Computer Network

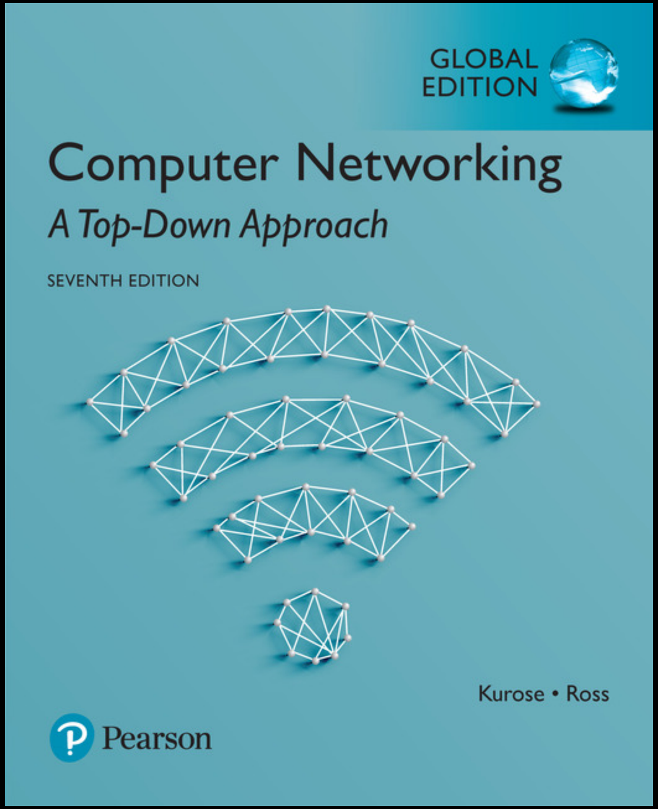
Network Security

School of Electric and Computer Engineering

Pusan National University, KOREA

Younghwan Yoo

Textbook



Computer Networking

A Top–Down Approach

7th edition

Jim Kurose, Keith Ross

Pearson

April 2016

Contents

Computer Network introduction

01. Network Security

02. Cryptography Principles

03. Message Integrity

04. End-Point Authentication

Contents

Computer Network introduction

05. Securing E-mail

06. IPsec and VPNs

07. Wi-Fi Security

08. Firewall and IDS/IPS

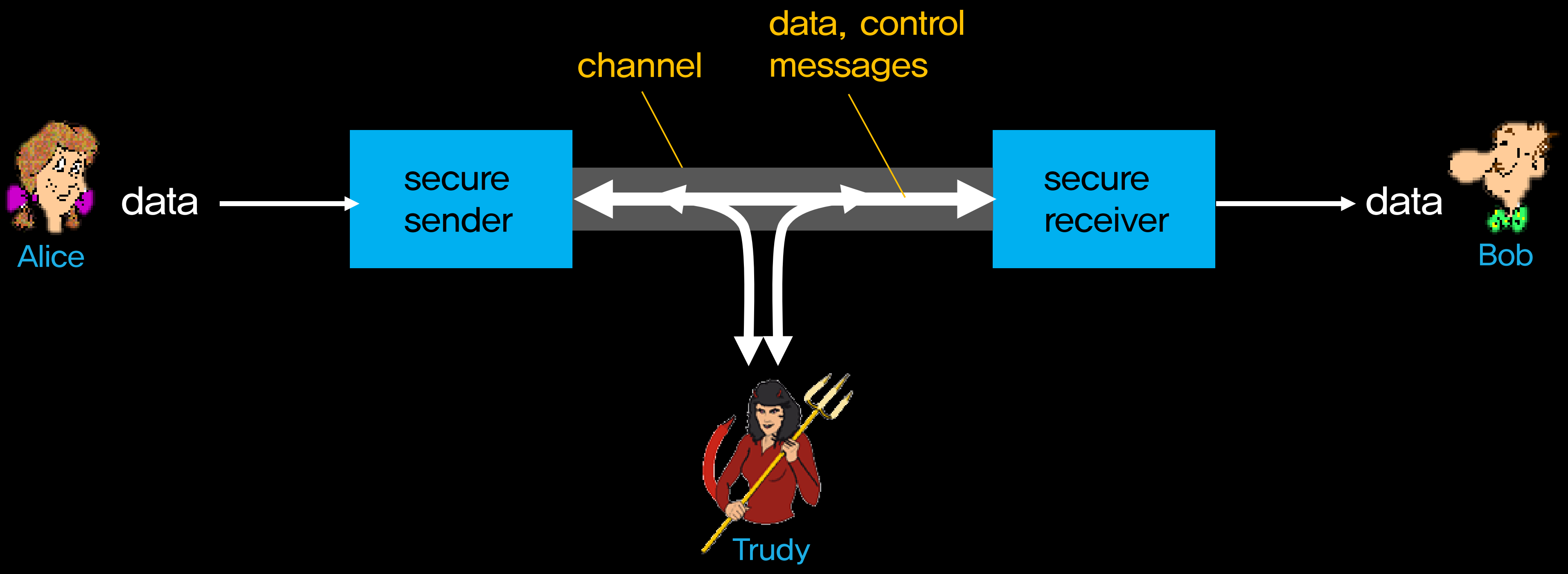
01. Network Security

Properties of Secure Communication

- Confidentiality: only sender, intended receiver should "understand" message contents
  - sender encrypts message
  - receiver decrypts message
- Message integrity: sender, receiver want to ensure message not altered (in transit, or afterwards) without detection
- End-point authentication: sender, receiver want to confirm identity of each other
- Operational security: access to the system and the availability must be controlled to protect the system against network attacks and intrusion

Friends and Enemies: Alice, Bob, Trudy

- Well–known in network security world
- Bob, Alice want to communicate “securely”
- Trudy (intruder) may intercept, delete, or add messages



Important Questions

Who Might Bob, Alice Be?

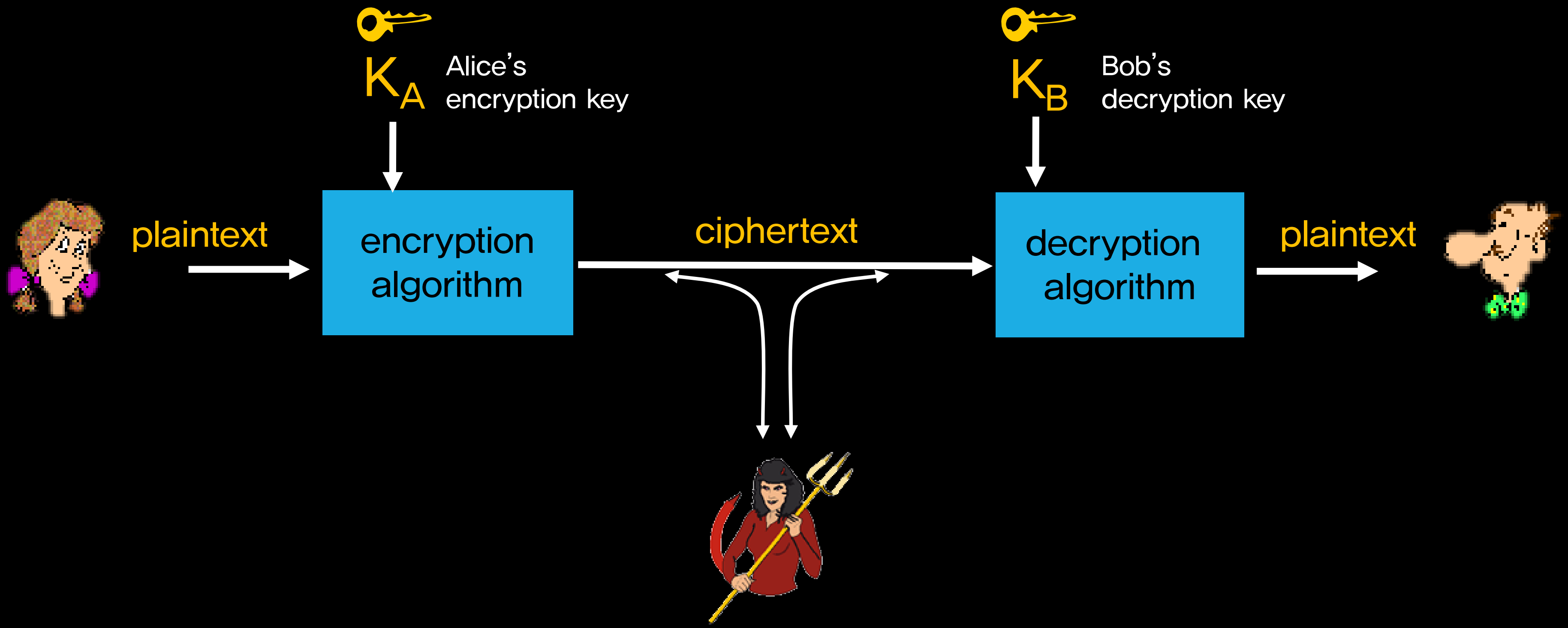
- … well, real-life Bobs and Alices!
- Web browser/server for electronic transactions (e.g., on-line purchases)
- On-line banking client/server
- DNS client/server
- Routers exchanging routing table updates

What can a "bad guy" do?

- Eavesdrop: intercept messages
- Actively insert 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)

02. Cryptography Principles

Cryptography Terminologies



m plaintext message

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

$m = K_B(K_A(m))$

Simple Encryption Scheme

- Substitution cipher: substituting one thing for another
- Mono-alphabetic cipher: substitute one letter for another

```
plaintext:  abcdefghijklmnopqrstuvwxyz

ciphertext: mnbvcxzasdfghjklpoiuytrewq
```

e.g.:

```
 Plaintext: bob. i love you. alice
ciphertext: nkn. s gktc wky. mgsbc
```

Encryption key: mapping from set of 26 letters to set of 26 letters

Types of Cryptography

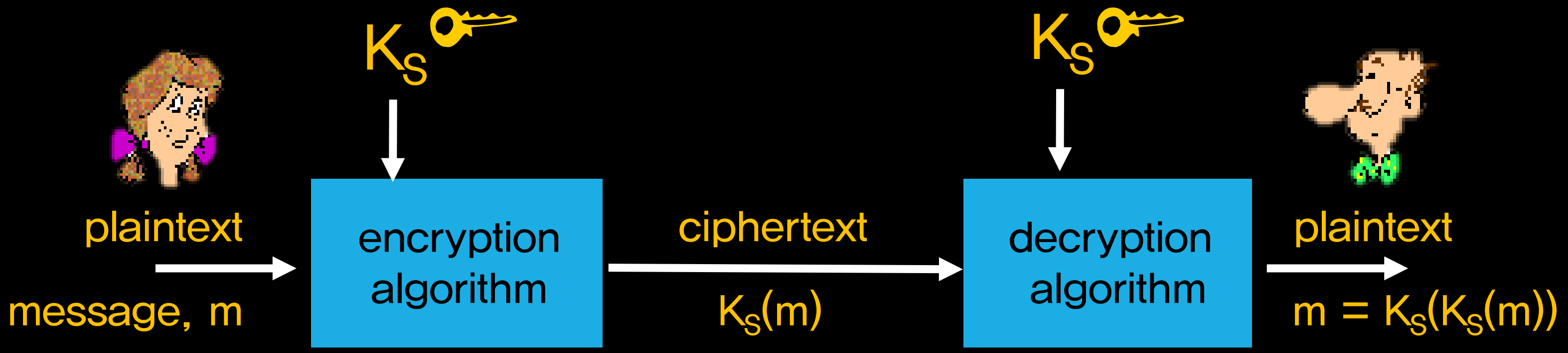
Symmetric key cryptosystem

- the same key is used for encryption and decryption
- the key must be kept secret
- secret key system

Asymmetric key cryptosystem

- different keys are used for encryption and decryption
- one of the two keys is exposed to other users
- public key system

Symmetric Key Cryptography



- Bob and Alice share same (symmetric) key: K_S
- E.g., key is a known substitution pattern in mono alphabetic substitution cipher
- Q: How do Bob and Alice agree on key value?

Symmetric Key Crypto: DES

DES: Data Encryption Standard

- US encryption standard[1993]
- 56-bit symmetric key,

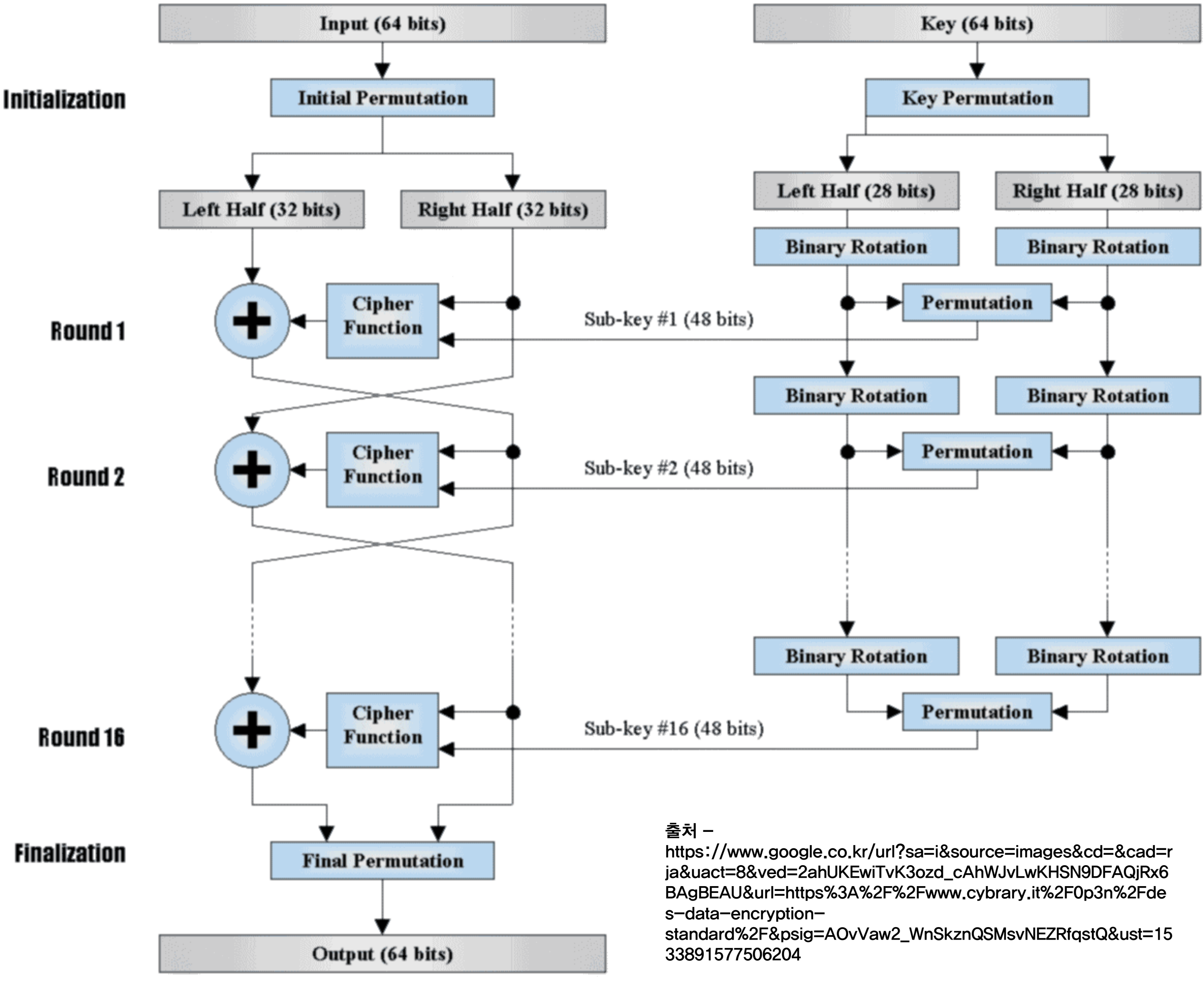
  64-bit plaintext input

DES Operation

① initial permutation

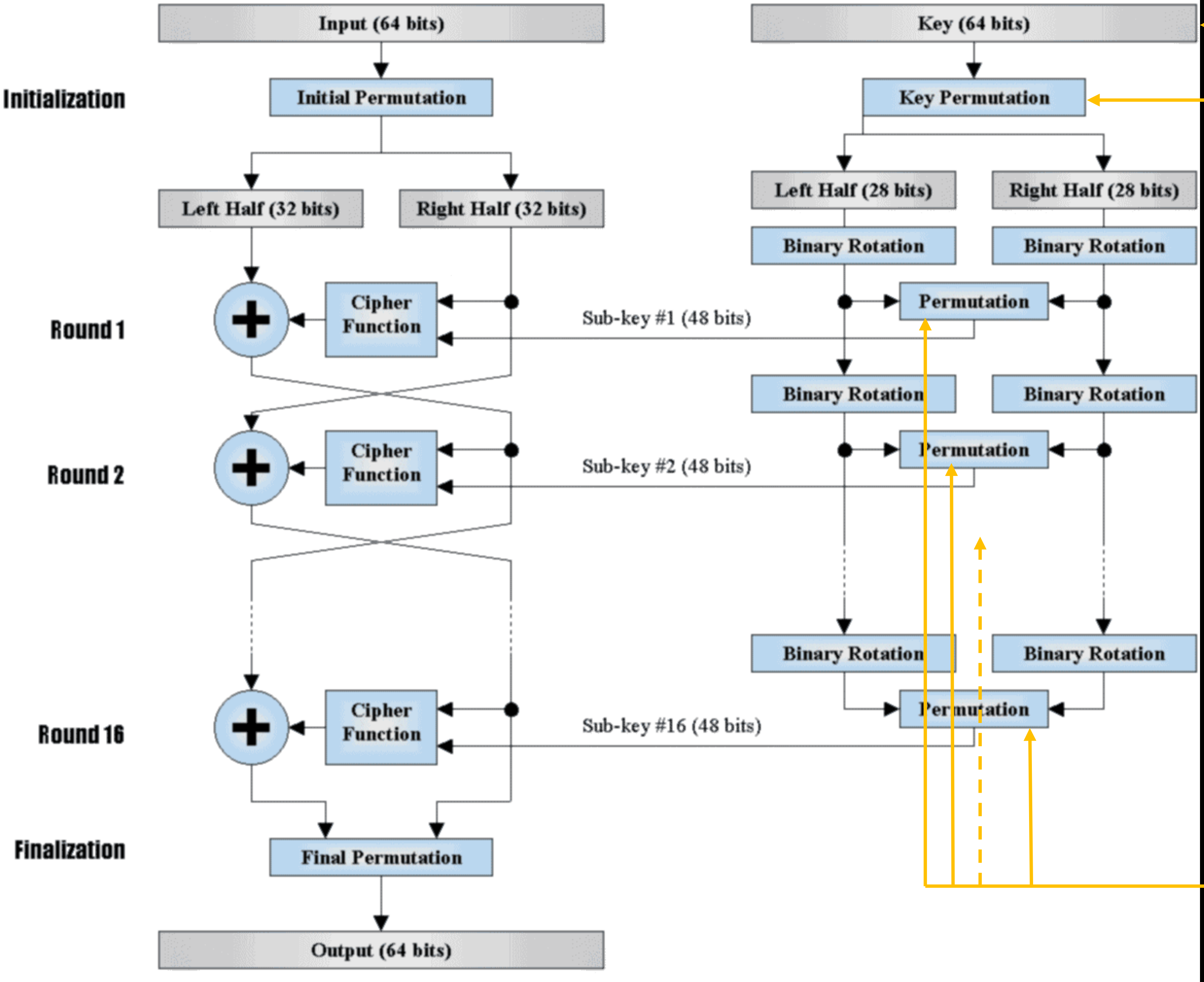
② 16 identical “rounds” of function application, each using different 48 bits of key

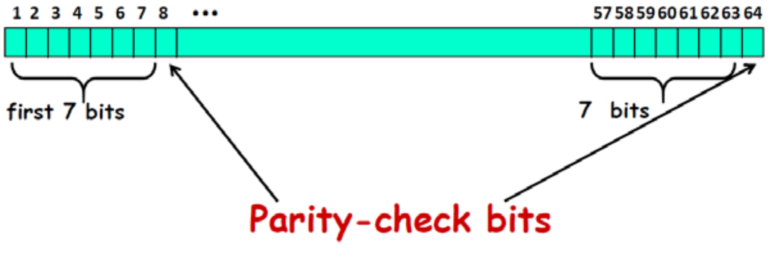
③ final permutation



출처 – https://www.google.co.kr/url?sa=i&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwiTvK3ozd_cAhWJvLwKHSN9DFAQjRx6BAgBEAU&url=https%3A%2F%2Fwww.cybrary.it%2F0p3n%2Fdes-data-encryption-standard%2F&psig=AOvVaw2_WnSkznQSMsvNEZRfqstQ&ust=1533891577506204

Symmetric Key Crypto: DES





Each parity-check bit is the XOR of the previous 7 bits

56 bit key + 8 parity bits

Left						
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
Right						
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

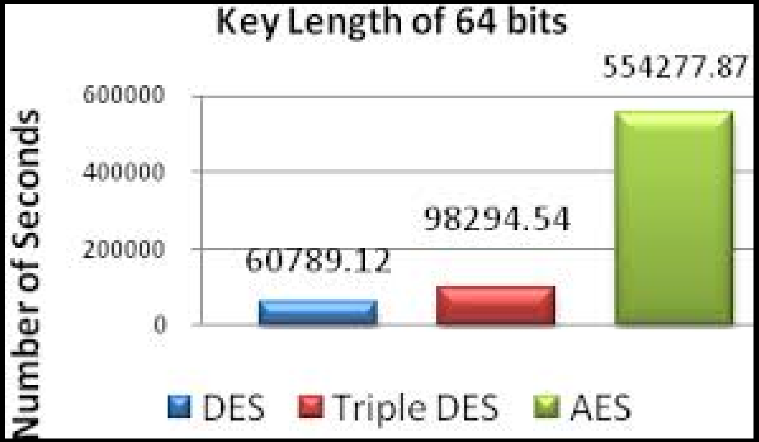
Permuted Choice 1 (PC–1)

14	17	11	24	1	5	3	28
15	6	21	10	23	19	12	4
26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32

Permuted Choice 2 (PC–2)

How Secure is DES?

- DES challenge: 56-bit-key-encrypted phrase can be decrypted in less than a day with the brute force attack
  - No known good analytic attack
- Making DES more secure:
  - 3DES: encrypt 3 times with 3 different keys
- AES (Advanced Encryption Standard)
  - 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 key) taking 1 sec on DES, takes 149 trillion years for AES

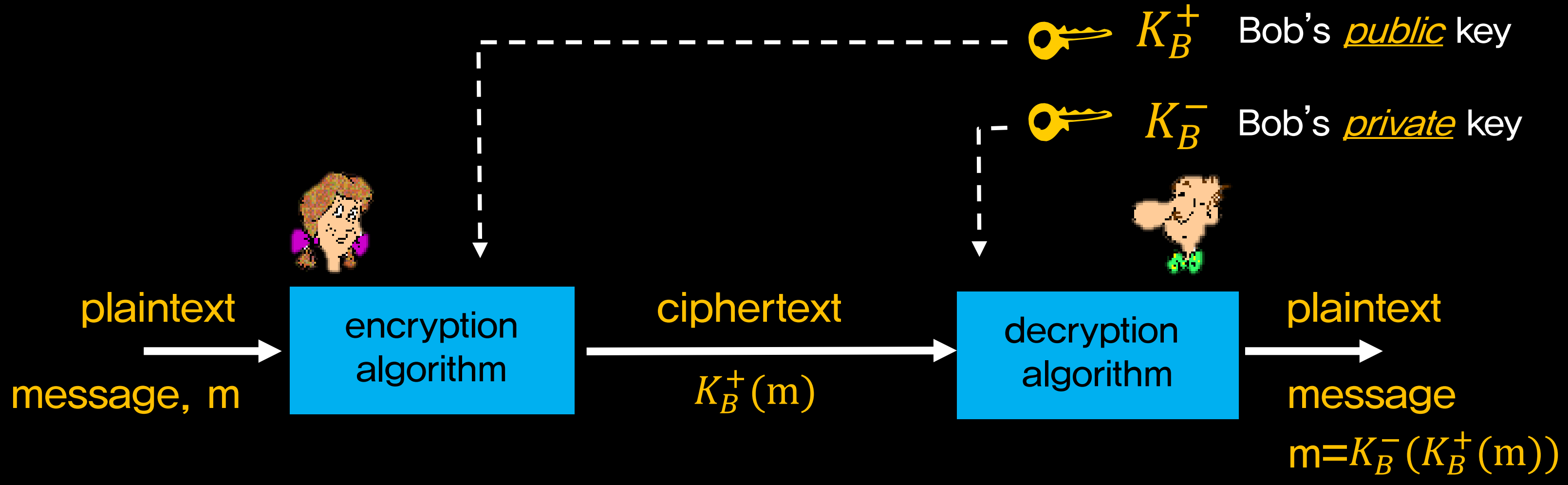


Required time for brute force attack

출처 – https://www.google.co.kr/url?sa=i&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwiR-_7j3-HcAhUHZt4KHSnKCy0QjRx6BAgBEAQ&url=http%3A%2F%2Fpaper.ijcsns.org%2F07_book%2F201001%2F20100139.pdf&psig=AOvVaw0H_dHHm9kZJlZvJJ6t-NbH&ust=1533965084737293

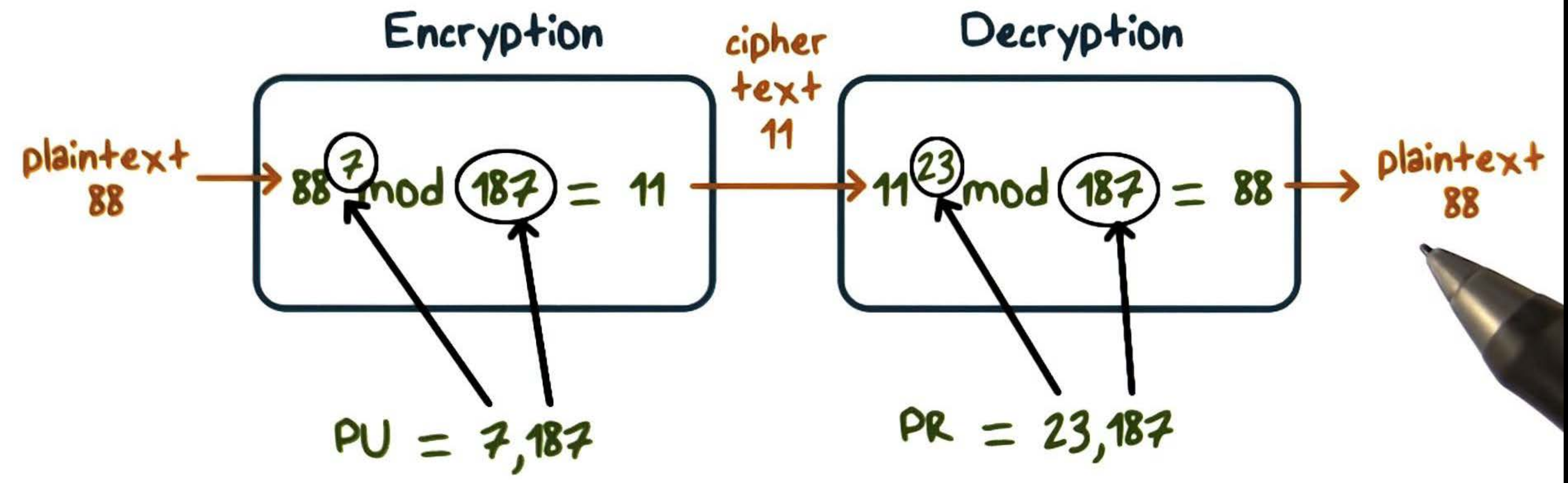
Asymmetric (Public) Key Cryptography

- Challenge of symmetric key cryptography
  - "How to agree on key in first place?" (particularly, if never meet each other?)
- Asymmetric key cryptography
  - sender, receiver do not share a secret key
  - public encryption key known to all
  - private decryption key known only to receiver



Asymmetric Key Crypto: RSA

- Public key encryption requirements
  - need $K_B^+(.)$ and $K_B^-(.)$ such that $$\boxed{K_B^-\left(K_B^+(m)\right) = m}$$
  - given public key K_B^+, it should be impossible to compute private key K_B^-
- RSA (Rivest, Shamir, Adleman) algorithm



출처 – https://www.google.co.kr/url?sa=i&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwiXoof58uHcAhVF_GEKHWwYDmwQjRx6BAgBEAU&url=https%3A%2F%2Fwww.youtube.com%2Fwatch%3Fv%3DkKgp0KdpOhQ&psig=AOvVaw2wngEWT9wW-usYZNyXnvCR&ust=1533970272030513

Asymmetric Key Crypto: RSA

Creating public/private key pair

1. Choose two large prime numbers *p*, *q*. (e.g., 1024 bits each)
2. Compute *n* = *pq*, *z* = (*p*–1)(*q*–1)
3. Choose *e* (with *e* 〈 *n*) that has no common factors with *z* (*e*, *z* are "relatively prime")
4. Choose *d* such that *ed*–1 is exactly divisible by *z* (i.e, *ed* mod *z* = 1)
5. Public key (*n*,*e*) and private key (*n*,*d*)

Encryption and decryption

- Message bit pattern represented by an integer number
- Given (*n*,*e*) and (*n*,*d*),
  - to encrypt message *m* (〈*n*)

$$\boxed{c = m^e \bmod n}$$

  - to decrypt received bit pattern *c*

$$\boxed{m = c^d \bmod n}$$

- Magic happens!

$$\boxed{m = \underbrace{(m^e \bmod n)}_{c}{}^{d} \bmod n}$$

Asymmetric Key Crypto: More on RSA

How secure is RSA?

- Suppose you know Bob's public key (n,e). How hard is it to determine d?
- Essentially need to find factors of n without knowing the two factors p, q
  - factoring a big number is very hard, since there is no easy factoring method yet

RSA in practice: used for exchanging session keys

- Exponentiation in RSA is computationally intensive
  - DES is at least 100 times faster than RSA
- RSA is used to establish a secure connection on which a session key is exchanged
  - the session key is a symmetric key to encode data using DES or AES

03. Message Integrity

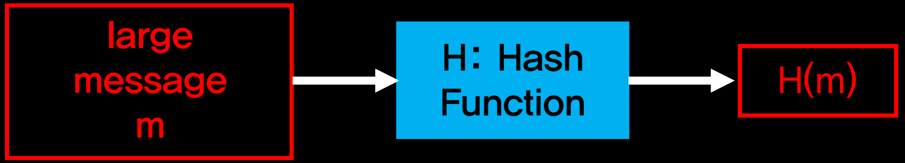
Message Integrity

- Two things for guaranteeing message integrity (or message authentication)
  - the message indeed originated from Alice
  - the message was not tampered with on its way to Bob
- Two methods
  - use of shared secret key: MAC (message authentication code)
  - use of public key mechanism: digital signature

Message Digest

- Computationally expensive to encrypt long messages
- Goal of “message integrity”: not to scramble message contents but rather to guarantee message not to be changed during transmission
- Message digest: fixed–length, easy–to–compute digital “fingerprint”
  - apply hash function H to m, get fixed size message digest, H(m)

- Requirements for hash function
  - many–to–one mapping
  - fixed–size message digest (fingerprint)
  - given message digest x, computationally infeasible to find m such that x = H(m)
  - e.g., MD5 (128–bit), SHA–1 (160–bit)


large message m

H: Hash Function

H(m)

Message Authentication Code (MAC)

- Ver. 1
  1) sender calculates the hash H(m) based on message m
  2) sender creates an extended message (m, H(m)) and sends it
  3) receiver calculates H(m) using m, then checks if it equals the hash received
- Trudy can create a bogus message m', calculate H(m'), and send (m', H(m'))

- Ver. 2: using shared key s called authentication key
  1) sender creates $m + s$ with a secret shared key s and calculates H($m + s$), which is called message authentication code (MAC)
  2) sender creates an extended message (m, H($m + s$)) and sends it
  3) receiver (already knows s) calculates H($m + s$) using m and s, then checks if it equals the hash received

Digital Signature

- Requirements of digital signature
  - given that sender (Bob) digitally signs document to mean he approves it, and sends it
  - recipient (Alice) can prove to someone, in a way that is verifiable and non–forgeable, that no one else (including Alice) but Bob must have signed document

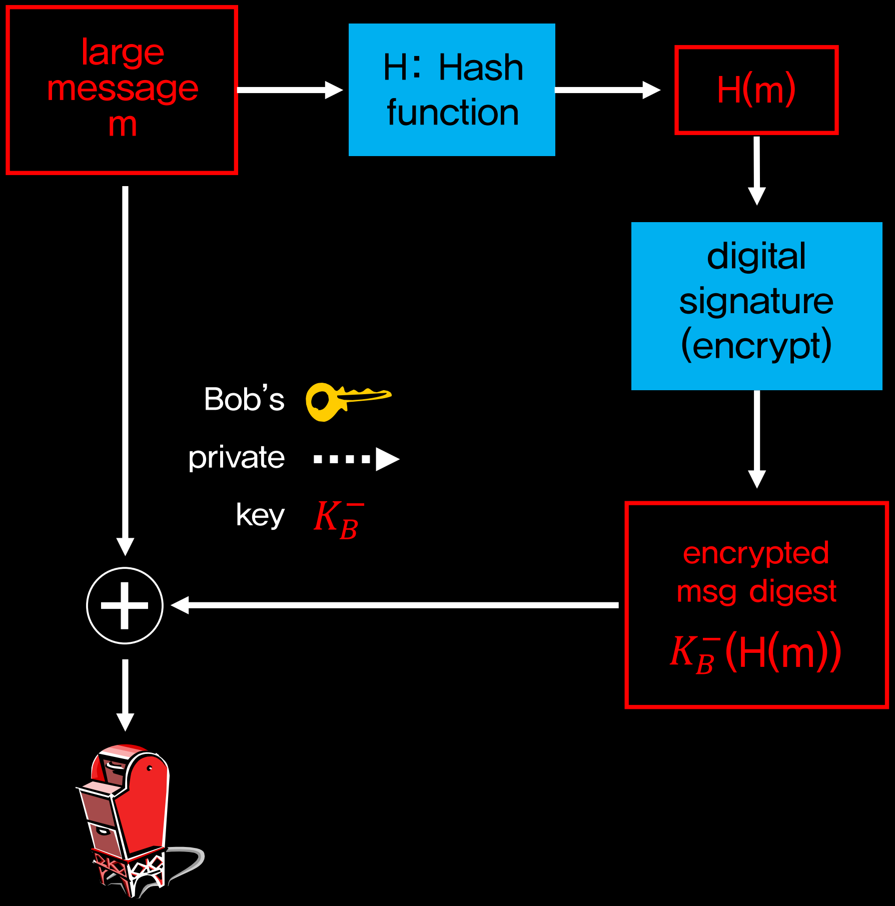
- Property of RSA

$$\underbrace{K_B^-(K_B^+(m))}_{\substack{\text{use public key first,}\\ \text{followed by private key}}} = m = \underbrace{K_B^+(K_B^-(m))}_{\substack{\text{use private key first,}\\ \text{followed by public key}}}$$

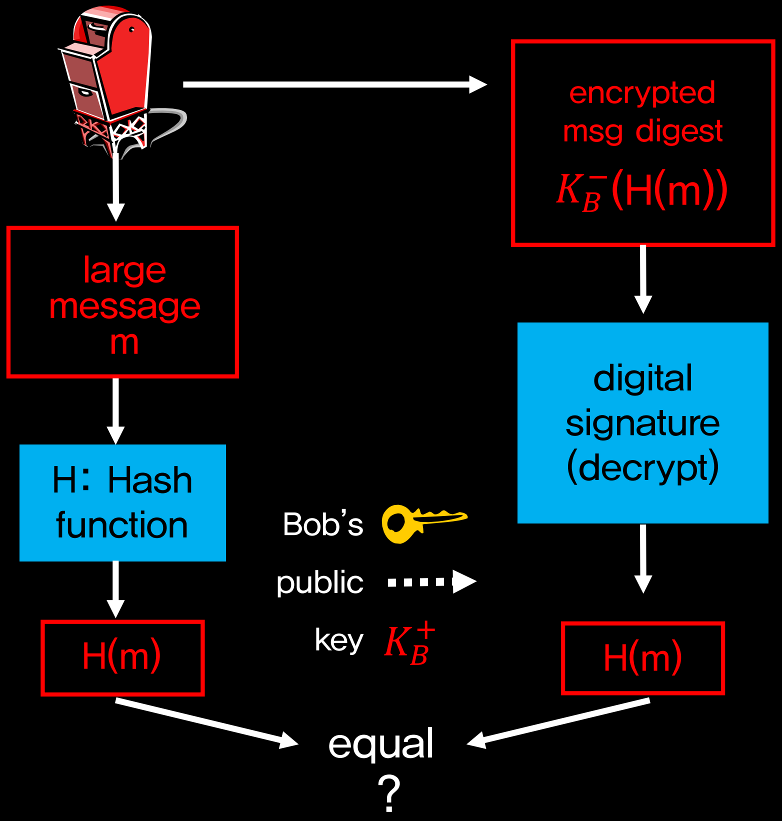
result is the same!

Digital Signature

- Bob sends digitally signed message with his private key:



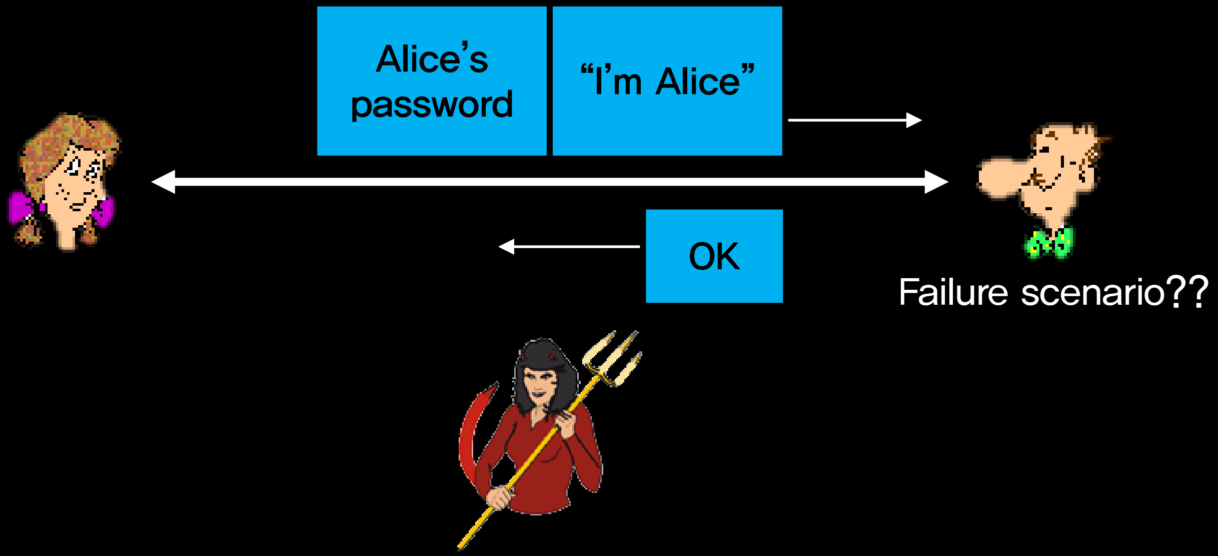
- Alice verifies signature and integrity of signed message with Bob's public key:



04. End-Point Authentication

Authentication

- End-point authentication: the process of one entity proving its identity to another entity over a computer network, e.g., a user proving its identity to an e-mail server
- Simple try: Alice says "I am Alice" and sends her secret password to "prove" it

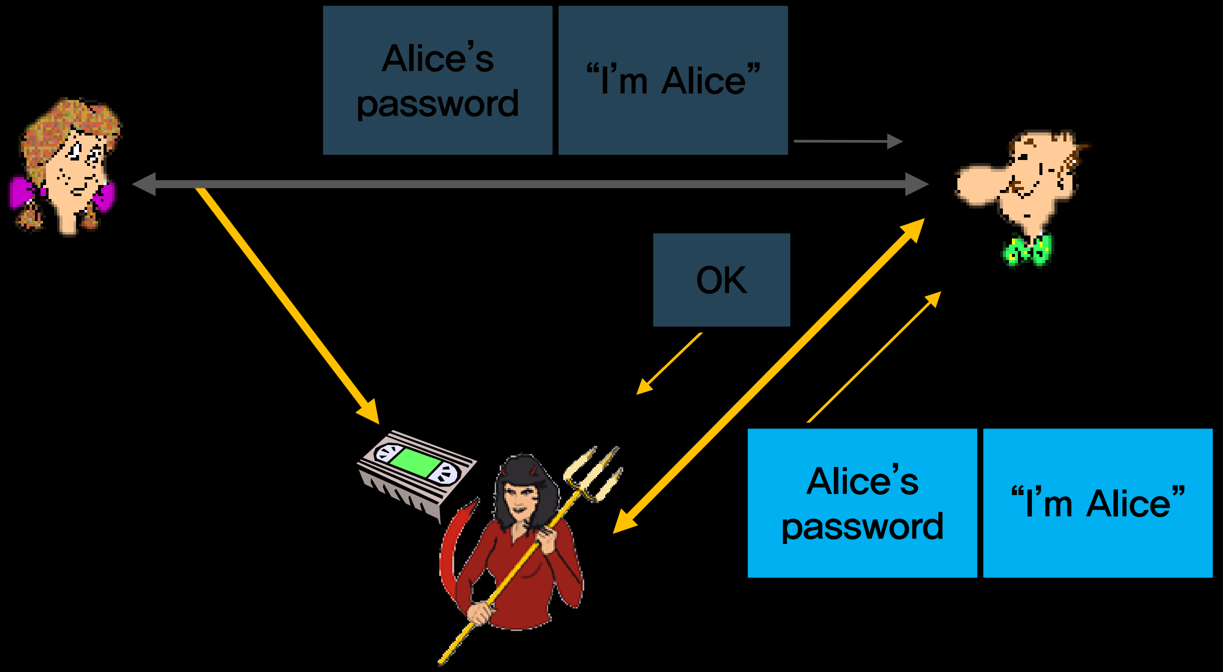


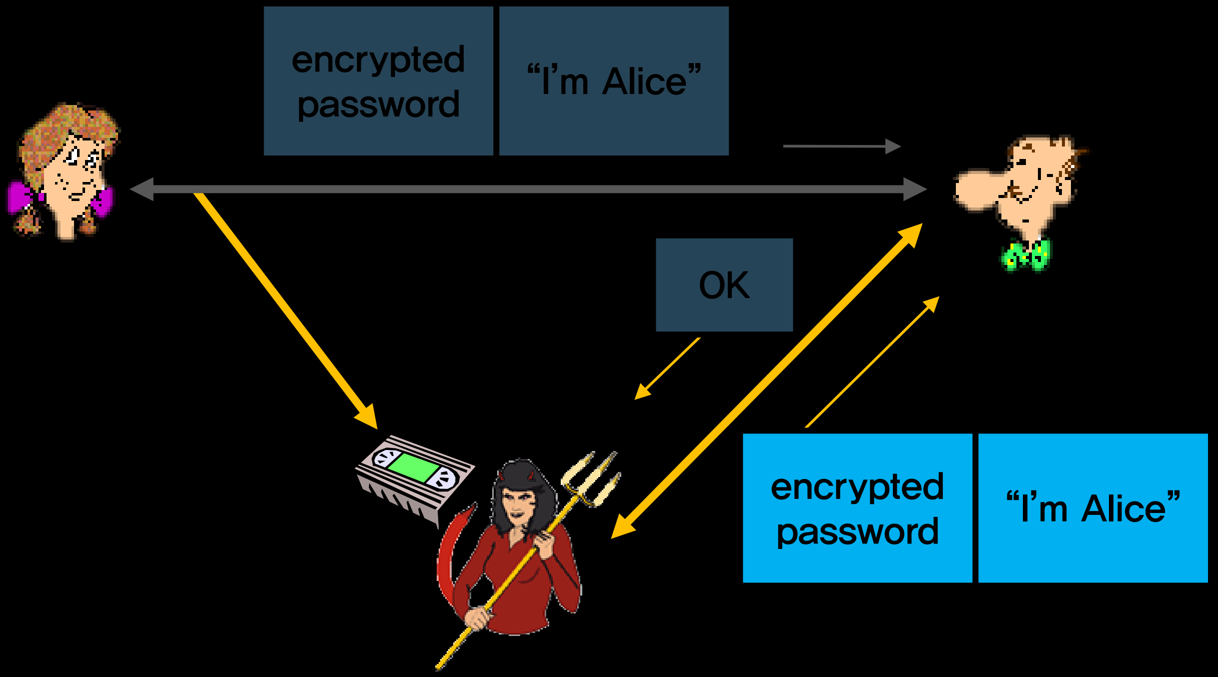
Failure scenario??

Simple Transfer of Password: Failure Scenario

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

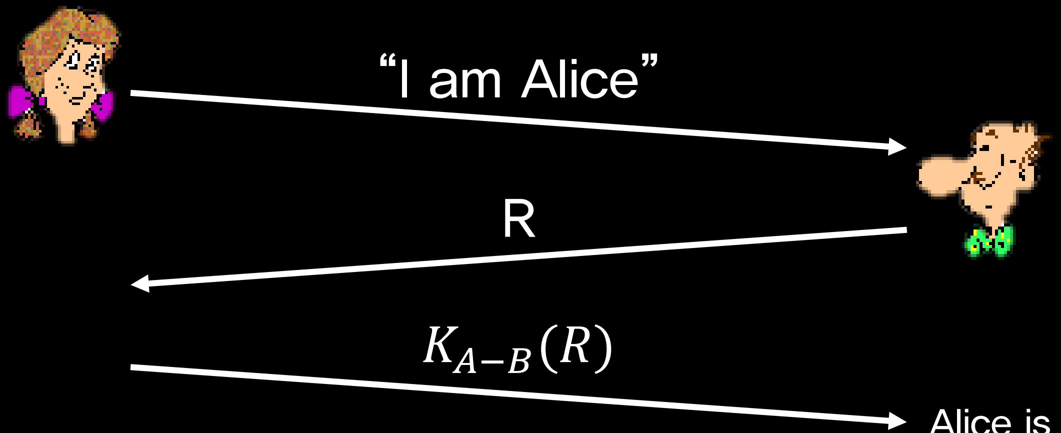
- Although the password is encrypted, the playback attack still works!





First Resolution to Playback Attack

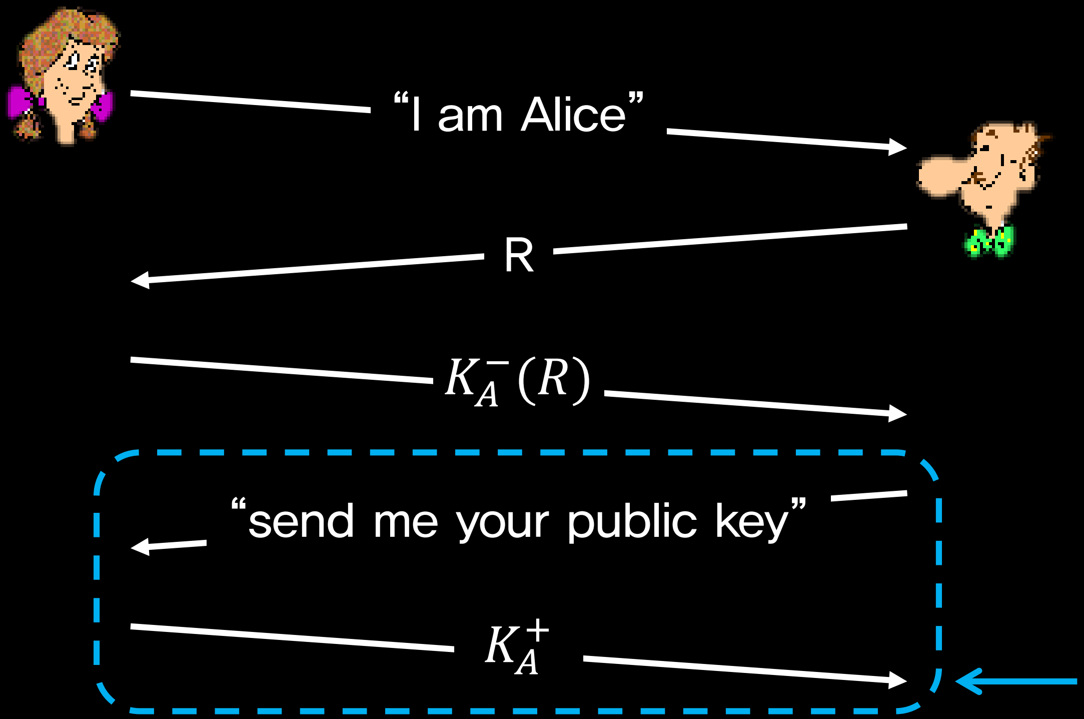
- Nonce: number (R) used only once–in–a–lifetime
- Authentication using nonce and secret key
  - to prove Alice "live", Bob sends Alice nonce, R, then
  - Alice returns R, encrypted with shared secret key



Alice is live, and only Alice knows key to encrypt nonce, so it must be Alice!

Second Resolution to Playback Attack

- The previous method requires shared symmetric key
- Authentication using public key techniques:



Bob checks if

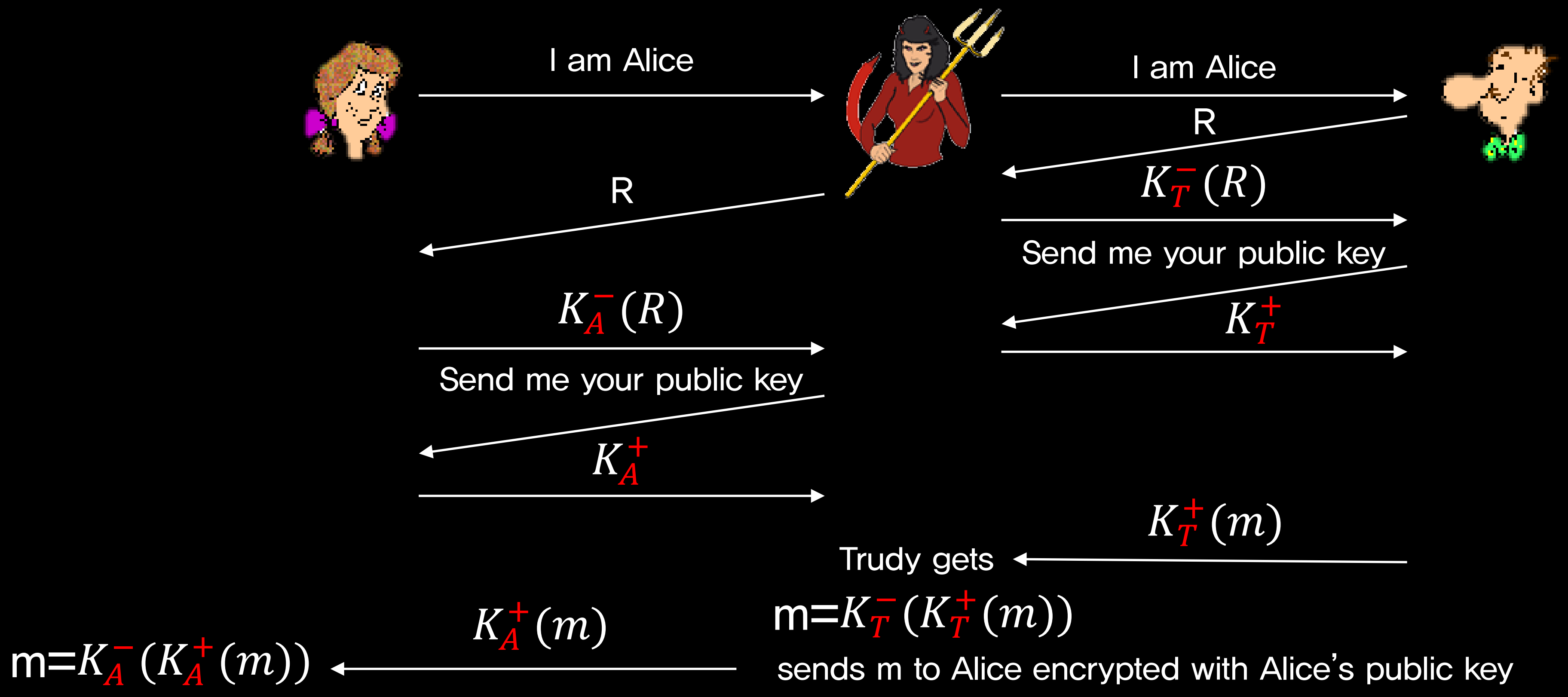
$$K_A^+\left(K_A^-(R)\right) = R$$

If it is, Bob knows that the opposite part is Alice, because only Alice could have the private key

Where can Bob get Alice's public key?

Security Hole

- Man–(or Woman)–in–the–middle–attack: Trudy poses as Alice (to Bob) and as Bob (to Alice)



I am Alice

I am Alice

R

R

$K_T^-(R)$

$K_A^-(R)$

Send me your public key

Send me your public key

K_T^+

K_A^+

$K_T^+(m)$

Trudy gets

$m=K_T^-(K_T^+(m))$

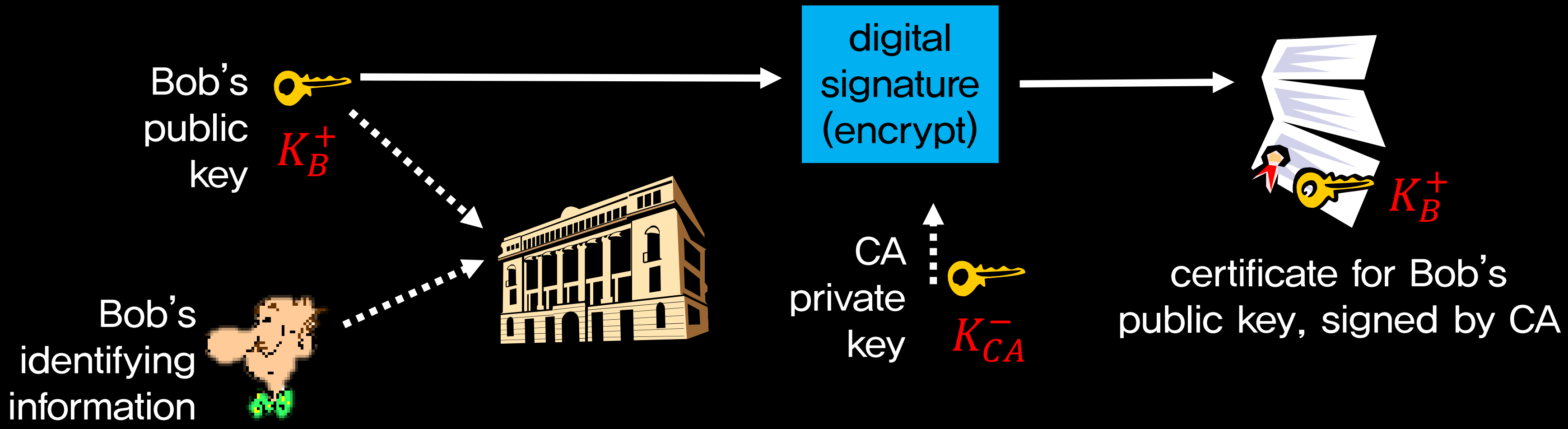
sends m to Alice encrypted with Alice's public key

$K_A^+(m)$

$m=K_A^-(K_A^+(m))$

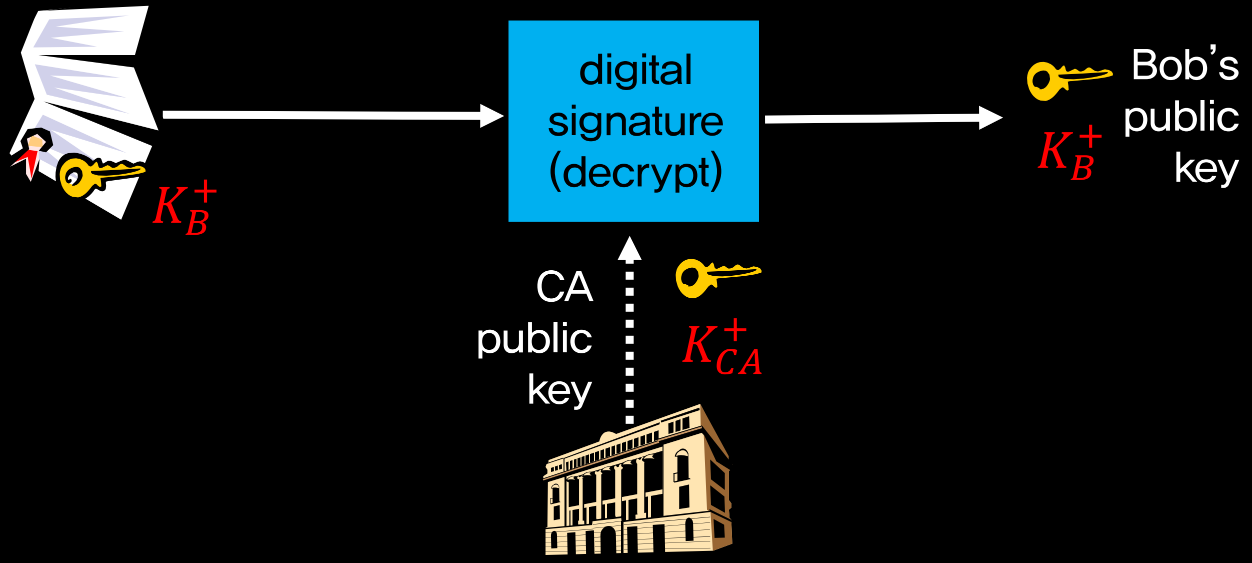
Certification Authorities (1/2)

- 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, saying "this is E's public key"


Bob's public key
K_B^+
digital signature (encrypt)
K_B^+
certificate for Bob's public key, signed by CA
Bob's identifying information
CA private key
K_{CA}^-

Certification Authorities (2/2)

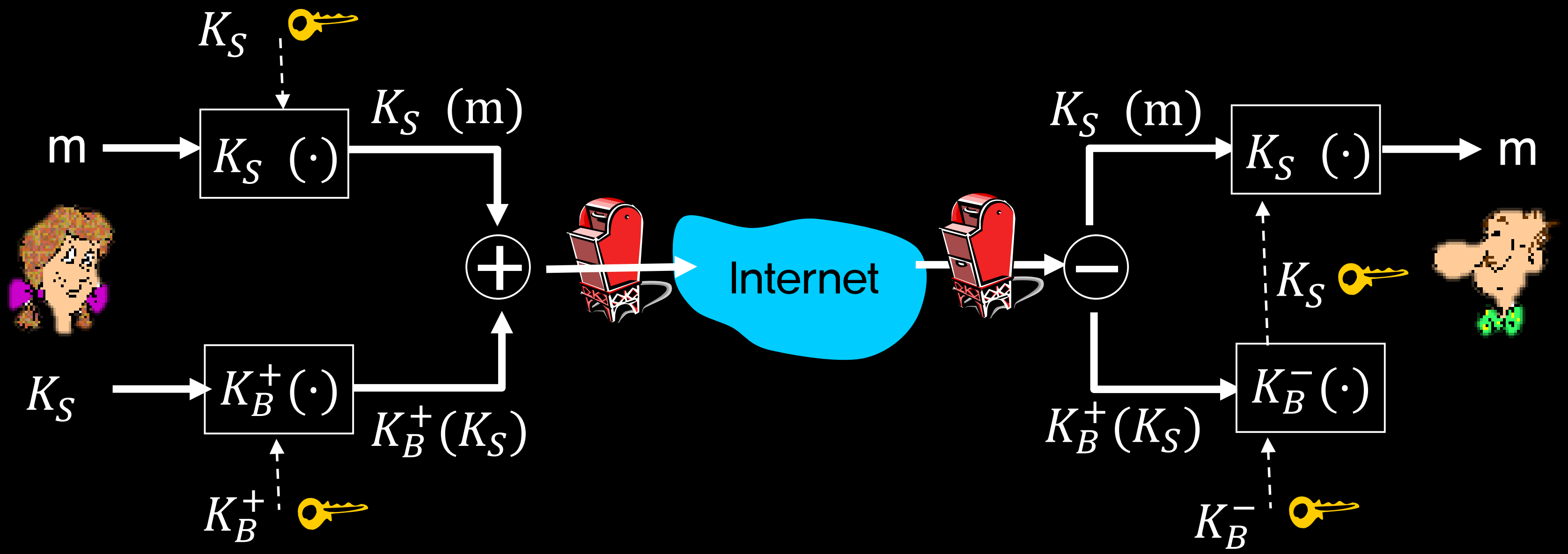
- When Alice wants Bob's public key:
  - gets Bob's certificate (Bob or elsewhere)
  - apply CA's public key to Bob's certificate, get Bob's public key



05. Securing E-mail

Secure E-mail

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

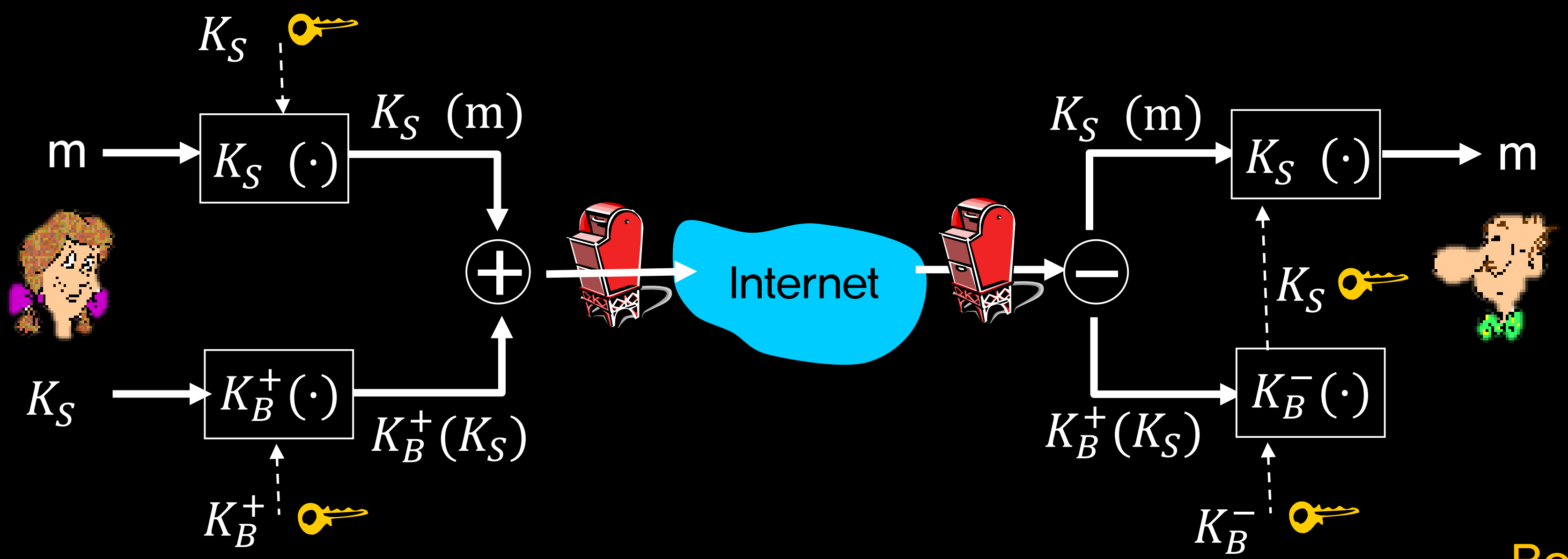


Alice:

- generates random symmetric private key, K_S
- encrypts message with K_S (for efficiency)
- also encrypts K_S with Bob's public key K_B^+
- sends both $K_S(\mathrm{m})$ and $K_B^+(K_S)$ to Bob

Secure E-mail

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



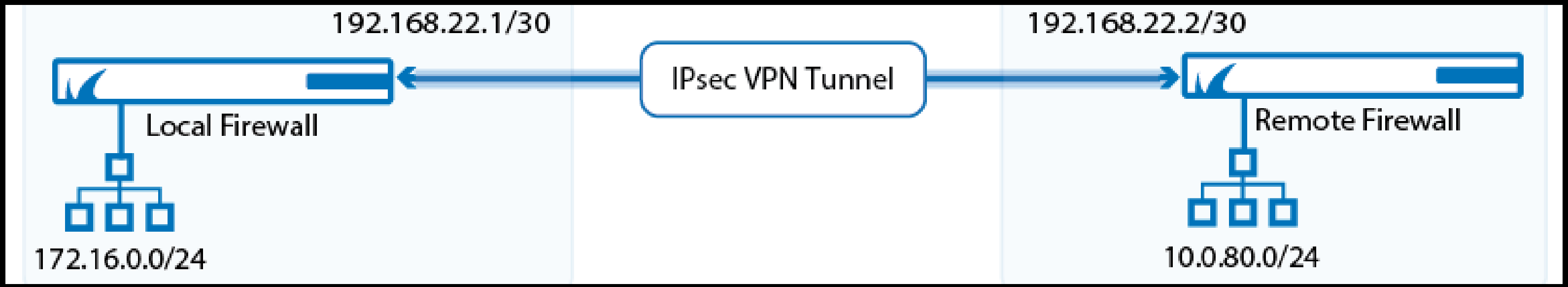
Bob:

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

06. IPsec and VPNs

Network Layer Confidentiality

- IPsec: IP security protocol
  - secures IP datagrams between any two network-layer entities, including host and routers
  - is used to create Virtual Private Networks (VPNs) that run over the public Internet

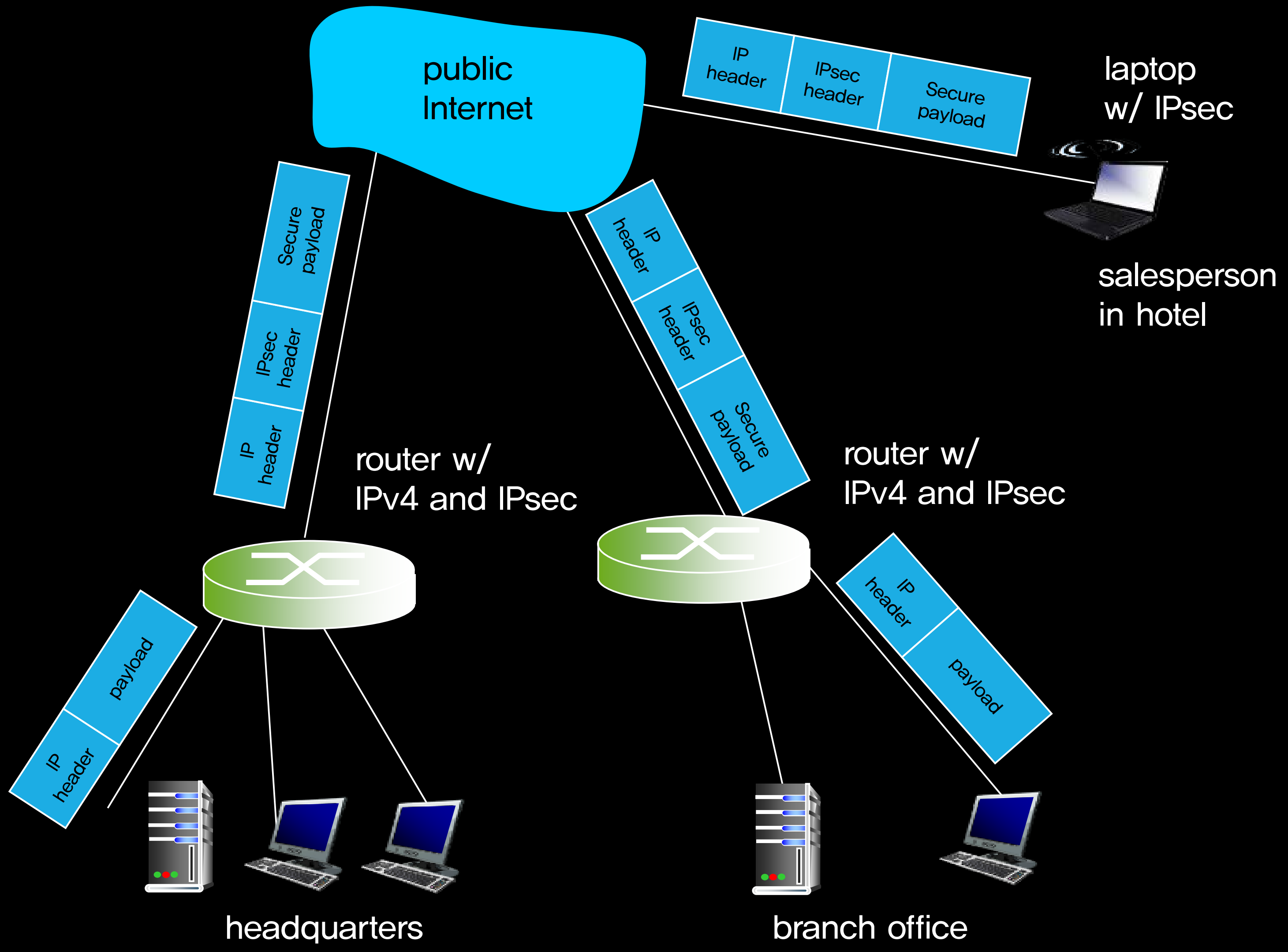


출처 – https://www.google.co.kr/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&ved=2ahUKEwjWjeT3qlXdAhWEZt4KHdkMCj8QjRx6BAgBEAU&url=https%3A%2F%2Fcampus.barracuda.com%2Fproduct%2Fcloudgenfirewall%2Fdoc%2F53248601%2Fhow-to-configure-bgp-routing-over-a-ikev1-ipsec-vpn-tunnel%2F&psig=AOvVaw0bx81PDLRmKyTnZlGA28Mg&ust=1535186165878916

- With network-layer confidentiality, the sending entity encrypts the payloads of all the datagrams it sends to the receiving entity
  - "blanket coverage": all data sent from one entity to the other would be hidden from any third party that might be sniffing the network

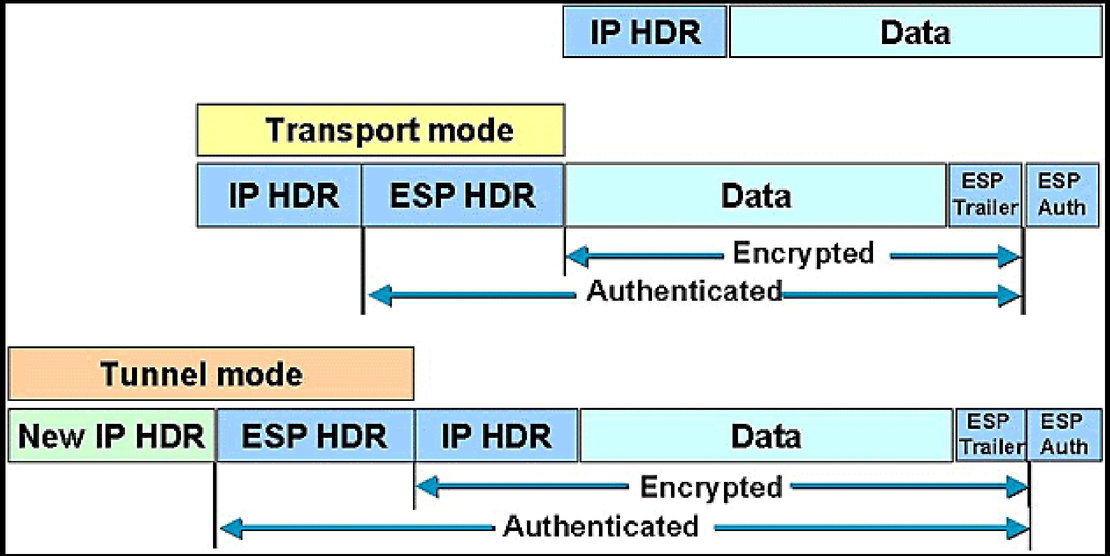
Virtual Private Networks (VPNs)

- Private networks for security are very costly
- VPN: institution's inter-office traffic is sent over public Internet instead
  - encrypted before entering public Internet
  - logically separate from other traffic



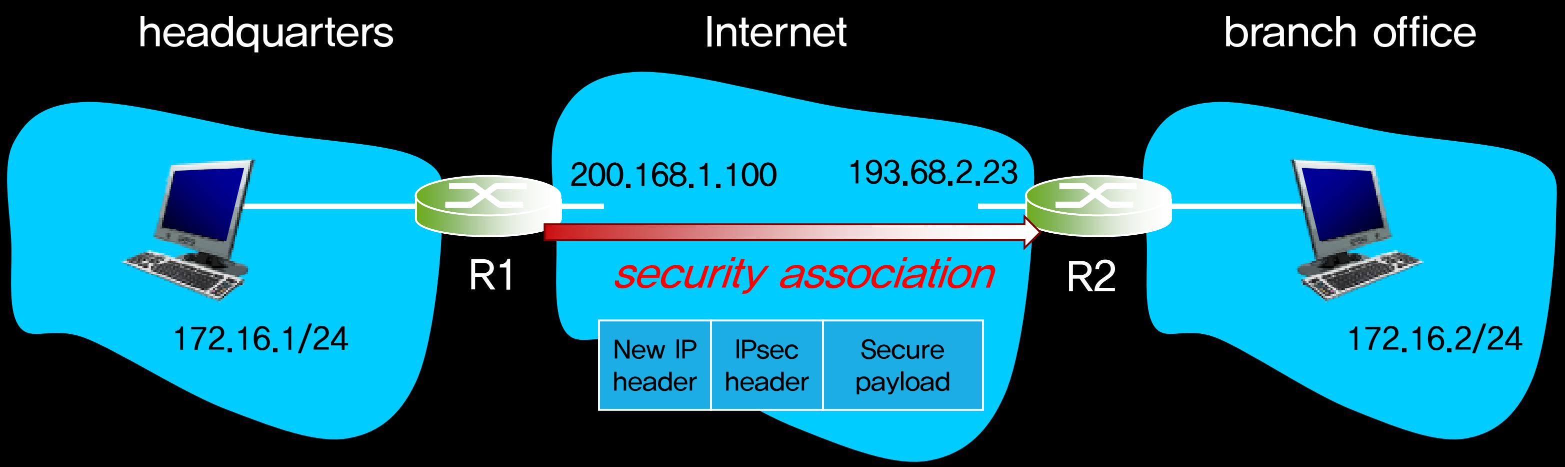
Protocols and Packets

- Two IPsec protocols
  - Authentication Header (AH) protocol
    - source authentication & data integrity but not confidentiality
  - Encapsulation Security Protocol (ESP)
    - source authentication, data integrity, and confidentiality
    - more widely used than AH
- Two packet forms
  - transport mode (host mode)
    - protects upper level protocols
  - tunnel mode
    - more appropriate for VPNs



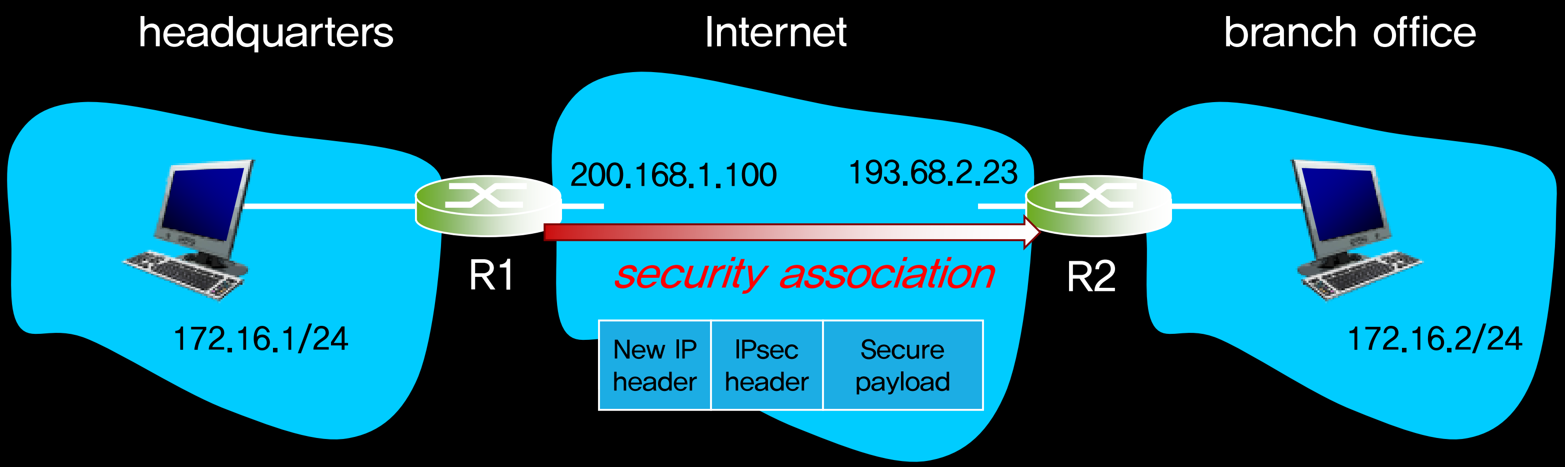
출처 – https://www.google.co.kr/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwiLt_PckIrdAhVGQd4KHai0ChUQjRx6BAgBEAU&url=http%3A%2F%2Fwww.ciscopress.com%2Farticles%2Farticle.asp%3Fp%3D25477&psig=AOvVaw10oO8plNA0ADf_RUfNNHeq&ust=1535352703547512

Security Associations (SAs)



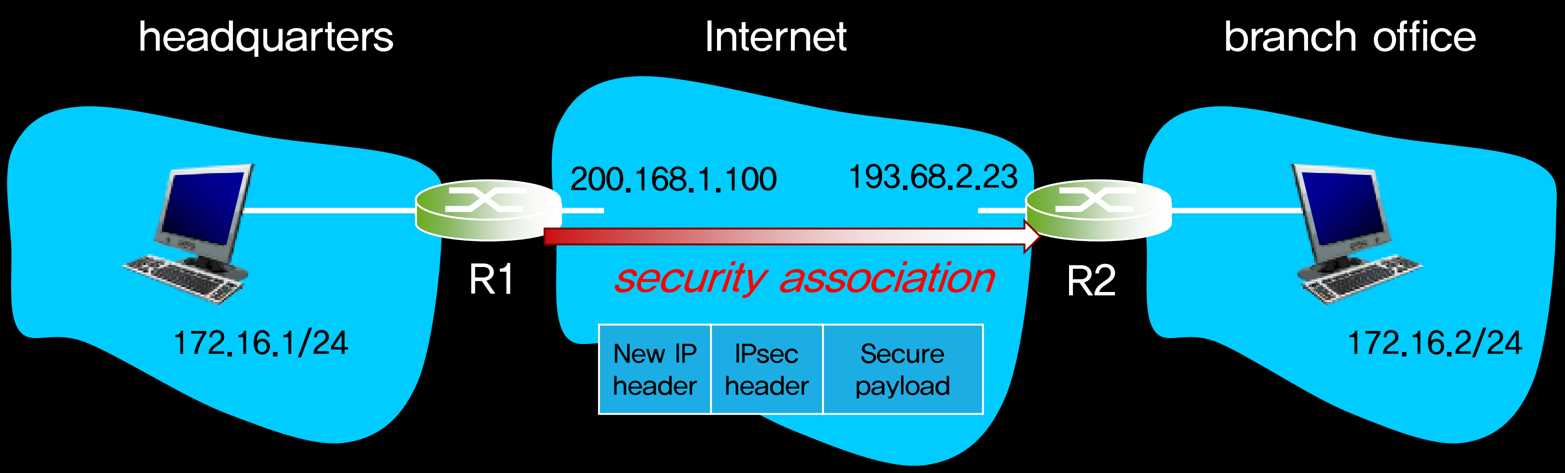
- Before sending data, "security association (SA)" established from sending to receiving entity
  - network-layer logical connection
  - SAs are simplex for only one direction, thus two SAs are needed for a pair of entities
- Sending and receiving entitles maintain state information about SA
  - 32-bit SA identifier: *Security Parameter Index* (SPI)
  - origin SA interface (200.168.1.100) and destination SA interface (193.68.2.23)
  - type of encryption used (e.g., 3DES) and encryption key
  - type of integrity check used (e.g., HMAC with MD5) and authentication key

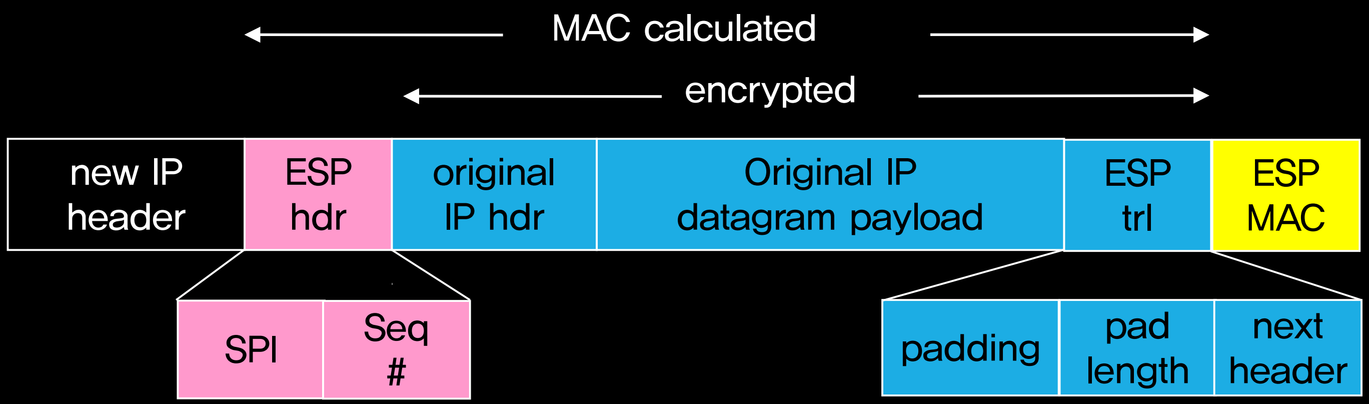
Security Association Database (SAD)



- Endpoint holds SA state in security association database (SAD), where it can locate them during processing
- 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

ESP Tunnel Format





- MAC: created with shared secret key
- ESP trailer: padding for block ciphers

- SPI: so receiving entity knows what to do
- Sequence number: to thwart replay attacks

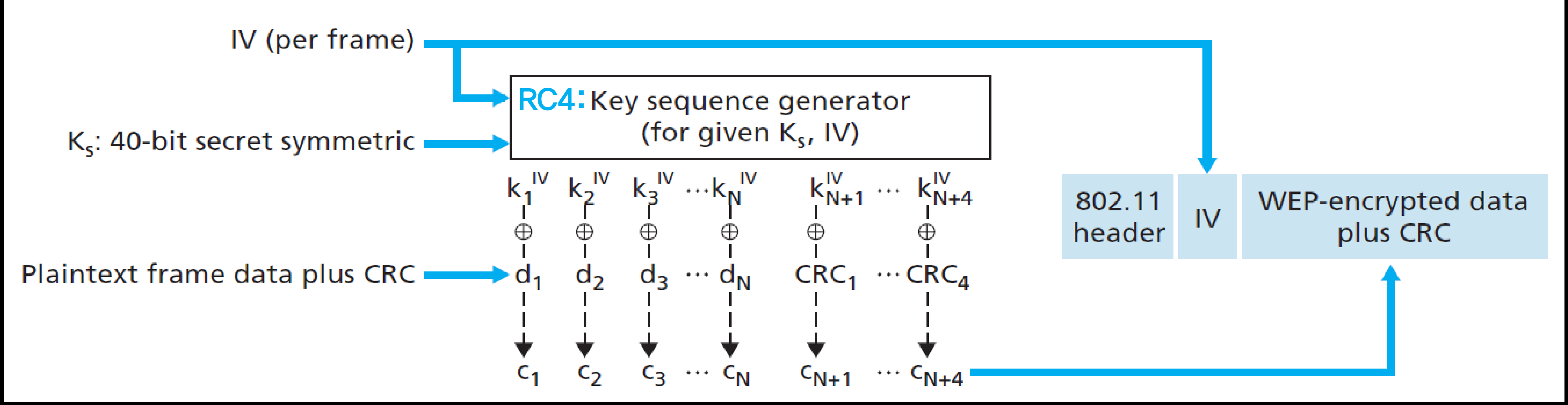
Internet Key Exchange (IKE) Protocol

- Through IKE protocol
  - the two entities exchange certificates,
  - negotiate authentication and encryption algorithms, and
  - securely exchange key material for creating session keys in the IPsec SAs

- IKE has two phases:
  - phase 1: establish bi-directional IKE SA
    - Diffie-Hellman algorithm (see Homework Problem P9): a kind of public-key algorithm
  - phase 2: securely negotiate the IPsec encryption and authentication for a pair of SAs

07. Wi-Fi Security

Wired Equivalent Privacy (WEP)


IV (per frame)
RC4: Key sequence generator
(for given K_s, IV)
K_s: 40-bit secret symmetric
k_1^{IV} k_2^{IV} k_3^{IV} … k_N^{IV} k_{N+1}^{IV} … k_{N+4}^{IV}
Plaintext frame data plus CRC
d_1 d_2 d_3 … d_N CRC_1 … CRC_4
c_1 c_2 c_3 … c_N c_{N+1} … c_{N+4}
802.11 header
IV
WEP-encrypted data plus CRC

- For confidentiality, RC4 produces a stream of key values $(k_1^{IV}, k_2^{IV}, \cdots)$ using a 64–bit key and encrypts data and 4–byte CRC by XOR operation: $c_i = d_i \oplus k_i^{IV}$
- The 64–bit key is composed of 40–bit shared secret and 24–bit initialization vector (IV) which sender creates
  - new IV for every frame ⇨keys to RC4 changed for every frame
  - sent in plaintext ⇨the same key stream can be generated by receiver $d_i = c_i \oplus k_i^{IV}$

WEP Vulnerabilities

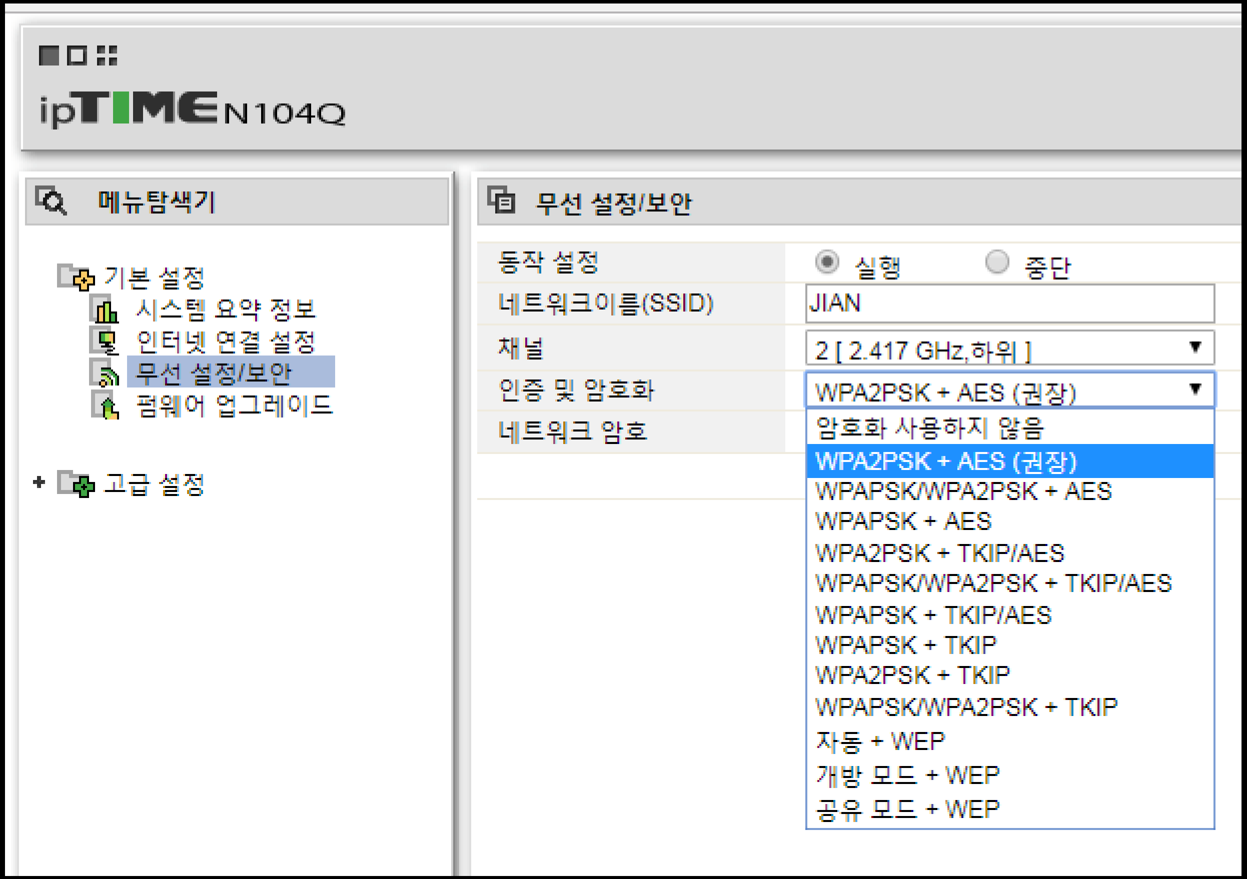
- Short and static key
  - actual keyspace is 40 bits
  - keys manually shared between AP and hosts
- IV is 24-bit long
  - only 2^{24} unique keys ⇨ same IV value with more than 99% chance after 12,000 frames
    - only a few seconds with 1 Kbyte frame sizes and 11 Mbps data transmission rate
  - IV is sent in plaintext, thus sending a request to transmit a file with known content d_1, d_2, d_3,⋯, an attacker can get to know the key stream k_i^{IV}s for a specific IV by XOR-ing of original data and encrypted data

$$d_i \oplus c_i = k_i^{IV}$$

  - pairs of an IV and the corresponding key stream can be stored into a table

Wi-Fi Protected Access (WPA)

- Developed as an intermediate measure until the availability of the full IEEE 802.11i standard (2004)
- PSK (Pre-Shared Key)
- TKIP (Temporal Key Integrity Protocol): dynamically generates a new 128-bit key for each packet
- MIC (Message Integrity Check)
- Encryption algorithm
  - WPA: RC4
  - WPA2: AES



ipTIME N104Q

메뉴탐색기

- 기본 설정
  - 시스템 요약 정보
  - 인터넷 연결 설정
  - 무선 설정/보안
  - 펌웨어 업그레이드
- 고급 설정

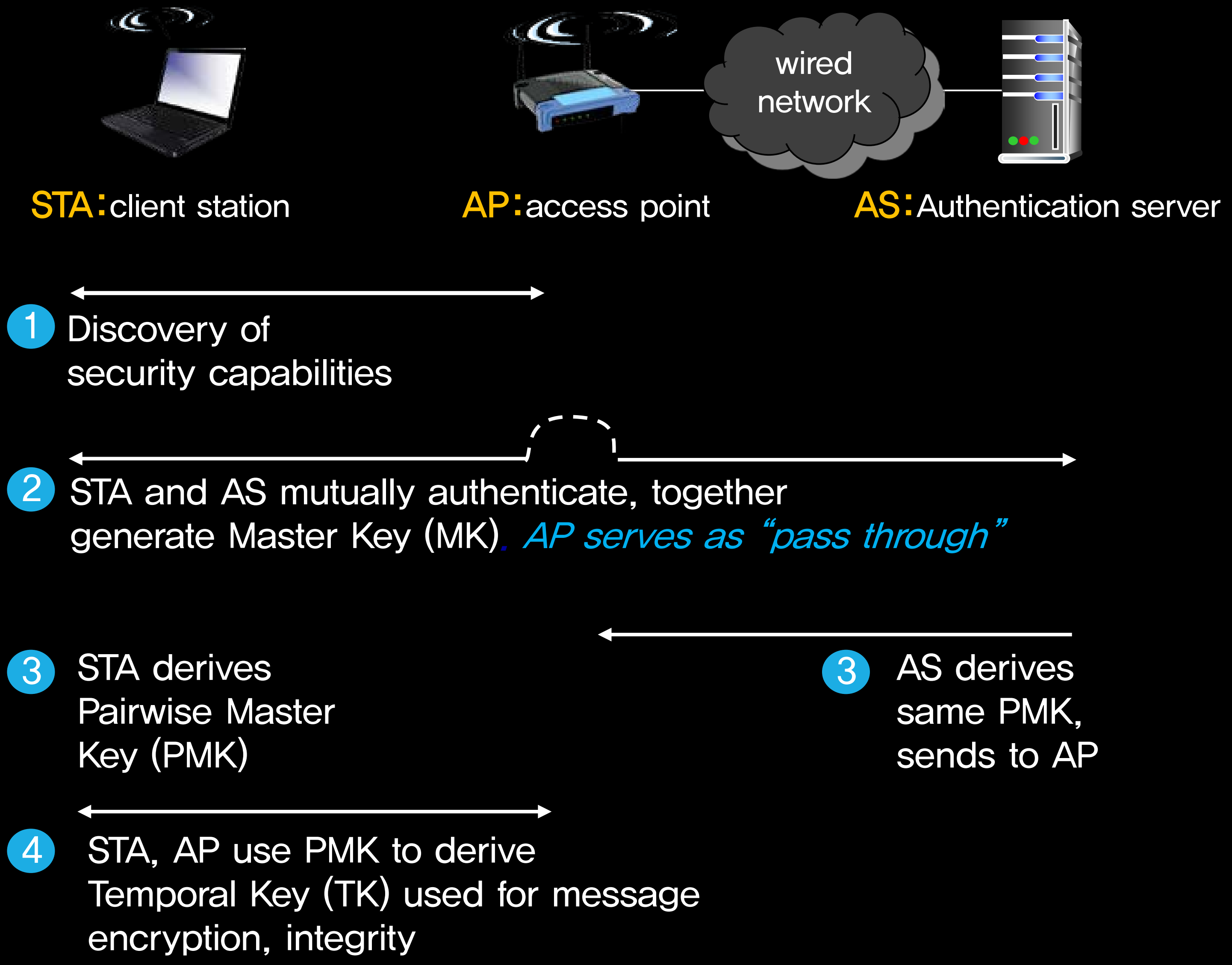
무선 설정/보안

동작 설정	● 실행 ○ 중단
네트워크이름(SSID)	JIAN
채널	2 [2.417 GHz,하위]
인증 및 암호화	WPA2PSK + AES (권장)
네트워크 암호	

암호화 사용하지 않음
WPA2PSK + AES (권장)
WPAPSK/WPA2PSK + AES
WPAPSK + AES
WPA2PSK + TKIP/AES
WPAPSK/WPA2PSK + TKIP/AES
WPAPSK + TKIP/AES
WPAPSK + TKIP
WPA2PSK + TKIP
WPAPSK/WPA2PSK + TKIP
자동 + WEP
개방 모드 + WEP
공유 모드 + WEP

802.11i: Improved Security

- Features
  - extensible set of authentication mechanisms
  - a key distribution mechanism (not a PSK method)
- Authentication server separated from access point



STA: client station

AP: access point

AS: Authentication server

1 Discovery of security capabilities

2 STA and AS mutually authenticate, together generate Master Key (MK). *AP serves as "pass through"*

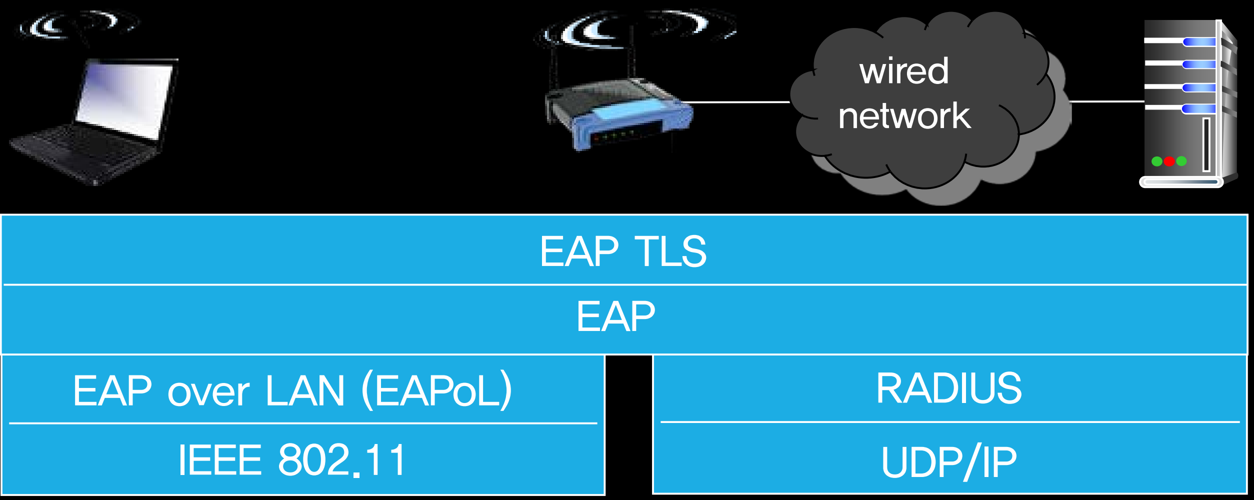
3 STA derives Pairwise Master Key (PMK)

3 AS derives same PMK, sends to AP

4 STA, AP use PMK to derive Temporal Key (TK) used for message encryption, integrity

Extensible Authentication Protocol (EAP)

- 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)



EAP TLS

EAP

EAP over LAN (EAPoL)

IEEE 802.11

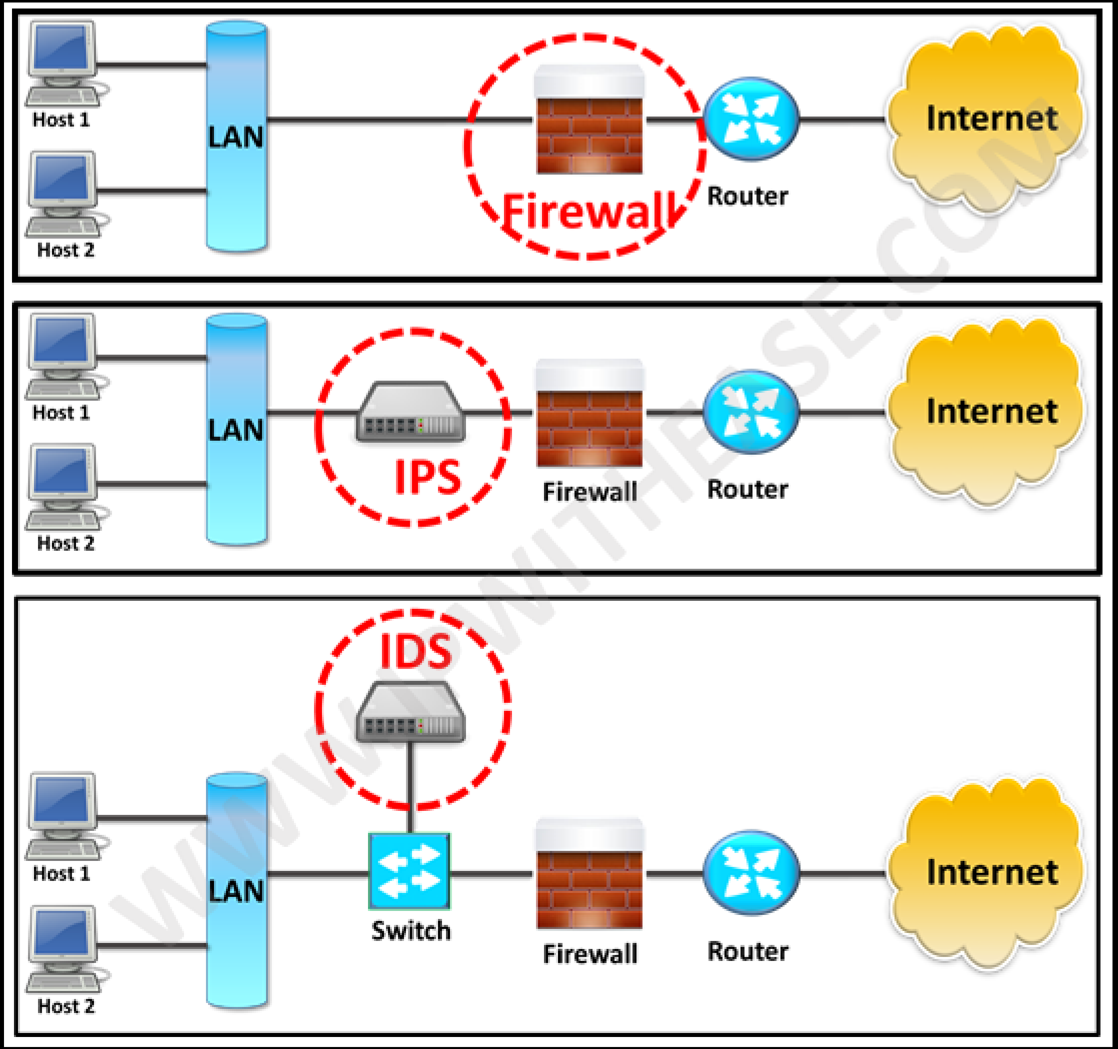
RADIUS

UDP/IP

08. Firewall and IDS/IPS

Firewall, IDS, and IPS

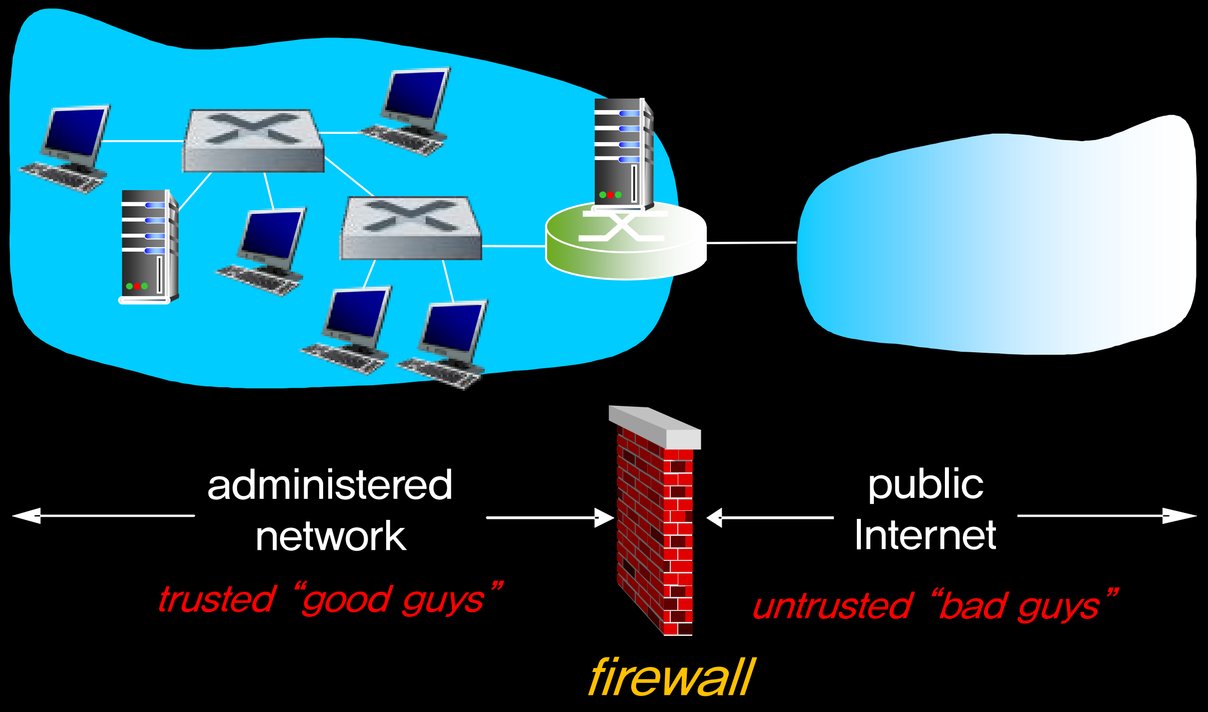
- Firewall
  - a device or application that enforces policy based on packet header information such as protocol type, src IP, dest IP, src port, and/or dest port number
- IDS (Intrusion Detection System) / IPS (Intrusion Protection System)
  - a device or application that analyzes whole packets, both header and payload, looking for suspicious events; if an event detected,
    - IDS: a log message is generated
    - IPS: the packet is rejected



출처 – https://www.google.co.kr/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwjj6MfUzo7dAhVMA4gKHQfIA-sQjRx6BAgBEAU&url=https%3A%2F%2Fipwithease.com%2Ffirewall-vs-ips-vs-ids%2F&psig=AOvVaw1mRPf-1qOscthEU7tspSvc&ust=1535506626466447

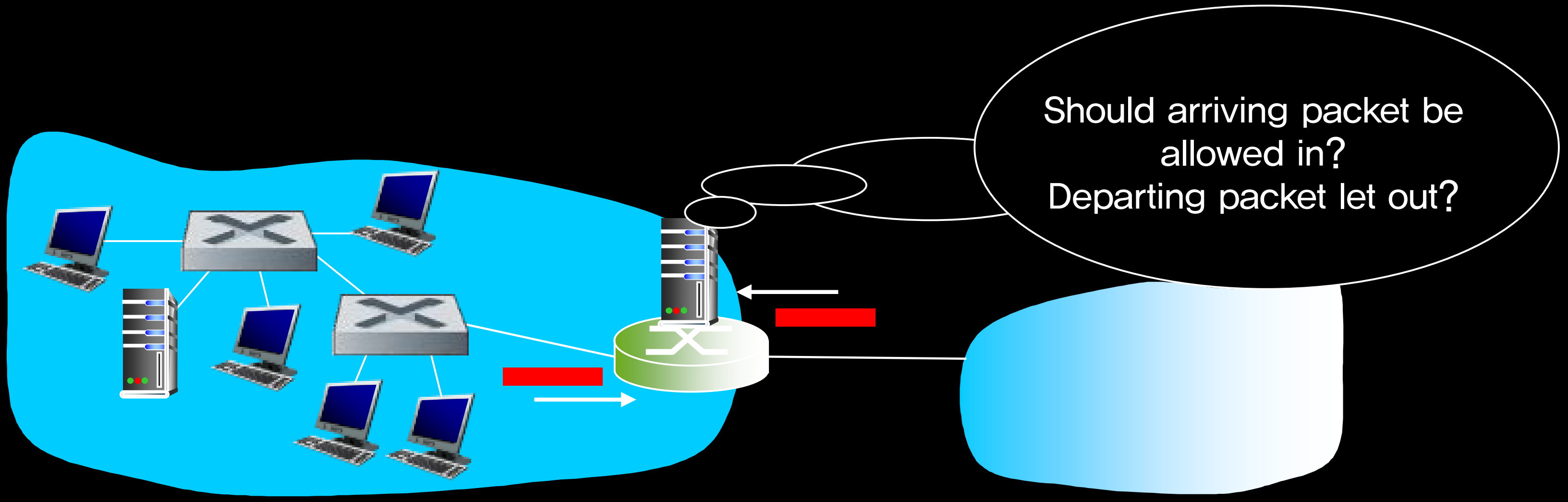
Firewall

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



- Three categories
  - stateless packet filters
  - stateful packet filters
  - application gateways

Stateless Packet Filter



Should arriving packet be allowed in?
Departing packet let out?

- 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 Filter Example

- Access control lists for a router interface

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 Filter

- Example of the security hole of stateless packet filter
  - may admit 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 Filter

- Connection table

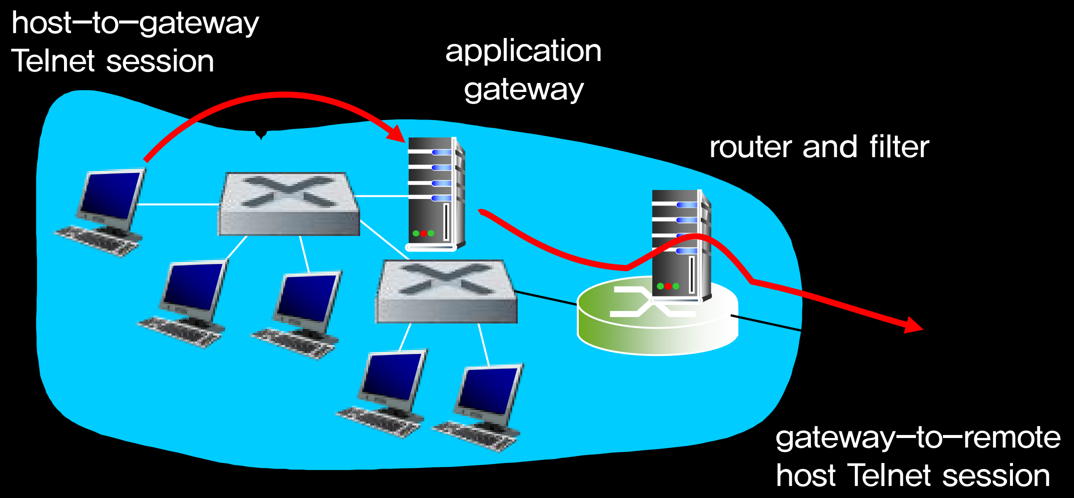
source address	dest address	source port	dest port
222.22.1.7	37.96.87.123	12699	80
222.22.93.2	199.1.205.23	37654	80
222.22.65.143	203.77.240.43	48712	80

- 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

Application Gateways

- Application gateway
  - an application-specific server through which all application data must pass
  - filters packets on application data as well as on IP/TCP/UDP fields



- Example: allow selected internal users to telnet outside

1) Router is set up to filter blocks all Telnet connections not originating from gateway
2) All Telnet users must telnet through gateway
3) For only authorized users, gateway sets up telnet connection to destination host. Gateway relays data between the two connections

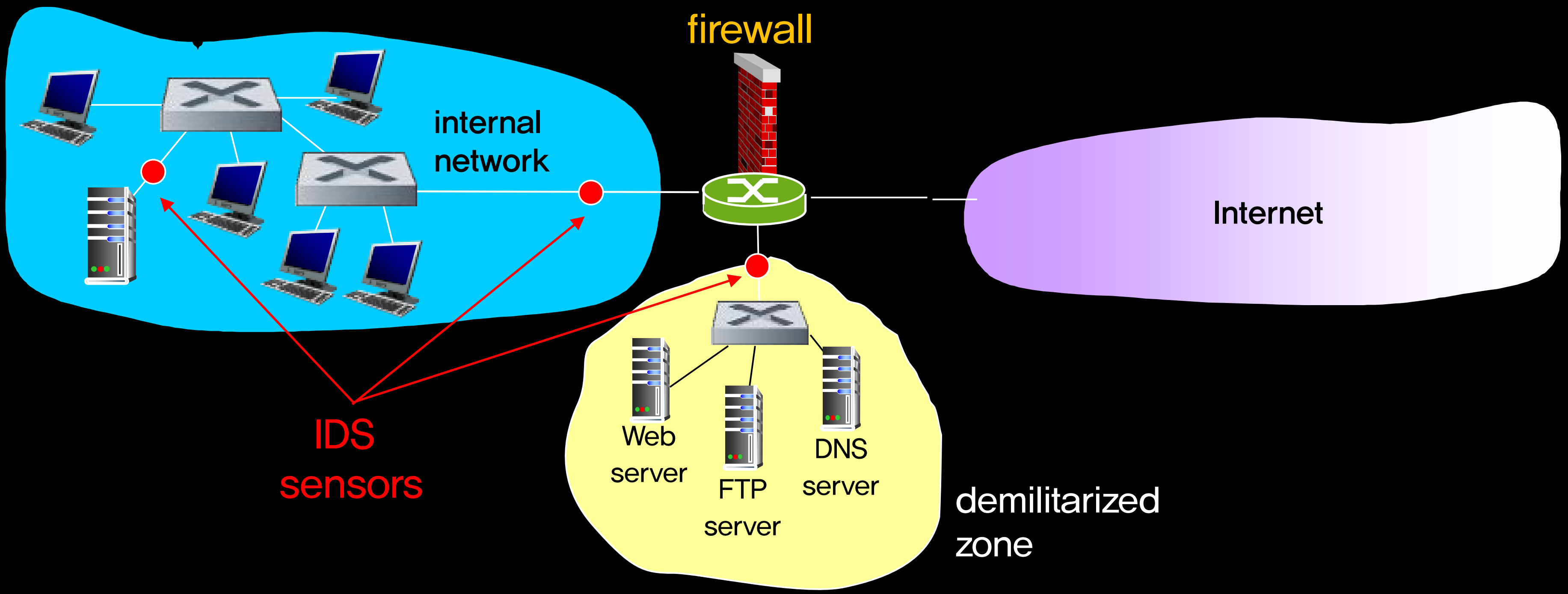
Intrusion Detection/Protection System (IDS/IPS)

- Firewall packet filtering
  - operates on TCP/IP headers only
    - cannot handle IP spoofing

      (cannot know if data “really” comes from claimed source or not)
  - no correlation check among sessions
- IDS/IPS
  - 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

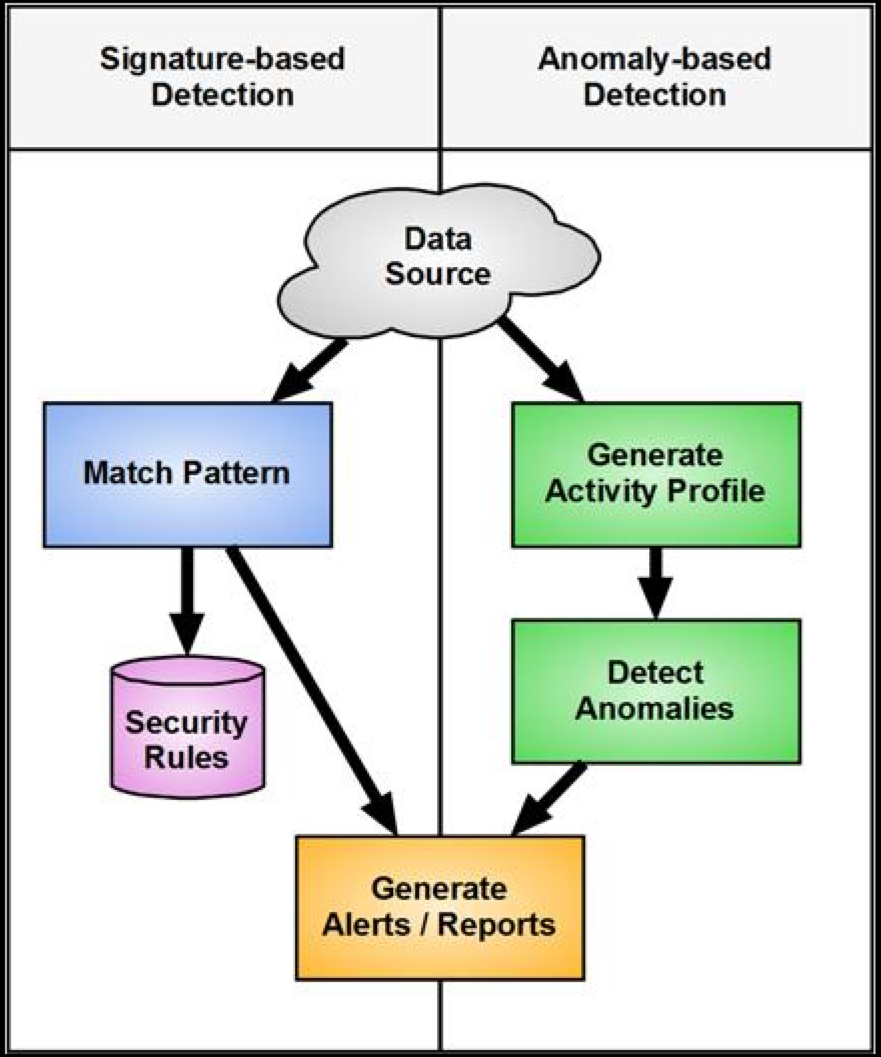
Intrusion Detection Systems

- Multiple IDSs
  - distribution of a significant amount of processing
  - different types of checking at different locations



IDS Types

- Signature–based system
  - maintains an extensive database of attack signatures
  - signature is a set of rules pertaining to an intrusion activity
    - may simply be a list of characteristics about a single packet or may relate to a series of packets
  - sniffs every packet passing by it, comparing each sniffed packet with the signatures in its database
  - requires previous knowledge of the attack to generate an accurate signature ⇨ completely blind to new attacks
- Anomaly–based system
  - creates a traffic profile as it observes in normal operation
  - looks for packet streams that are statistically unusual, e.g, an inordinate percentage of ICMP packets
  - extremely challenging to distinguish between normal traffic and statistically unusual traffic



출처 – https://www.google.co.kr/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwjf2qHvi4_dAhUFdt4KHdueBwcQjRx6BAgBEAU&url=https%3A%2F%2Fwww.researchgate.net%2Ffigure%2FSIGNATURE–AND–ANOMALY–BASED–IDS–5_fig1_297171228&psig=AOvVaw0WE_4u6k4flv3TmN619byL&ust=1535523139929256

Summary

Summary (1/2)

01

Network Security

- security properties: confidentiality, integrity, authentication
- examples of network security attacks

02

Cryptography Principles

- types of cryptosystem: symmetric vs. asymmetric cryptography
- DES vs. RSA

03

Message Integrity

- shared secret key method: MAC (message authentication code)
- public key mechanism: digital signature

04

End-Point Authentication

- nonce with secret key
- nonce with public key + certificate authority

Summary (2/2)

05

Securing E-mail

- encrypt messages with a symmetric key the sender generates
- encrypt the symmetric key with receiver's public key

06

IPsec and VPNs

- IPsec: IP security protocol
- VPNs using security association (SA) between routers

07

Wi-Fi Security

- operation and weakness of Wired Equivalent Privacy (WEP)
- IEEE 802.11i

08

Firewall and IDS/IPS

- firewall: stateless/stateful packet filter, application gateway
- IDS/IPS: signature-based system, anomaly-based system