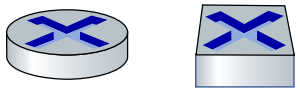
Network Security

Instructor: Sheikh Rabiul Islam, Ph.D.

Background: The Internet - a "nuts and bolts" view

Billions of connected computing *devices*:

- *hosts = end systems*
- running *network apps* at Internet's "edge"



Packet switches: forward packets (chunks of data)

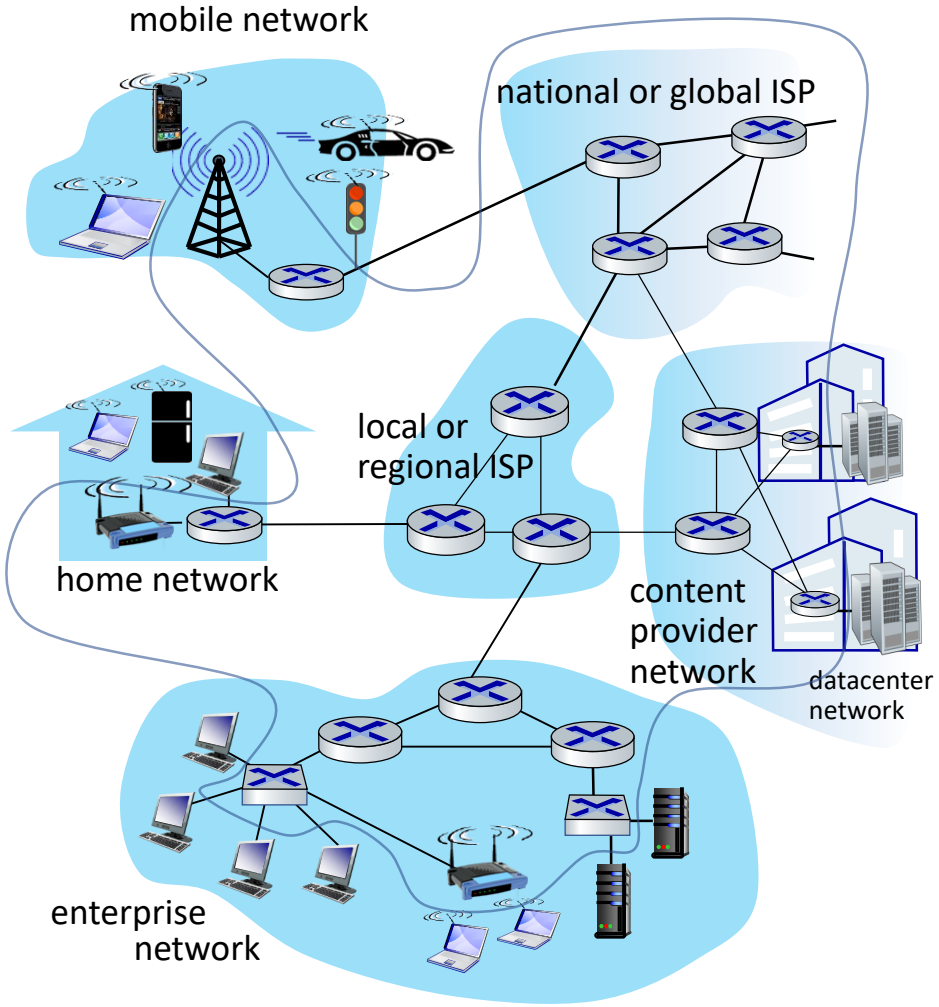
- *routers, switches*

Communication links

- fiber, copper, radio, satellite
- transmission rate: *bandwidth*

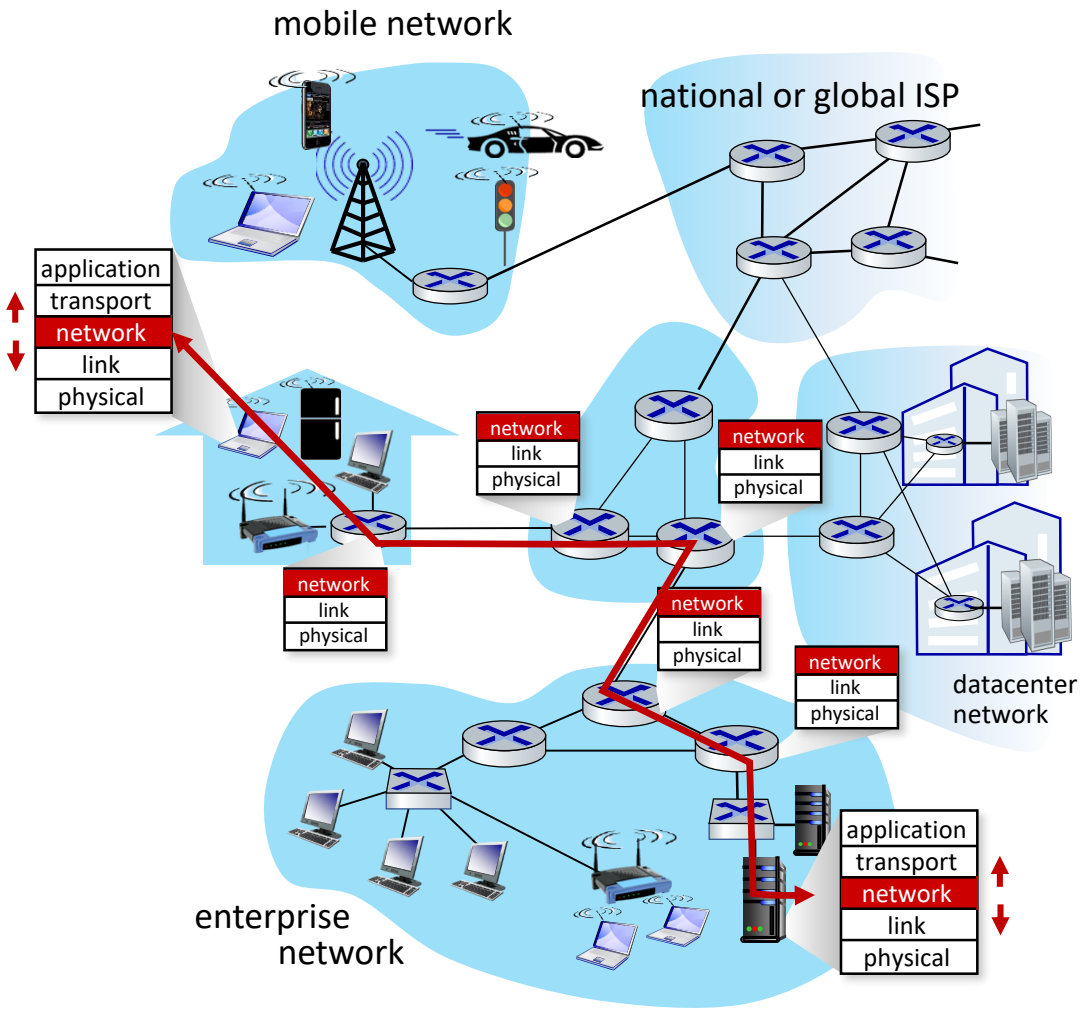
Networks

- collection of devices, routers, links: managed by an organization



Background: Network-layer services and protocols

- transport segment from sending to receiving host
  - sender: encapsulates segments into datagrams, passes to link layer
  - receiver: delivers segments to transport layer protocol
- network layer protocols in *every Internet device*: hosts, routers
- routers:
  - examines header fields in all IP datagrams passing through it
  - moves datagrams from input ports to output ports to transfer datagrams along end-end path



Network Security: overview

- understand principles of network security:
  - cryptography and its many uses beyond “confidentiality”
  - authentication
  - message integrity

- security in practice:
  - firewalls and intrusion detection systems
  - security in application, transport, network, link layers

Lecture Outline

- What is network security?
- Principles of cryptography

What is network security?

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

- sender encrypts message
- receiver decrypts message

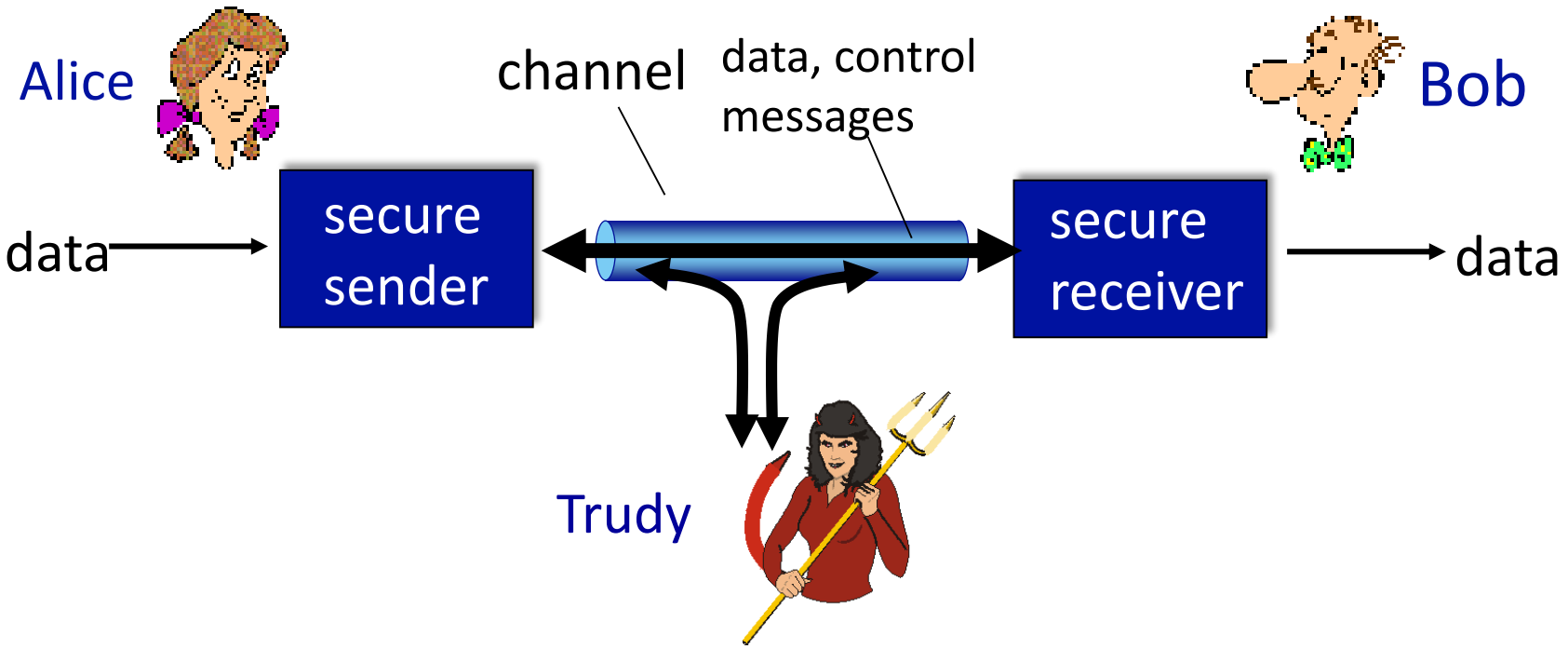
authentication: sender, receiver want to confirm identity of each other

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

Friends and enemies: Alice, Bob, Trudy

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



Friends and enemies: Alice, Bob, Trudy

Who might Bob and Alice be?

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

Any other example?

- DNS Servers

There are bad people (attacker) out there!

Q: What can a "bad people" do?

A: A lot!

- 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/herself in place
- denial of service: prevent service from being used by others (e.g., by overloading resources)

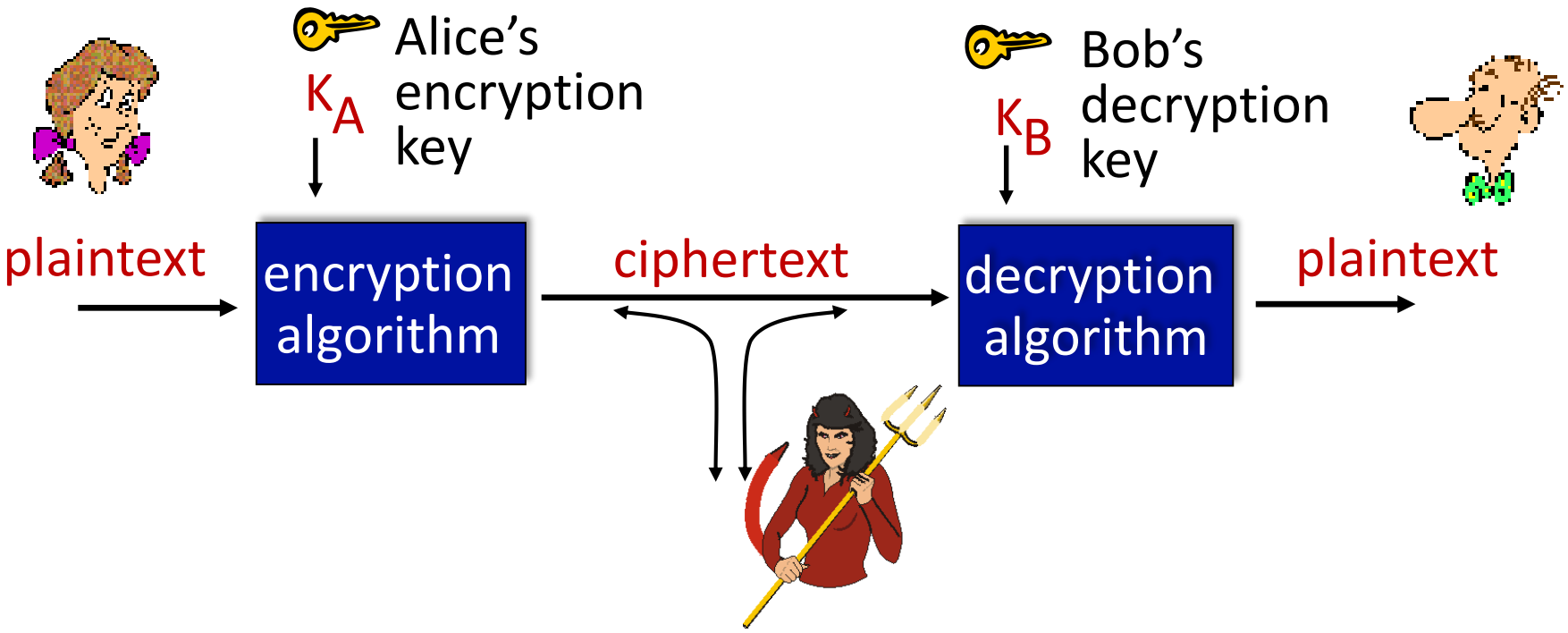
Example: DNS poisoning or DNS spoofing: https://usa.kaspersky.com/resource-center/definitions/dns

Outline

- What is network security?
- Principles of cryptography

5 minutes video lecture to view before class:
https://www.youtube.com/watch?v=AQDCe585Lnc

The language of cryptography



m: plaintext message : "Hello World"

$K_A(m)$: ciphertext, encrypted with key K_A : "A46RD89JT3HHFWQ1"

$m = K_B(K_A(m))$ = "Hello World"

Caesar Cipher

First we translate all of our characters to numbers, 'a'=0, 'b'=1, 'c'=2, ... , 'z'=25. We can now represent the caesar cipher encryption function, e(x), where x is the character we are encrypting, as:

$$e(x) = (x + k) \pmod{26}$$

Where k is the key (the shift) applied to each letter. After applying this function the result is a number which must then be translated back into a letter. The decryption function is :

$$e(x) = (x - k) \pmod{26}$$

Plain Text: ABZ
Key = 1
Cipher Text: BCA

Q: how do sender and receiver agree on the value of the key?

Further reading:
http://practicalcryptography.com/ciphers/caesar-cipher/
https://www.geeksforgeeks.org/caesar-cipher-in-cryptography/

Symmetric key crypto: DES

DES: Data Encryption Standard

- US encryption standard [NIST 1993]
- 56-bit symmetric key, 64-bit plaintext input
- block cipher with cipher block chaining
- how secure is DES?
  - DES Challenge: 56-bit-key-encrypted phrase decrypted (brute force) in less than a day
  - no known good analytic attack
- making DES more secure:
  - 3DES: encrypt 3 times with 3 different keys

Breaking an encryption scheme

- cipher-text only attack:
  Trudy has ciphertext she can analyze
- two approaches:
  - brute force: search through all keys
  - statistical analysis (e.g., letter **e** and **t** are most frequent letters)
- known-plaintext attack:
  Trudy has plaintext corresponding to ciphertext
  - *e.g.,* in monoalphabetic cipher, Trudy determines pairings for t,h,e; i,t; i,n; i,n,g;
- chosen-plaintext attack:
  Trudy can get ciphertext for chosen plaintext

Substitution Cracking Tool: https://simonsingh.net/The_Black_Chamber/substitutioncrackingtool.html

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

Public Key Cryptography

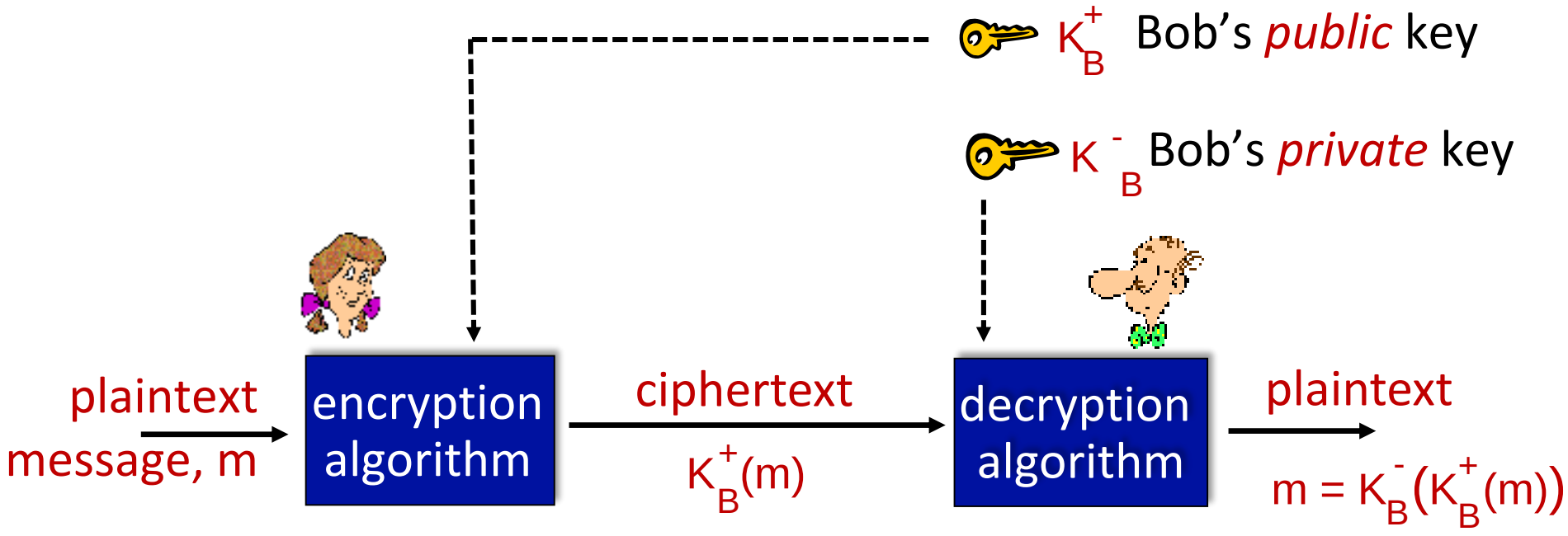
symmetric key crypto:

- 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

Public Key Cryptography



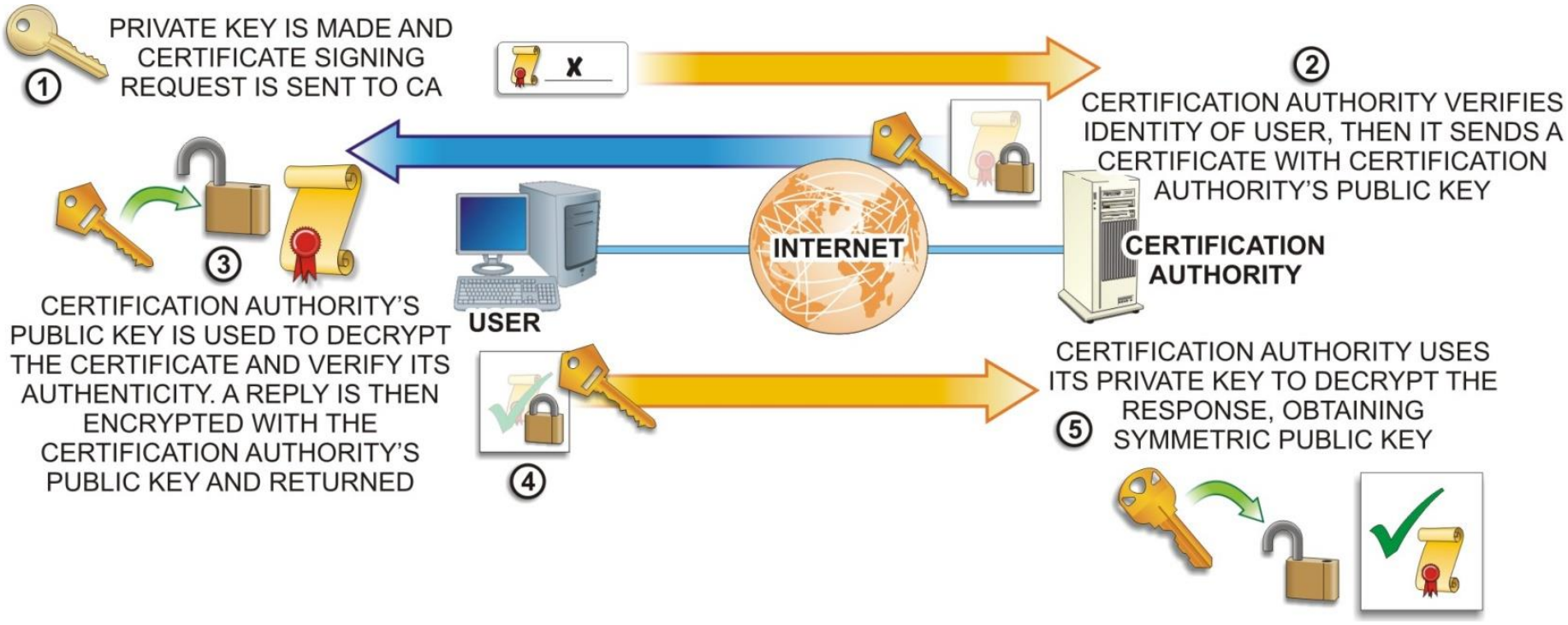
Wow - public key cryptography revolutionized 2000-year-old (previously only symmetric key) cryptography!

- similar ideas emerged at roughly same time, independently in US and UK (classified)

Public-key Cryptography

- Transport Layer Security and Secure Sockets Layer (TLS and SSL)
- Secure Shell (SSH)
- PGP (Pretty Good Privacy)
- GNU Privacy Guard (GPG)
- Secure/Multipurpose Internet Mail Extensions (S/MIME)
- Digital Signature Standard (DSS)
- RSA encryption algorithm

Digital Certificates



More about Digital Certificate: https://www.fortinet.com/resources/cyberglossary/digital-certificates