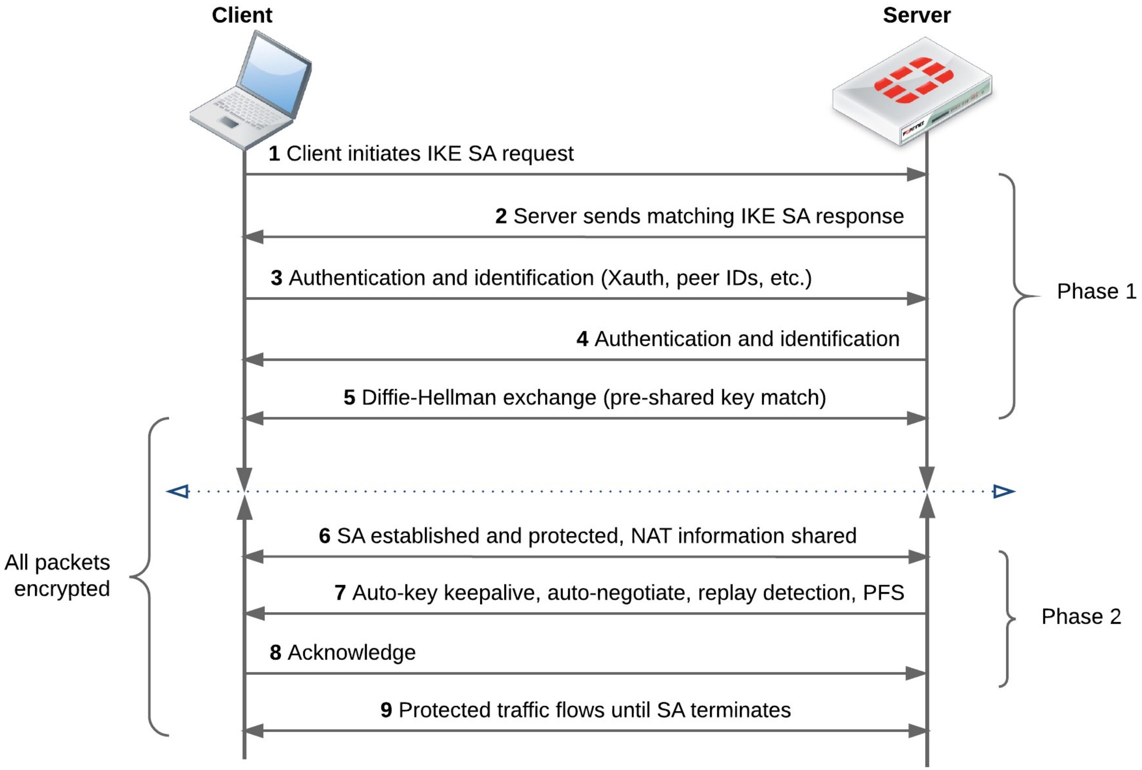
Virtual Private Networks (VPNs)



IPsec services

- Data integrity
- Origin authentication
- Replay attack prevention
- Confidentiality

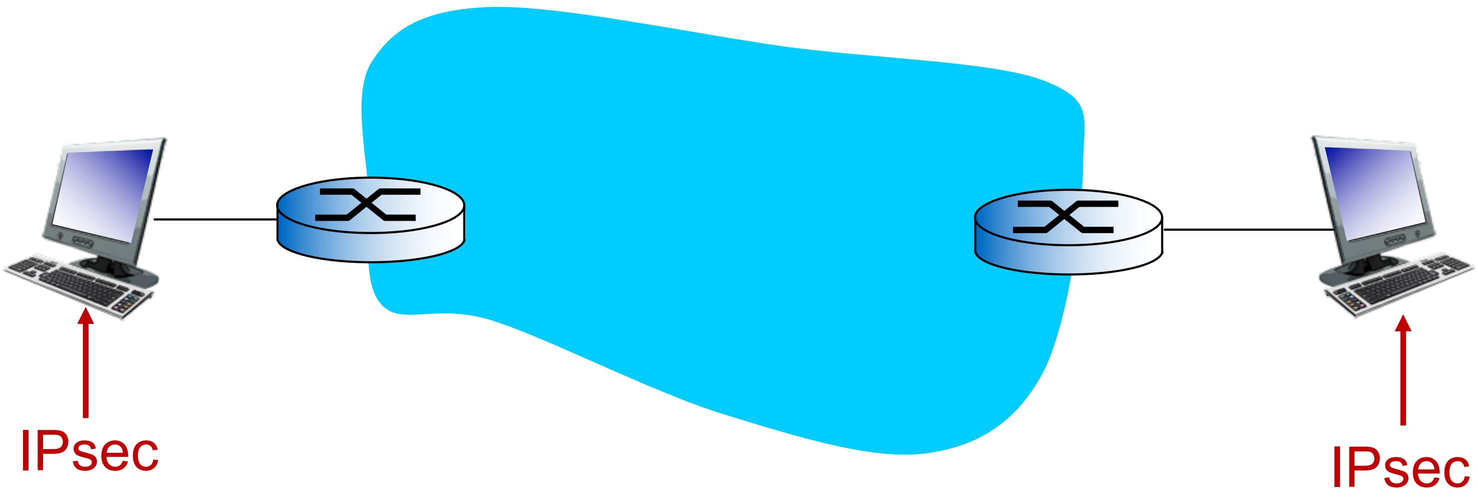
- Two protocols providing different service models:
  - Authentication Header (AH) protocol
    - provides source authentication & data integrity but *not* confidentiality
  - Encapsulation Security Protocol (ESP)
    - provides source authentication, data integrity, *and confidentiality*
    - more widely used than AH



Source: https://help.fortinet.com/fos50hlp/54/Content/FortiOS/fortigate-ipsecvpn-54/IPsec_VPN_Concepts/IKE_Packet_Processing.htm

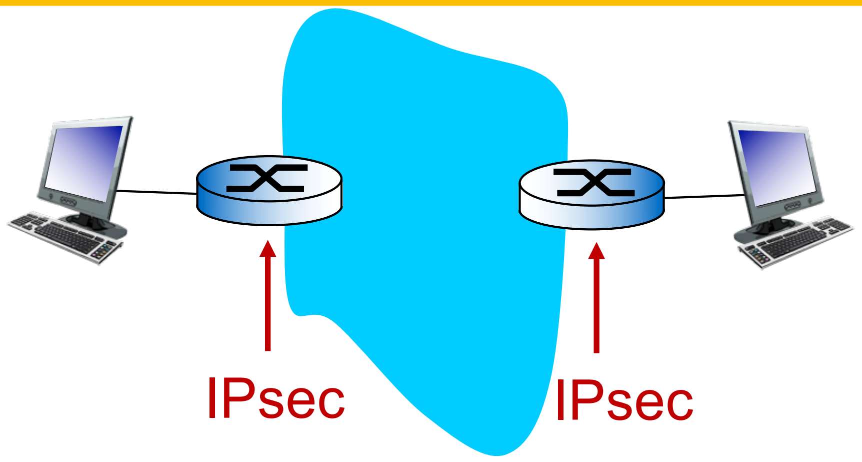
Internet Key Exchange (IKE)
Security Association (SA)

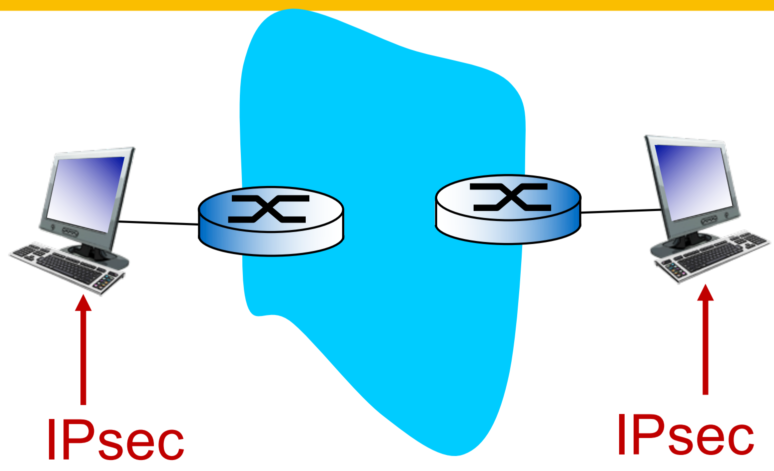
IPsec transport mode



- IPsec datagram emitted and received by end-system
- protects upper level protocols

IPsec – tunneling mode





- edge routers IPsec-aware

- hosts IPsec-aware

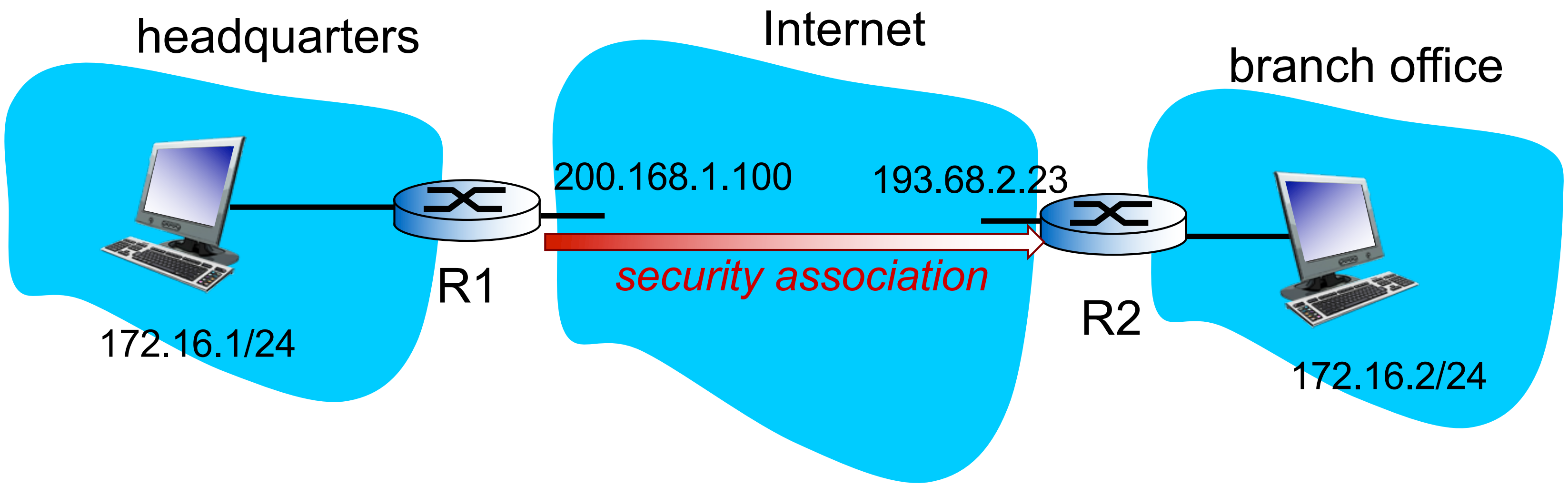
Host mode with AH	Host mode with ESP
Tunnel mode with AH	Tunnel mode with ESP

most common and
most important

Security Associations (SAs)

- Before sending data, "security association (SA)" established from sending to receiving entity
  - SAs are simplex: for only one direction
- Ending, receiving entitles maintain *state information* about SA
  - recall: TCP endpoints also maintain state info
  - IP is connectionless; IPsec is connection-oriented!
- How many SAs in VPN w/ headquarters, branch office, and n traveling salespeople?

Example SA from R1 to R2



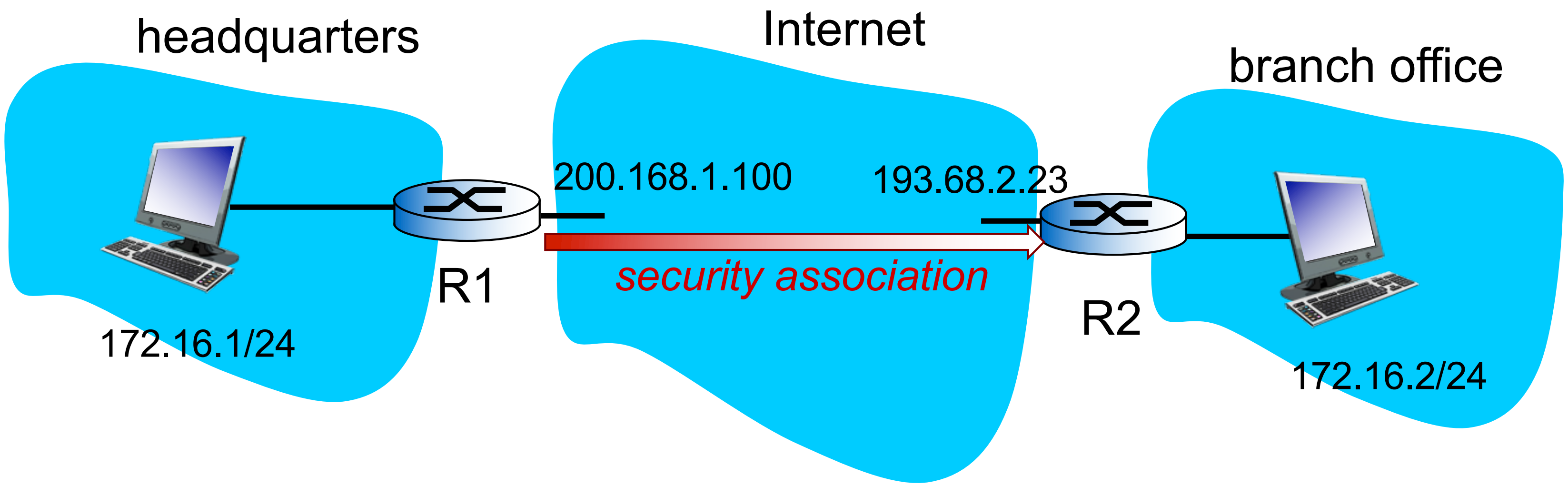
R1 stores for SA:

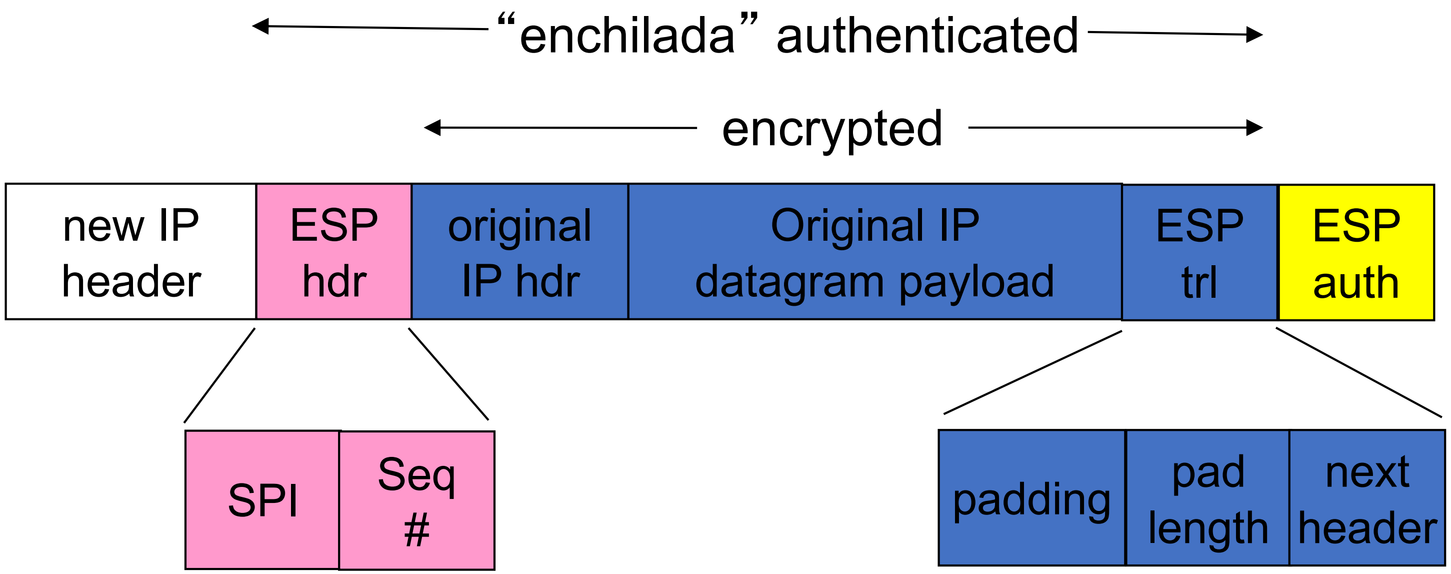
- 32-bit SA identifier: *Security Parameter Index (SPI)*
- origin SA interface (200.168.1.100)
- destination SA interface (193.68.2.23)
- type of encryption used (e.g., 3DES with CBC)
- encryption key
- type of integrity check used (e.g., HMAC with MD5)
- authentication key

Security Association Database (SAD)

- endpoint holds SA state in *security association database (SAD)*, where it can locate them during processing.
- with n salespersons, 2 + 2n SAs in R1's SAD
- when sending IPsec datagram, R1 accesses SAD to determine how to process datagram.
- when IPsec datagram arrives to R2, R2 examines SPI in IPsec datagram, indexes SAD with SPI, and processes datagram accordingly.

What happens?





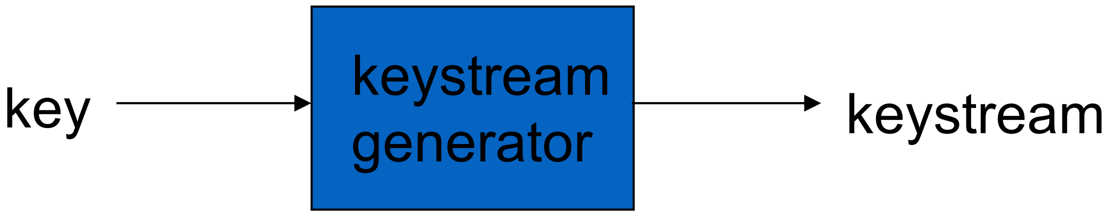
Topic of the lecture

- What is network security?
- Principles of cryptography
- Message integrity, authentication
- Securing e-mail
- Securing TCP connections: SSL
- Network layer security: IPsec
- Securing wireless LANs
- Operational security: firewalls and IDS

WEP design goals

- Symmetric key crypto
  - confidentiality
  - end host authorization
  - data integrity
- self-synchronizing: each packet separately encrypted
  - given encrypted packet and key, can decrypt; can continue to decrypt packets when preceding packet was lost (unlike Cipher Block Chaining (CBC) in block ciphers)
- Efficient
  - implementable in hardware or software

Review: symmetric stream ciphers



- *combine each byte of keystream with byte of plaintext to get ciphertext:*
  - m(i) = ith unit of message
  - ks(i) = ith unit of keystream
  - c(i) = ith unit of ciphertext
  - $c(i) = ks(i) \oplus m(i)$ ($\oplus$ = exclusive or)
  - $m(i) = ks(i) \oplus c(i)$
- WEP uses RC4

Cryptographic Algorithms and Protocols

- Cryptographic algorithms
  - Algorithm executed by a single entity
  - Algorithms performing cryptographic functions
  - Encryption, Hash, digital signature, etc…

- Cryptographic protocols
  - Protocols executed between multiple entities through pre-defined steps of communication performing security-related functions
  - Perform more complicated functions than what the primitive algorithms can provide
  - Primitives: Key agreement, secret sharing, blind signature, coin toss, secure multiparty computations, etc …
  - Complex application protocols: e-commerce, e-voting, e-auction, etc …

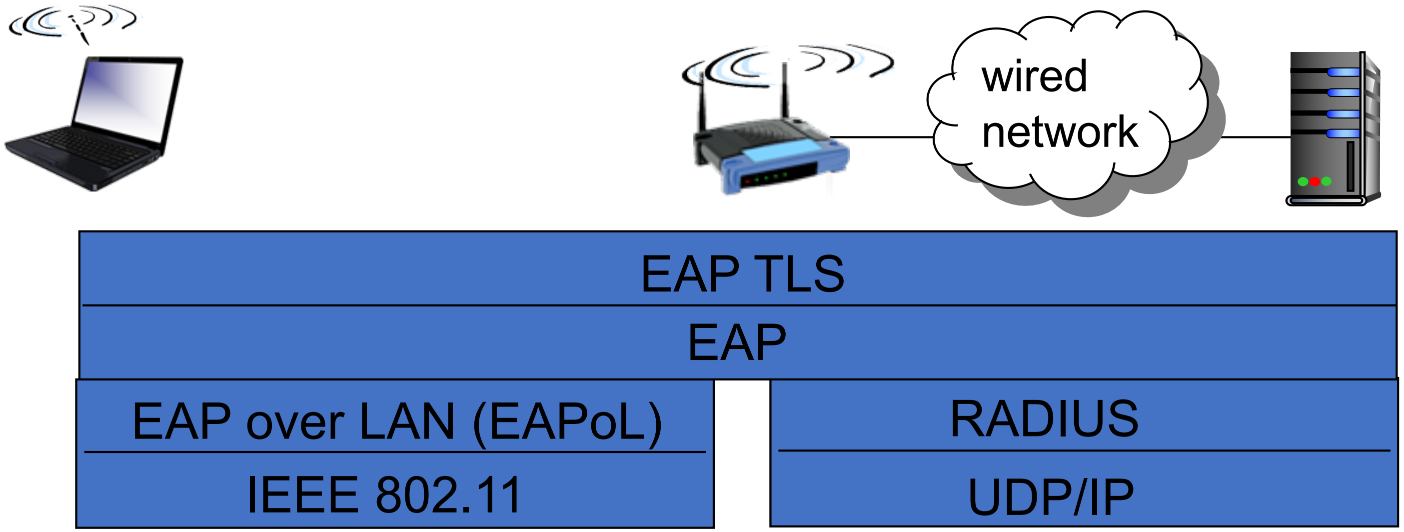
Protocol Primitives

- Zero-knowledge Proofs
  - An interactive method for one party to prove to another that a (usually mathematical) statement is true, without revealing anything other than the validity of the statement.

- Identification, Authentication
  - Over the communication network, one party, Alice, shows to another party, Bob, that she is the real Alice.
  - Allows one party, Alice, to prove to another party, Bob, that she possesses secret information without revealing to Bob what that secret information is.

EAP: extensible authentication protocol

- EAP: end-end client (mobile) to authentication server protocol
- EAP sent over separate "links"
  - mobile-to-AP (EAP over LAN)
  - AP to authentication server (RADIUS over UDP)



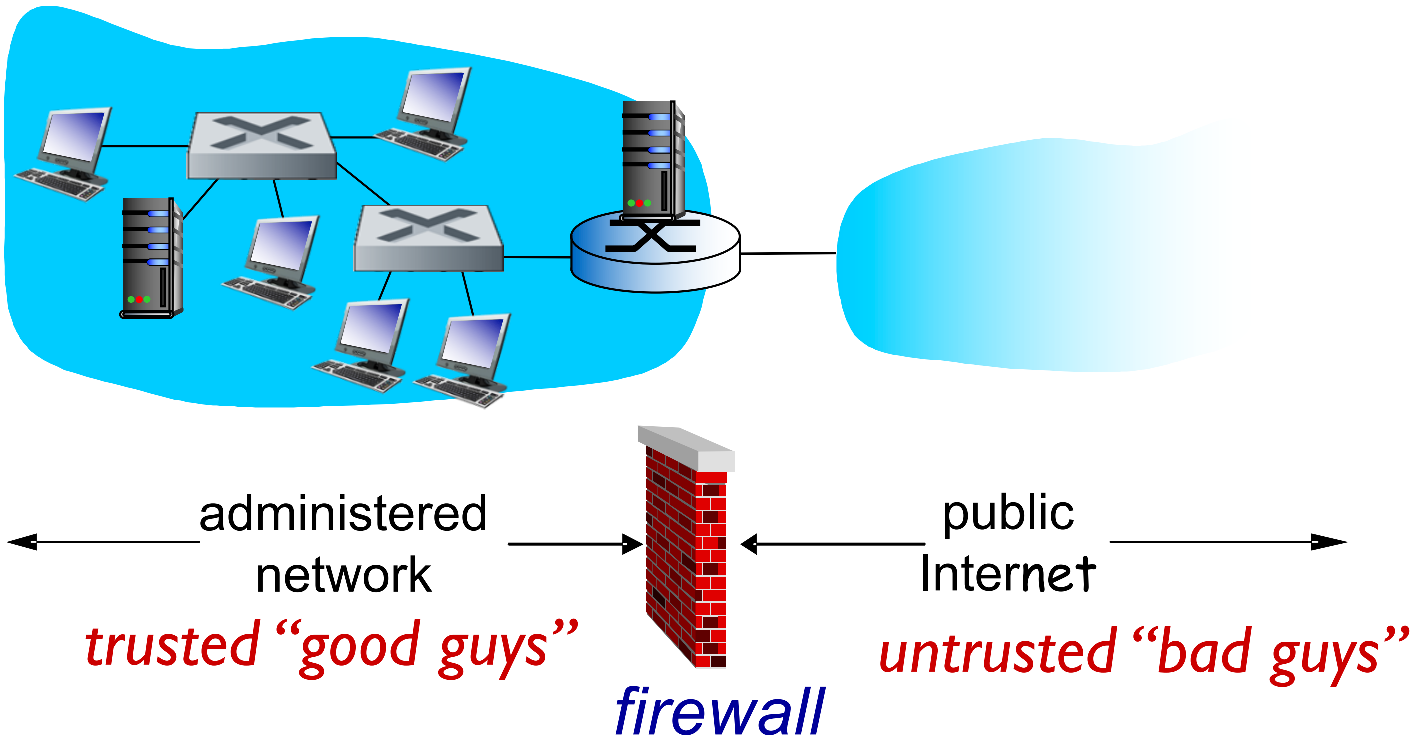
Topic of the lecture

- What is network security?
- Principles of cryptography
- Message integrity, authentication
- Securing e-mail
- Securing TCP connections: SSL
- Network layer security: IPsec
- Securing wireless LANs
- Operational security: firewalls and IDS

Firewalls

firewall

isolates organization's internal net from larger Internet, allowing some packets to pass, blocking others



Firewalls: why

prevent denial of service attacks:

- SYN flooding: attacker establishes many bogus TCP connections, no resources left for "real" connections

prevent illegal modification/access of internal data

- e.g., attacker replaces CIA's homepage with something else

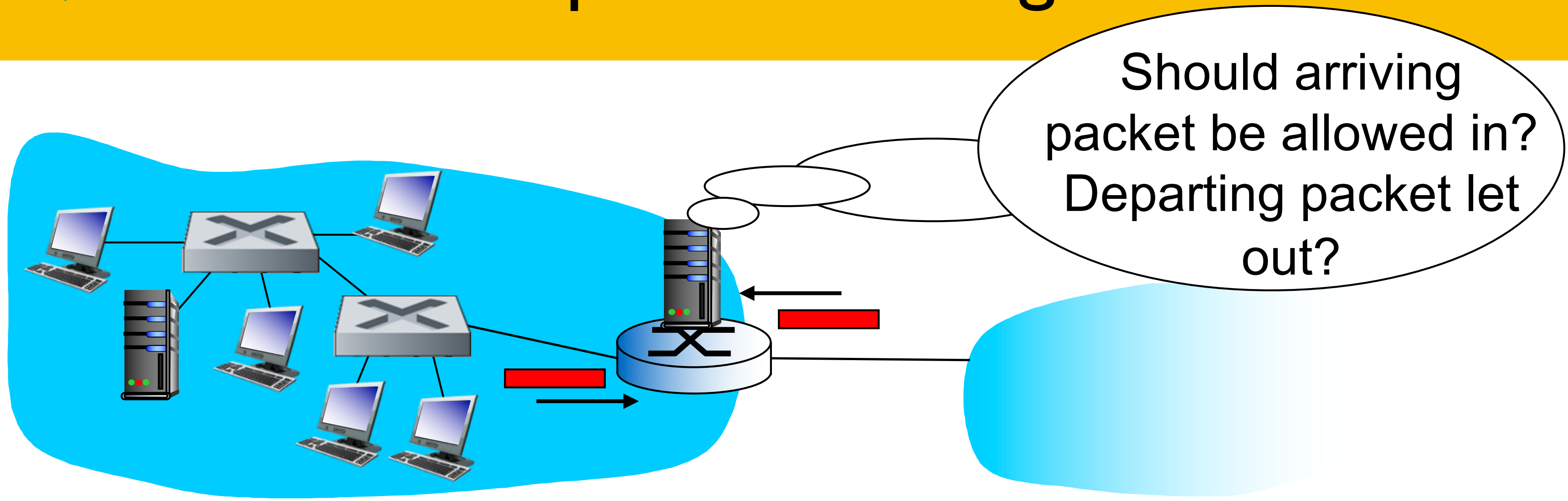
allow only authorized access to inside network

- set of authenticated users/hosts

three types of firewalls:

- stateless packet filters
- stateful packet filters
- application gateways

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

Innopolis University

Stateless packet filtering: example

- *Example 1:* block incoming and outgoing datagrams with IP protocol field = 17 and with either source or dest port = 23
  - *result:* all incoming, outgoing UDP flows and telnet connections are blocked
- *Example 2:* block inbound TCP segments with ACK=0.
  - *result:* prevents external clients from making TCP connections with internal clients, but allows internal clients to connect to outside.

Stateless packet filtering: more examples

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 (e.g. 130.207.255.255).
Prevent your network from being tracerouted	Drop all outgoing ICMP TTL expired traffic

Access Control Lists

- *ACL:* table of rules, applied top to bottom to incoming packets: (action, condition) pairs

action	source address	dest address	protocol	source port	dest port	flag bit
allow	222.22/16	outside of 222.22/16	TCP	> 1023	80	any
allow	outside of 222.22/16	222.22/16	TCP	80	> 1023	ACK
allow	222.22/16	outside of 222.22/16	UDP	> 1023	53	---
allow	outside of 222.22/16	222.22/16	UDP	53	> 1023	----
deny	all	all	all	all	all	all

Stateful packet filtering

- *stateless packet filter:* heavy handed tool
  - admits packets that "make no sense," e.g., dest port = 80, ACK bit set, even though no TCP connection established:

action	source address	dest address	protocol	source port	dest port	flag bit
allow	outside of 222.22/16	222.22/16	TCP	80	> 1023	ACK

- *stateful packet filter:* track status of every TCP connection
  - track connection setup (SYN), teardown (FIN): determine whether incoming, outgoing packets "makes sense"
  - timeout inactive connections at firewall: no longer admit packets

Stateful packet filtering

- ACL augmented to indicate need to check connection state table before admitting packet

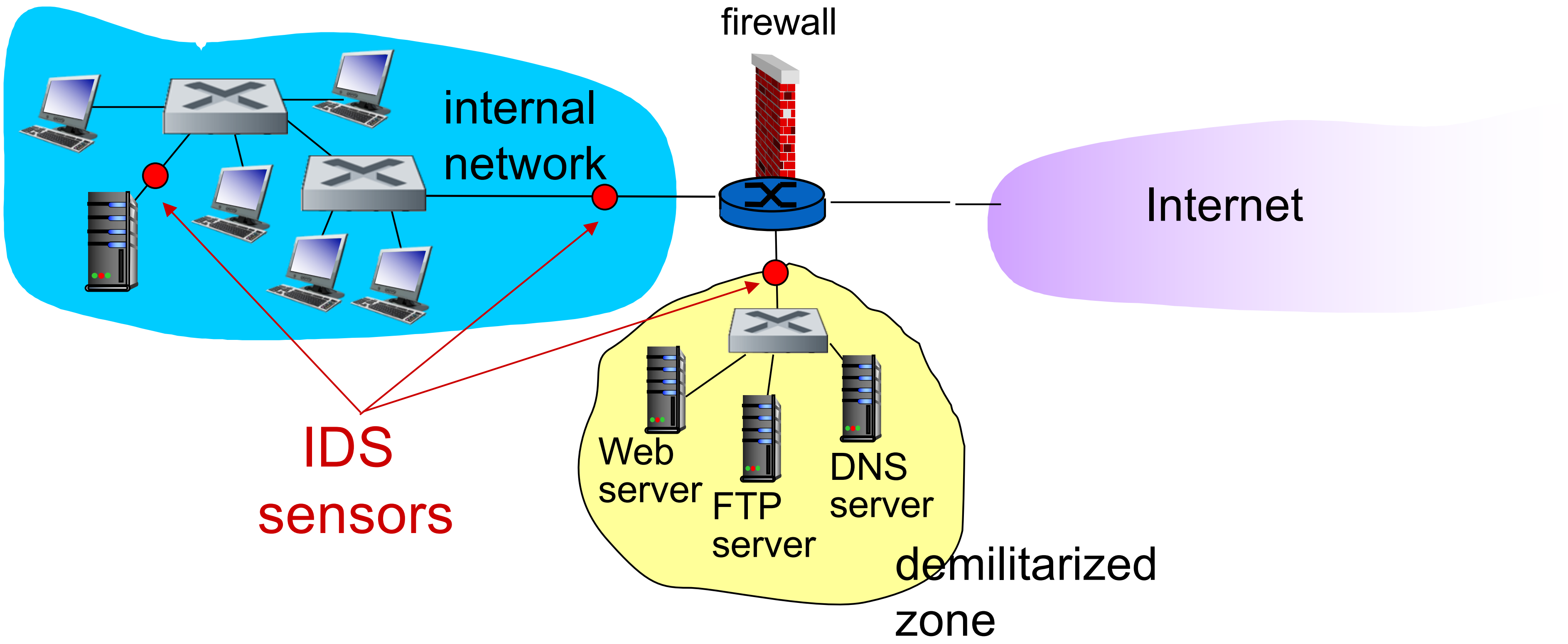
action	source address	dest address	proto	source port	dest port	flag bit	check conxion
allow	222.22/16	outside of 222.22/16	TCP	> 1023	80	any	
allow	outside of 222.22/16	222.22/16	TCP	80	> 1023	ACK	X
allow	222.22/16	outside of 222.22/16	UDP	> 1023	53	---	
allow	outside of 222.22/16	222.22/16	UDP	53	> 1023	----	X
deny	all	all	all	all	all	all	

Intrusion detection systems

- packet filtering:
  - operates on TCP/IP headers only
  - no correlation check among sessions
- *IDS: intrusion detection system*
  - *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

Intrusion detection systems

- multiple IDSs: different types of checking at different locations



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)
- IP sec
- 802.11

operational security: firewalls and IDS