

Chương 8: Bảo mật Mạng

Các mục tiêu:

- ❑ hiểu rõ các nguyên lý của bảo mật mạng:
 - mật mã học và những ứng dụng của nó cho "tính bí mật"
 - xác thực
 - toàn vẹn thông điệp
- ❑ bảo mật trong thực tế:
 - tường lửa và các hệ thống phát hiện xâm nhập
 - bảo mật trong các tầng ứng dụng, truyền tải, mạng, liên kết

Chương 8 Mục lục

8.1 Bảo mật mạng là gì?

8.2 Các nguyên lý của mật mã

8.3 Toàn vẹn thông điệp

8.4 Bảo vệ email

8.5 Bảo vệ kết nối TCP: SSL

8.6 Bảo mật tầng Mạng: IPsec

8.7 Bảo vệ mạng LAN không dây

8.8 Bảo mật hành vi: tường lửa và IDS

Bảo mật mạng là gì?

Tính cơ mật: chỉ có n/gửi, n/nhận chủ định có thể "hiểu" nội dung thông điệp

- n/gửi mã hóa thông điệp
- n/nhận giải mã thông điệp

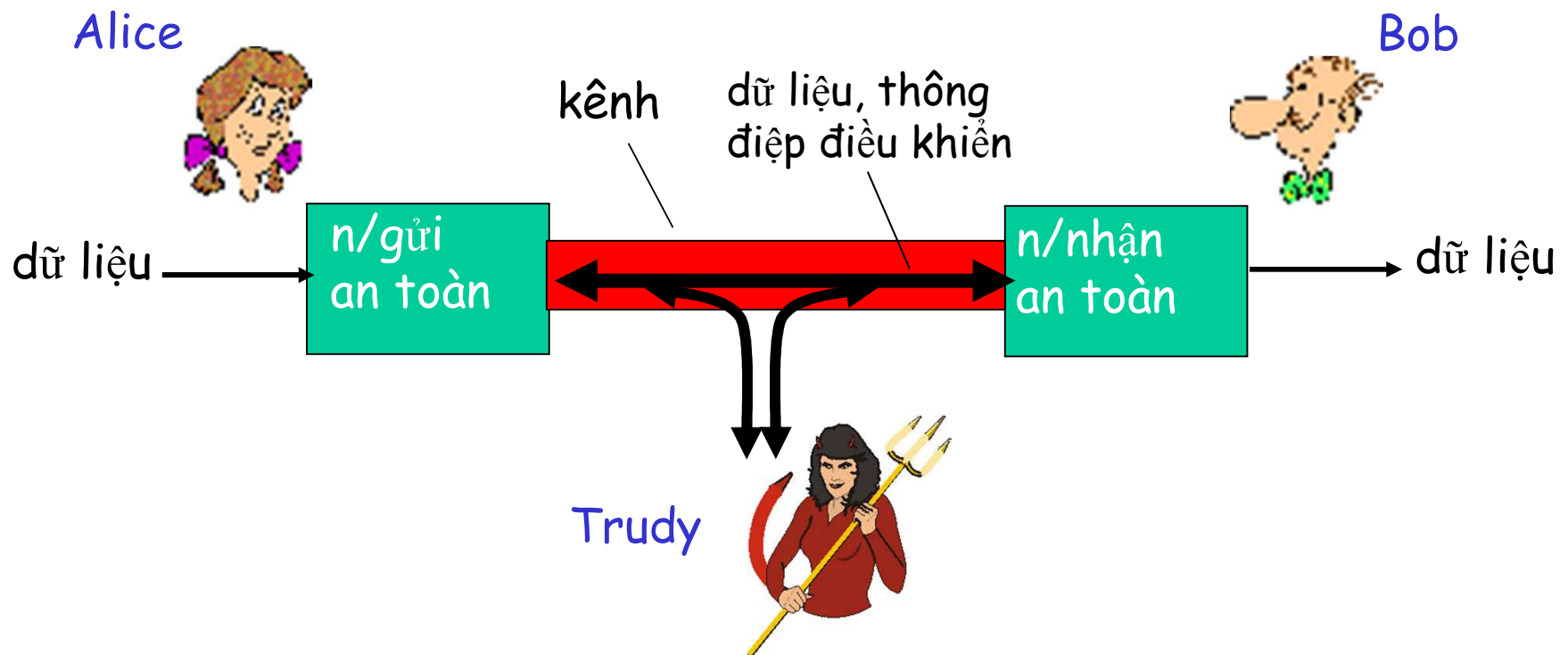
Xác thực: n/gửi, n/nhận muốn xác nhận đúng người mà mình đang nói chuyện

Toàn vẹn thông điệp: n/gửi, n/nhận muốn đảm bảo rằng thông điệp không bị thay đổi (trong quá trình gửi, hoặc sau đó) mà ko bị phát hiện

Khả năng truy cập và tính sẵn sàng: dịch vụ phải luôn truy cập được và sẵn sàng cho n/dùng

Bạn bè và kẻ thù: Alice, Bob, Trudy

- rất phổ biến trong thế giới bảo mật mạng
- Bob, Alice (người iêu!) muốn liên lạc "một cách bí mật"
- Trudy (ng xâm phạm) có thể may chặn, xóa, thêm thông điệp



Ai có thể là Bob, Alice?

- ❑ Những Bob và Alice ngoài đời thực!
- ❑ Trình duyệt/máy chủ Web cho giao dịch điện tử (vd: mua bán on-line)
- ❑ máy chủ/khách thực hiện tác vụ ngân hàng trực tuyến
- ❑ máy chủ DNS
- ❑ bộ định tuyến trao đổi thông tin cập nhật bảng định tuyến
- ❑ những ví dụ khác?

Luôn có những kẻ xấu trong mạng!

H: Kẻ xấu có thể làm gì?

Đ: Rất nhiều!

- *nghe lén*: các thông điệp
- chủ động *chèn* thông điệp vào kết nối
- *giả danh*: có thể giả (lừa) địa chỉ nguồn trong gói tin (hoặc bất kì trường nào trong gói)
- *chiếm quyền(hijacking)*: "kiểm soát" kết nối đang diễn ra bằng cách vô hiệu vai trò n/gửi và nhận, chèn bản thân hấn ta vào.
- *từ chối dịch vụ*: ngăn chặn việc cung cấp dịch vụ cho người dùng khác (vd: bằng cách làm quá tải bộ nhớ)

Chương 8 Mục lục

8.1 Bảo mật mạng là gì?

8.2 Các nguyên lý của mật mã

8.3 Toàn vẹn thông điệp

8.4 Bảo vệ email

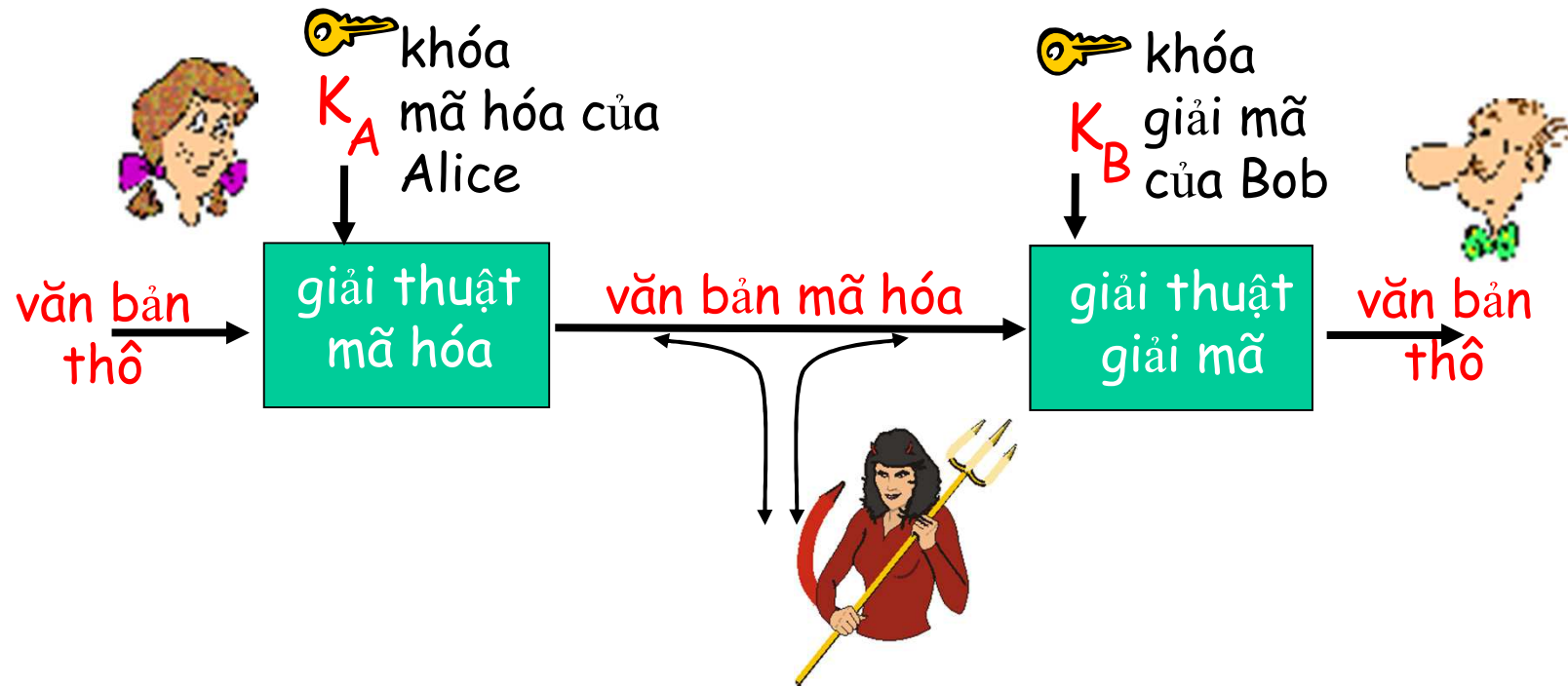
8.5 Bảo vệ kết nối TCP: SSL

8.6 Bảo mật tầng Mạng: IPsec

8.7 Bảo vệ mạng LAN không dây

8.8 Bảo mật hành vi: tường lửa và IDS

Ngôn ngữ của mật mã học



m thông điệp văn bản thô

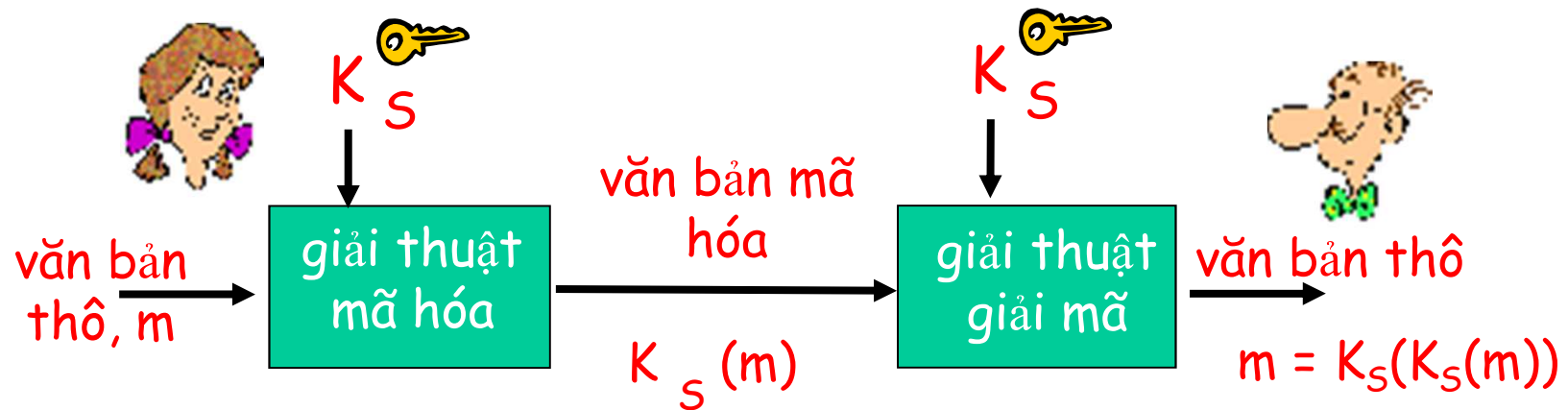
$K_A(m)$ văn bản mã hóa, mã hóa với khóa K_A

$m = K_B(K_A(m))$ giải mã văn bản đã được mã hóa

Các loại mã hóa

- ❑ Mật mã thường sử dụng khóa:
 - Giải thuật được công bố rộng rãi
 - Chỉ có "khóa" là bí mật
- ❑ Mã hóa khóa đối xứng
 - Sử dụng một khóa
- ❑ Mã hóa khóa công khai
 - Sử dụng hai khóa
- ❑ Các hàm băm
 - Không sử dụng khóa
 - Không có gì bí mật: Làm sao để có thể hữu dụng?

Mã hóa khóa đối xứng



mã hóa **khóa đối xứng**: Bob và Alice chia sẻ cùng một khóa (đối xứng): K

□ vd: khóa là một mẫu S thay thế đã biết trong mã hóa thay thế một kí tự

H: làm sao để Bob và Alice có thể thống nhất về khóa?

Mã hóa khóa đối xứng: DES

DES: Chuẩn mã hóa dữ liệu (Data Encryption Standard)

- ❑ chuẩn mã hóa US [NIST 1993]
- ❑ khóa đối xứng 56-bit, dữ liệu đầu vào 64-bit
- ❑ mã hóa Khối với chuỗi mã hóa khối (CBC)
- ❑ DES an toàn như thế nào?
 - Thử thách của DES: khóa mã hóa 56-bit bị giải mã (vét cạn) trong t/gian ít hơn 1 ngày
 - Chưa có kiểu tấn công phân tích nào mạnh
- ❑ tăng độ an toàn cho DES:
 - 3DES: mã hóa DES 3 lần với 3 khóa khác nhau (Mã hóa, Giải mã, Mã hóa)

AES: Chuẩn mã hóa cao cấp (Advanced Encryption Standard)

- ❑ chuẩn khóa đối xứng mới (10/2001) của NIST, thay thế DES
- ❑ xử lý dữ liệu theo khối 128 bit
- ❑ khóa dài 128, 192 hoặc 256 bit
- ❑ giải mã vết cạn (thử từng khóa) tốn 1 giây với DES, tốn 149 tỉ tỉ năm với AES

Mã hóa khóa công khai

mã hóa khóa đối xứng

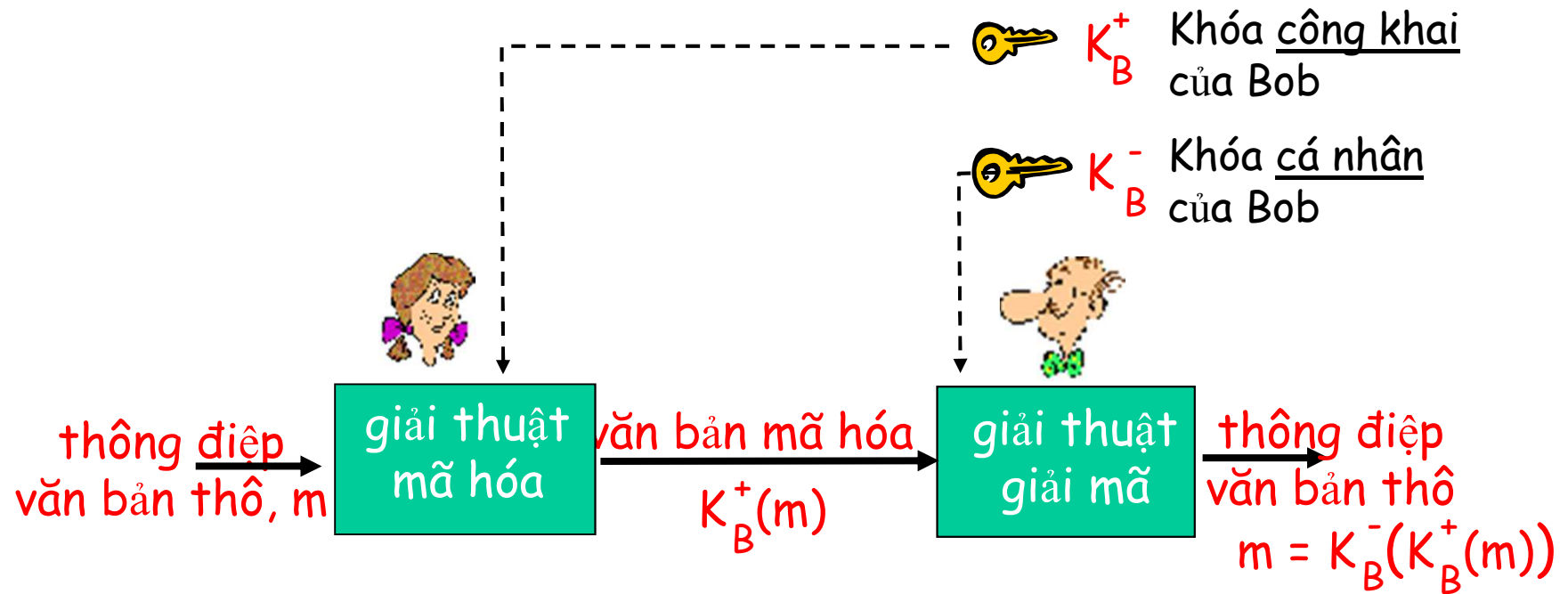
- yêu cầu n/gửi, nhận phải biết khóa bí mật
- H: làm sao để thống nhất (trao đổi) khóa bí mật từ đầu (nếu không gặp trực tiếp nhau)?

mã hóa khóa công khai

- một cách tiếp cận khác [Diffie-Hellman76, RSA78]
- n/gửi, nhận *không* chia sẻ khóa bí mật
- *Mọi người* đều biết khóa mã hóa *công khai*
- chỉ có n/nhận biết khóa giải mã *cá nhân*



Mã hóa khóa công khai



RSA: Tạo cặp khóa công khai/cá nhân

1. Chọn 2 số nguyên tố lớn p, q .
(e.g., 1024 bits each)
2. Tính $n = pq$, $z = (p-1)(q-1)$
3. Chọn e (với $e < n$) sao cho ko có ước chung nào với z .
(e, z là nguyên tố cùng nhau).
4. Chọn d sao cho $ed-1$ chia hết cho z .
(hay: $ed \bmod z = 1$).
5. *khóa công khai* ($\underbrace{n, e}_{K_B^+}$). *Khóa cá nhân* ($\underbrace{n, d}_{K_B^-}$).

RSA: Mã hóa, giải mã

0. Cho (n,e) và (n,d) như đã tính ở trước

1. Để mã hóa đoạn dữ liệu $m (< n)$, ta tính

$$c = m^e \bmod n$$

2. Để giải mã đoạn bit nhận được, c , ta tính

$$m = c^d \bmod n$$

Kết quả là!
$$m = \underbrace{(m^e \bmod n)}_c^d \bmod n$$

Ví dụ RSA:

Bob chọn $p=5$, $q=7$. Sau đó $n=35$, $z=24$.

$e=5$ (vì vậy e , z là nguyên tố cùng nhau).

$d=29$ (vì vậy $ed-1$ chia hết cho z).

mã hóa thông điệp 8-bit.

| | | | | |
|---------|----------------|-----------------------|-------------------------|-------------------------------------|
| mã hóa: | <u>mẫu bit</u> | <u>m</u> | <u>m^e</u> | <u>$c = m^e \bmod n$</u> |
| | 00001100 | 12 | 24832 | 17 |

| | | | |
|----------|-----------------------|--------------------------------------|-------------------------------------|
| giải mã: | <u>c</u> | <u>c^d</u> | <u>$m = c^d \bmod n$</u> |
| | 17 | 481968572106750915091411825223071697 | 12 |

Chương 8 Mục lục

8.1 Bảo mật mạng là gì?

8.2 Các nguyên lý của mật mã

8.3 Toàn vẹn thông điệp

8.4 Bảo vệ email

8.5 Bảo vệ kết nối TCP: SSL

8.6 Bảo mật tầng Mạng: IPsec

8.7 Bảo vệ mạng LAN không dây

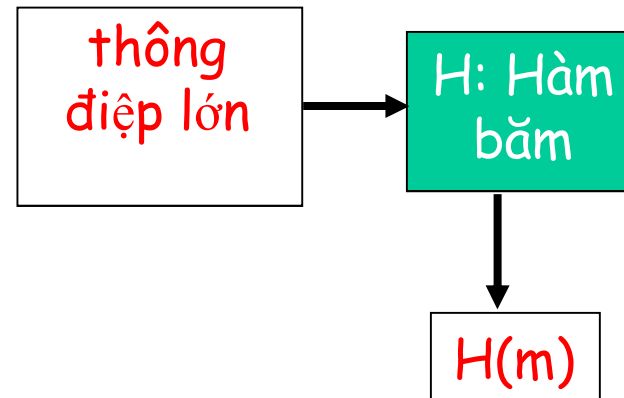
8.8 Bảo mật hành vi: tường lửa và IDS

Tính toàn vẹn thông điệp

- Cho phép các bên liên lạc xác minh rằng các tin nhắn nhận được được xác thực.
 - Nội dung thông điệp chưa bị thay đổi
 - Nguồn của thông điệp đúng là người gửi thật
 - Thông điệp chưa bị phát lại
 - Sự liên tục của thông điệp được duy trì
- Đầu tiên hãy xem xét “Sự chuyển hóa thông điệp”

Sự chuyển hóa thông điệp (message digest)

- Hàm $H()$ có tham số là một thông điệp có độ dài bất kì và xuất ra một chuỗi văn bản độ dài xác định: "kí hiệu nhận biết thông điệp"
- Chú ý rằng $H()$ là hàm nhiều-tới-1
- $H()$ thường được gọi là "hàm băm"

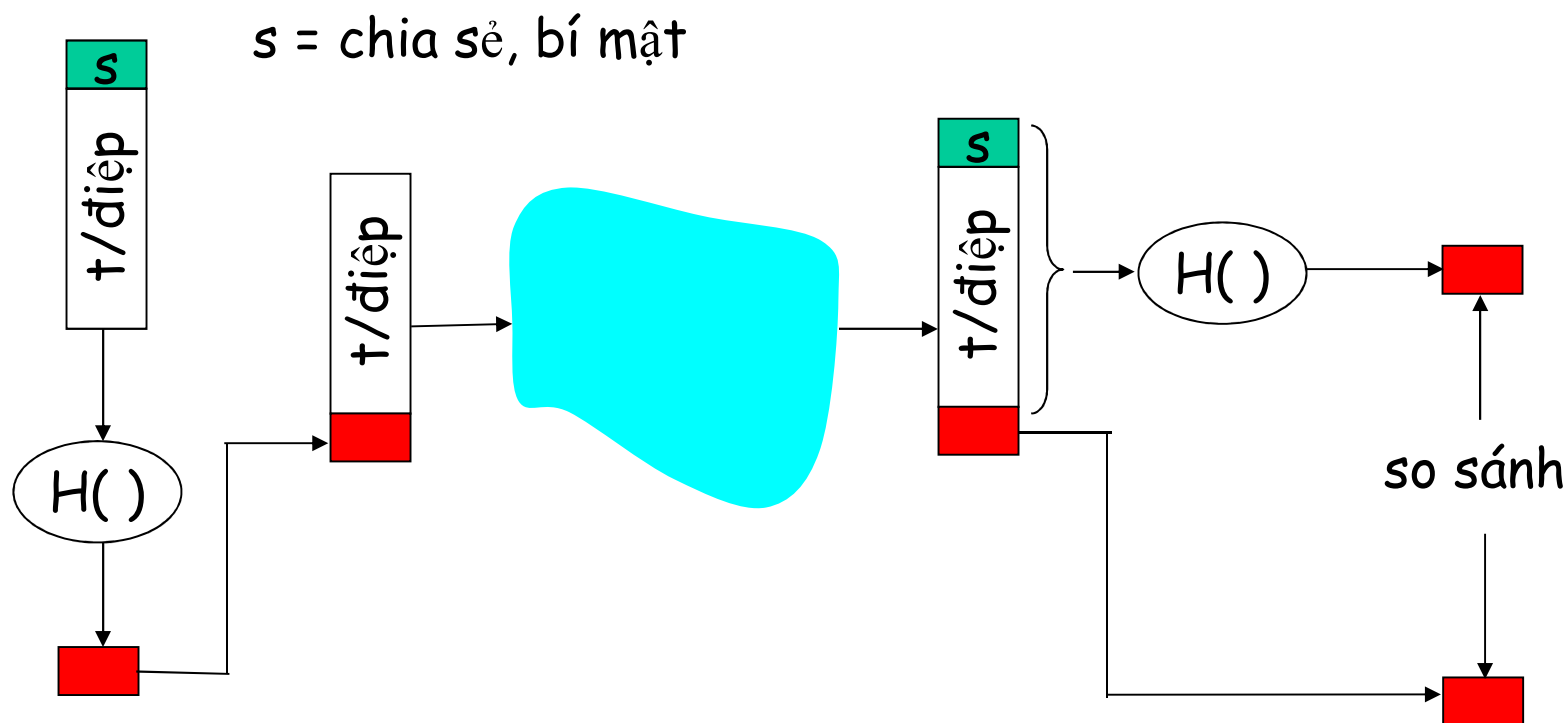


- Các tính chất cần thiết:
 - Dễ tính toán
 - Không thể tính ngược: không thể tính được m từ $H(m)$
 - Chống đụng độ: rất khó về phương diện tính toán để có thể tìm ra m và m' sao cho $H(m) = H(m')$
 - Kết quả gần ngẫu nhiên

Giải thuật Hàm Băm

- ❑ MD5 được sử dụng rộng rãi (RFC 1321)
 - tính ra giá trị băm 128-bit sau một quá trình 4 bước.
- ❑ SHA-1 cũng được dùng.
 - chuẩn của Mỹ [NIST, FIPS PUB 180-1]
 - giá trị băm 160-bit

Mã xác thực thông điệp(MAC)



- ❑ *Xác thực người gửi*
- ❑ *Kiểm tra tính toàn vẹn*
- ❑ Không mã hóa !
- ❑ Còn được gọi là "Băm có khóa"
- ❑ Chú thích: $MD_m = H(s||m)$; gửi $m||MD_m$

HMAC

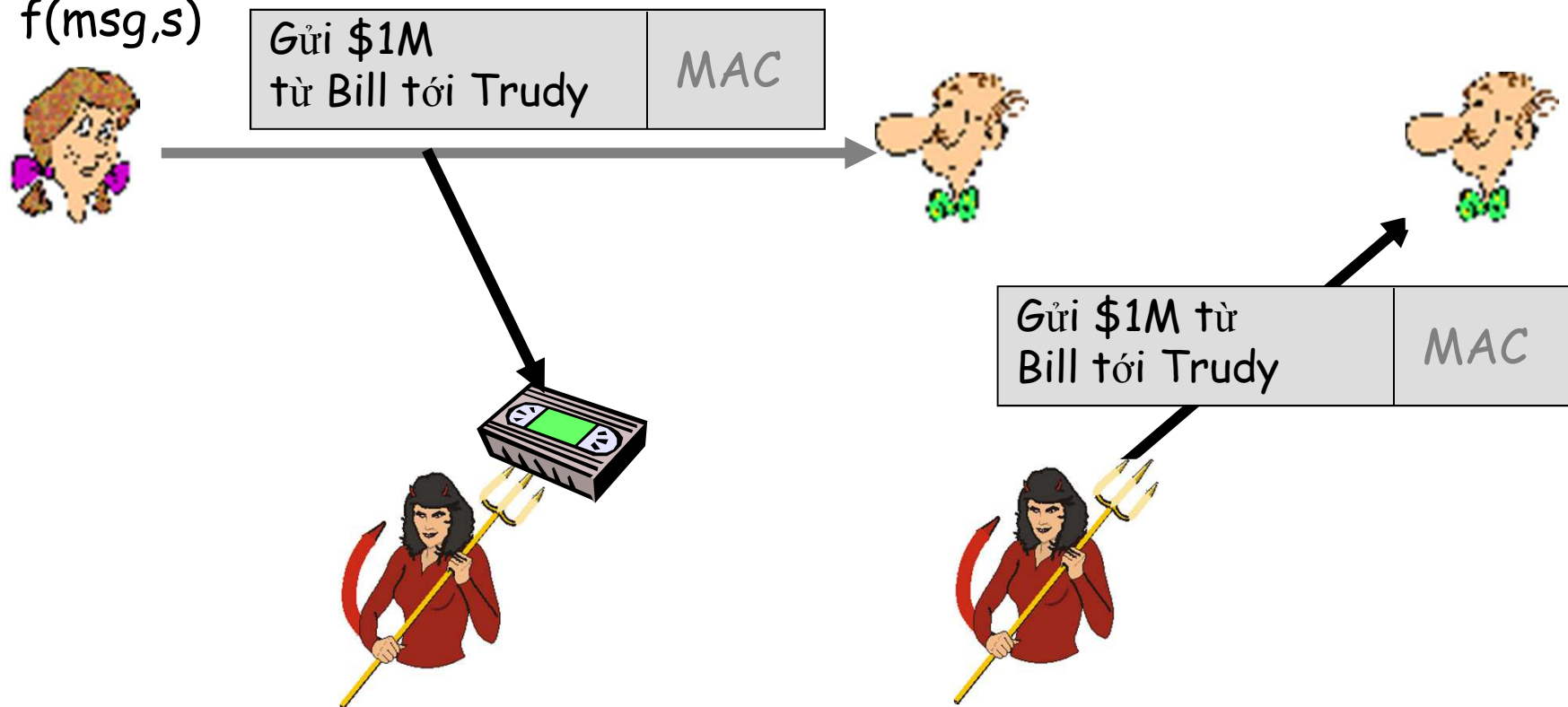
- ❑ Chuẩn MAC phổ biến
 - ❑ Nhắm vào những lỗ hổng bảo mật quan trọng
1. Gắn khóa bí mật vào đầu thông điệp
 2. Băm thông điệp đã được nối
 3. Gắn khóa bí mật vào đầu của mã băm
 4. Băm mã một lần nữa.

Xác thực đầu-cuối

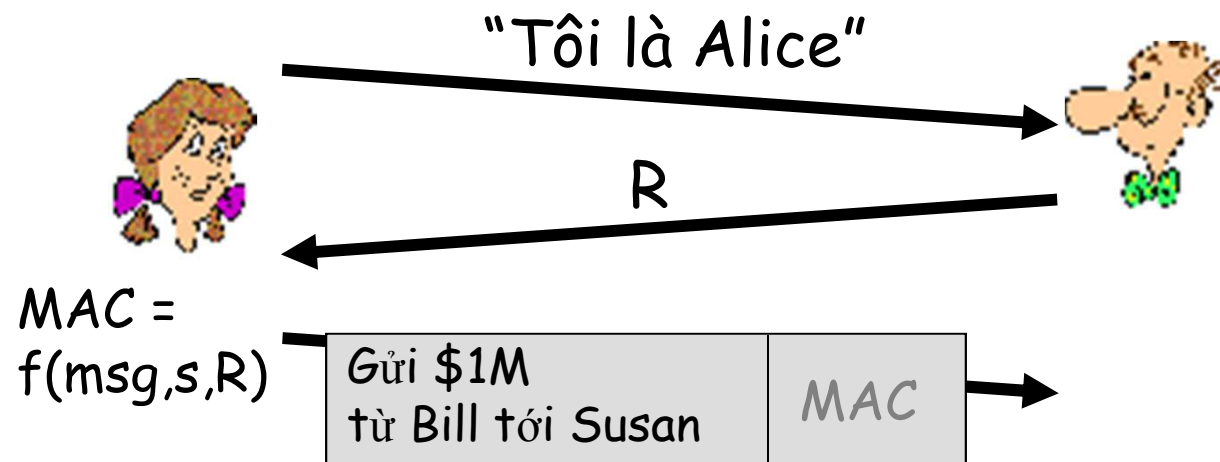
- ❑ Muốn chắc chắn về người gửi thông điệp - *xác thực đầu-cuối*.
- ❑ Giả sử Alice và Bob có một bí mật chia sẻ, liệu MAC có cung cấp sự xác thực đầu cuối không?
 - Ta biết được là Alice tạo ra thông điệp.
 - Nhưng có đúng là cô ta gửi nó đi không?

Tấn công "Phát lại"

MAC =
 $f(msg, s)$



Phòng chống tấn công phát-lại: chỉ-một-lần



Chữ kí số

Là kĩ thuật mật mã tương tự như chữ kí viết tay.

- ❑ n/gửi (Bob) kí số vào văn bản, chứng minh rằng anh ta là người tạo ra văn bản.
- ❑ Mục đích tương tự như MAC, ngoại trừ việc sử dụng mật mã khóa công khai
- ❑ có thể xác minh, ko thể làm giả: n/nhận (Alice) có thể chứng minh rằng chỉ có Bob, mà không ai khác (kể cả Alice), đã kí vào văn bản


Chữ kí số

Chữ kí số đơn giản cho thông điệp m :

- Bob kí vào m bằng cách mã hóa với khóa cá nhân của anh ta K_B , tạo ra thông điệp "đã kí", $K_B(m)$

t/điệp của Bob, m

Dear Alice
Oh, how I have missed
you. I think of you all the
time! ... (blah blah blah)
Bob

 K_B^- Khóa cá nhân
của Bob

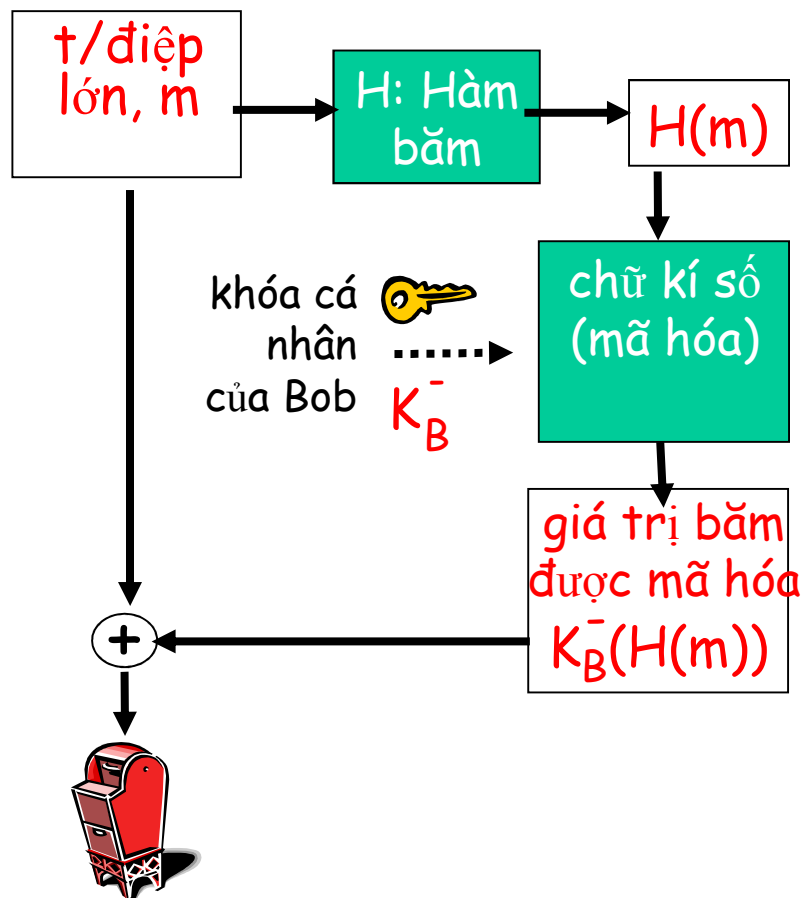
giải thuật
mã hóa
khóa c/khai

$K_B^-(m)$

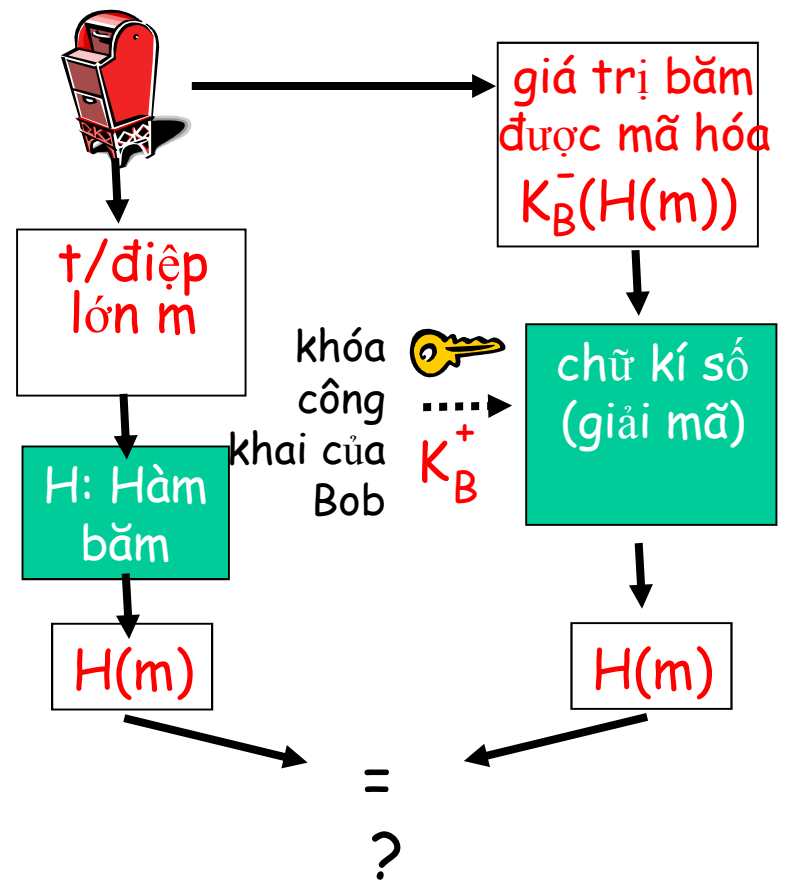
t/điệp của Bob, m ,
đã được kí (mã
hóa) với khóa cá
nhân của anh ta

Chữ kí số = chuỗi băm thông điệp được kí

Bob gửi thông điệp được kí số:



Alice kiểm tra chữ kí và sự toàn vẹn của thông điệp được kí số:



Chữ kí số (++)

- Giả sử Alice nhận t/điệp m, chữ kí số $K_B^-(m)$
- Alice kiểm tra m kí bởi Bob: giải mã $K_B^-(m)$ bằng khóa công khai của Bob K_B^+ , kiểm tra xem $K_B^+(K_B^-(m)) = m$.
- Nếu $K_B^+(K_B^-(m)) = m$, thì người kí vào m phải có khóa cá nhân của Bob.

Alice bằng cách đó có thể kiểm tra:

- Bob đã kí vào m.
- Ko ai khác kí vào m.
- Bob kí vào m mà không phải m'.

Không-thoái-thác:

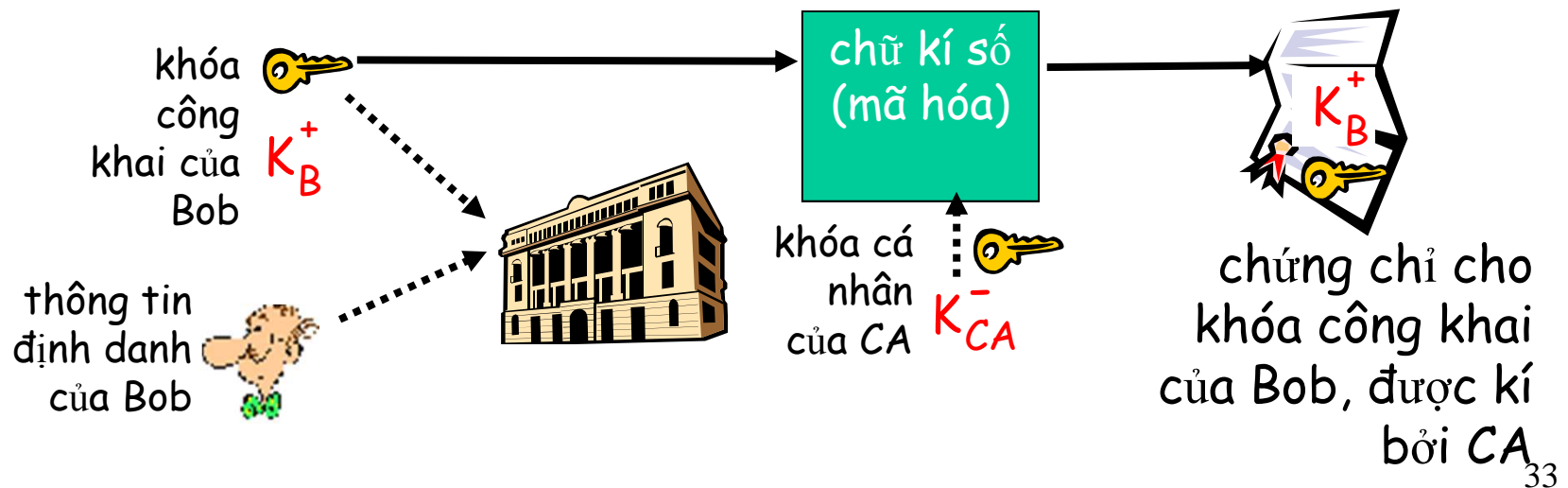
- ✓ Alice có thể lấy m, và chữ kí $K_B^-(m)$ tới tòa án và chứng minh rằng Bob kí vào m.

Sự chứng nhận khóa-Công khai

- ❑ Động cơ: Trudy muốn chơi xỏ Bob với tiệm bánh pizza
 - Trudy tạo một email đặt hàng:
Xin chào, làm ơn đem cho tôi 4 bánh pizza pepperoni. Cảm ơn, Bob
 - Trudy kí vô đơn đặt hàng với khóa cá nhân của cô
 - Trudy gửi đơn đặt hàng tới cửa hàng Pizza
 - Trudy gửi tới cửa hàng Pizza khóa công khai của cô, nhưng nói rằng đó là khóa công khai của Bob.
 - Cửa hàng Pizza kiểm tra chữ kí; sau đó giao cho Bob 4 bánh pizza.
 - Bob hoàn toàn ko biết gì.

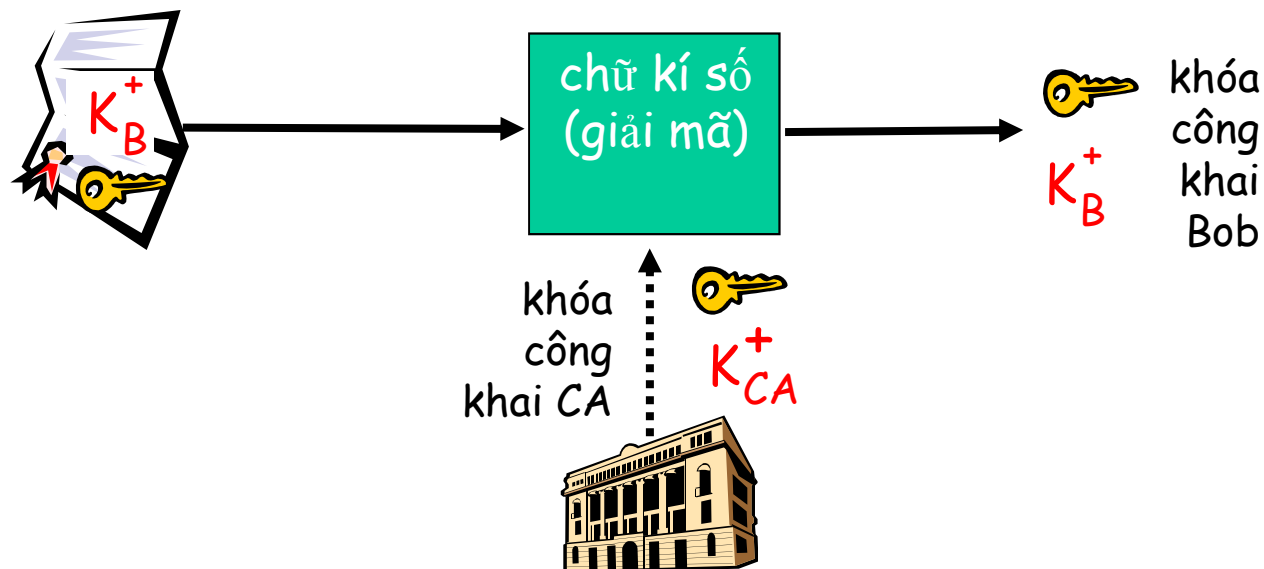
Các nhà có thẩm quyền chứng nhận

- ❑ **Nhà thẩm quyền chứng nhận (CA):** liên kết khóa công khai tới một thực thể cụ thể E.
- ❑ E (người, BĐT) đăng kí khóa công khai của nó với CA.
 - E cung cấp "bằng chứng định danh" cho CA.
 - CA tạo ra chứng chỉ mà liên kết E với khóa công khai của nó.
 - chứng chỉ chứa khóa công khai của E được kí số bởi CA - CA nói "đây là khóa công khai của E"



Các nhà có thẩm quyền chứng nhận

- Khi Alice muốn có khóa công khai của Bob:
 - lấy chứng chỉ của Bob (từ Bob hoặc ai khác).
 - dùng khóa công khai của CA giải mã chứng chỉ của Bob, lấy khóa công khai của Bob



Chứng chỉ: tóm tắt

- ❑ Chuẩn nguyên thủy X.509 (RFC 2459)
- ❑ Chứng chỉ chứa:
 - Tên người phát hành
 - Tên, địa chỉ, tên miền, v.v.. của thực thể.
 - Khóa công khai của thực thể
 - Chữ kí số (kí với khóa cá nhân của ng phát hành)
- ❑ Cơ sở hạ tầng khóa công khai (PKI)
 - Chứng chỉ và các nhà có thẩm quyền chứng nhận
 - Thường bị xem là “nặng nề”

Chương 8 Mục lục

8.1 Bảo mật mạng là gì?

8.2 Các nguyên lý của mật mã

8.3 Toàn vẹn thông điệp

8.4 Bảo vệ email

8.5 Bảo vệ kết nối TCP: SSL

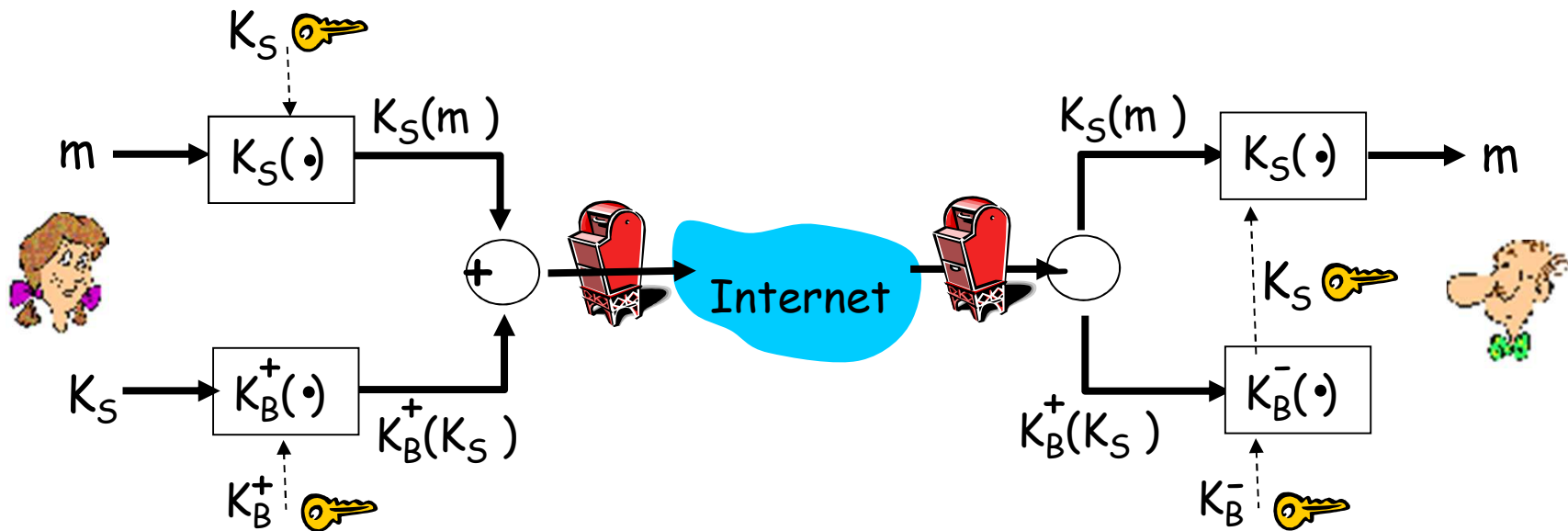
8.6 Bảo mật tầng Mạng: IPsec

8.7 Bảo vệ mạng LAN không dây

8.8 Bảo mật hành vi: tường lửa và IDS

Bảo mật e-mail

- Alice muốn gửi email bí mật, m , cho Bob.

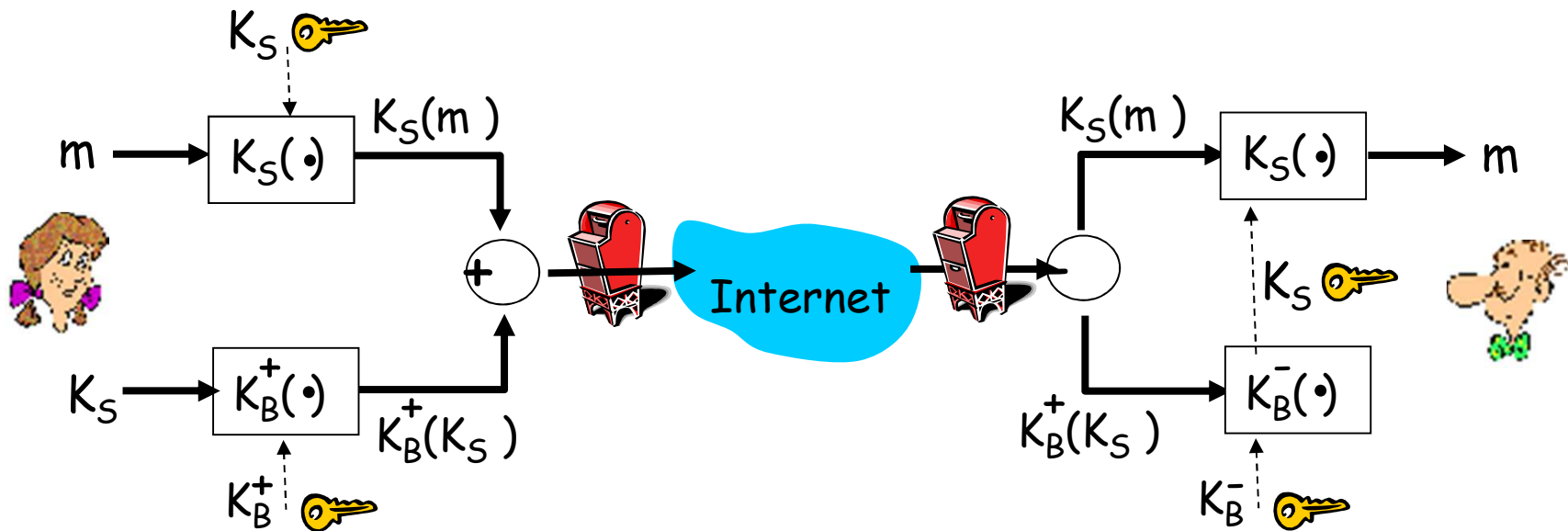


Alice:

- sinh ngẫu nhiên khóa cá nhân *đối xứng*, K_S .
- mã hóa t/điệp với K_S (tăng hiệu suất)
- đồng thời mã hóa K_S với khóa công khai của Bob.
- gửi cả hai $K_S(m)$ và $K_B(K_S)$ cho Bob.

Bảo mật e-mail (tt)

- Alice muốn gửi email bí mật, m , cho Bob.

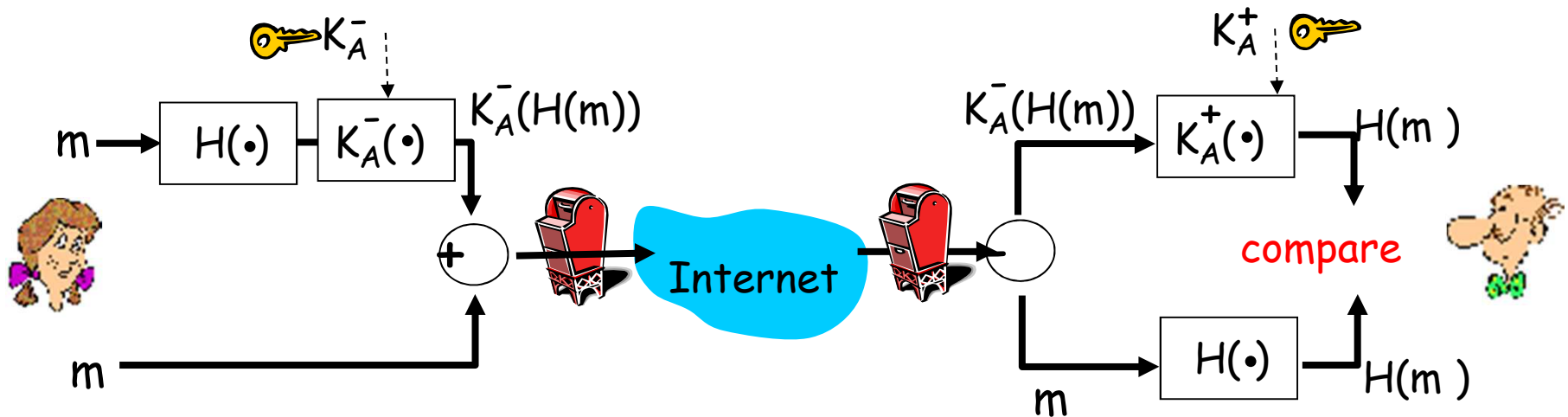


Bob:

- sử dụng khóa cá nhân của anh để giải mã và lấy được K_S
- sử dụng K_S để giải mã $K_S(m)$ để lấy được m

Bảo mật e-mail (tt)

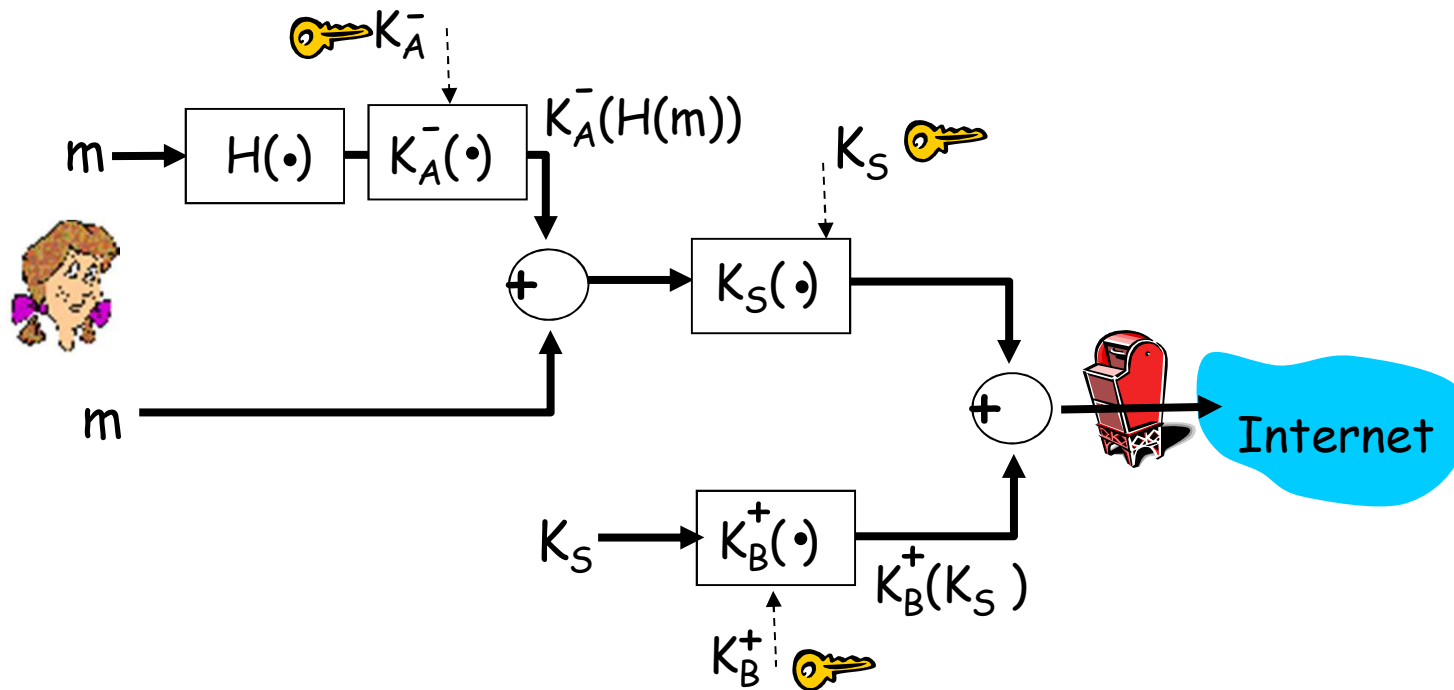
- Alice muốn cung cấp sự xác thực người gửi, tính toàn vẹn thông điệp.



- Alice kí số vào thông điệp.
- gửi cả thông điệp (chưa mã hóa) và chữ kí số.

Bảo mật e-mail (tt)

- Alice muốn cung cấp tính bí mật, sự xác thực người gửi, tính toàn vẹn thông điệp.



Alice sử dụng 3 khóa: khóa cá nhân của cô ta, khóa công khai của Bob, khóa đối xứng vừa tạo ra

Chương 8 Mục lục

8.1 Bảo mật mạng là gì?

8.2 Các nguyên lý của mật mã

8.3 Toàn vẹn thông điệp

8.4 Bảo vệ email

8.5 Bảo vệ kết nối TCP: SSL

8.6 Bảo mật tầng Mạng: IPsec

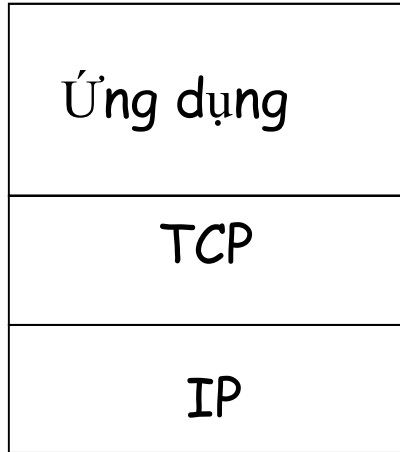
8.7 Bảo vệ mạng LAN không dây

8.8 Bảo mật hành vi: tường lửa và IDS

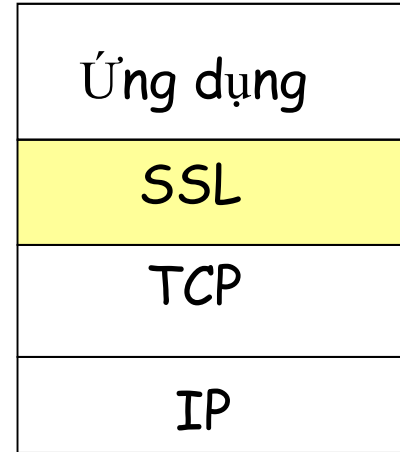
SSL: Secure Sockets Layer

- Giao thức bảo mật được triển khai rộng rãi
 - được hỗ trợ bởi hầu hết các trình duyệt và máy chủ web
 - https
 - hàng chục tỉ \$ được sử dụng hàng năm qua SSL
- Thiết kế bởi Netscape vào 1993
- Có vài biến đổi:
 - TLS: transport layer security, RFC 2246
- Cung cấp:
 - Bí mật
 - Toàn vẹn
 - Xác thực
- Các mục tiêu ban đầu:
 - Có giao dịch thương mại điện tử
 - Mã hóa (đặc biệt là số thẻ-tín dụng)
 - xác thực máy chủ Web
 - xác thực khách (tùy chọn)
 - Hạn chế thủ tục khi mà buôn bán với bạn hàng mới
- Có sẵn trong tất cả ứng dụng TCP
 - giao diện hóc kết nối an toàn (Secure socket interface)

SSL and TCP/IP



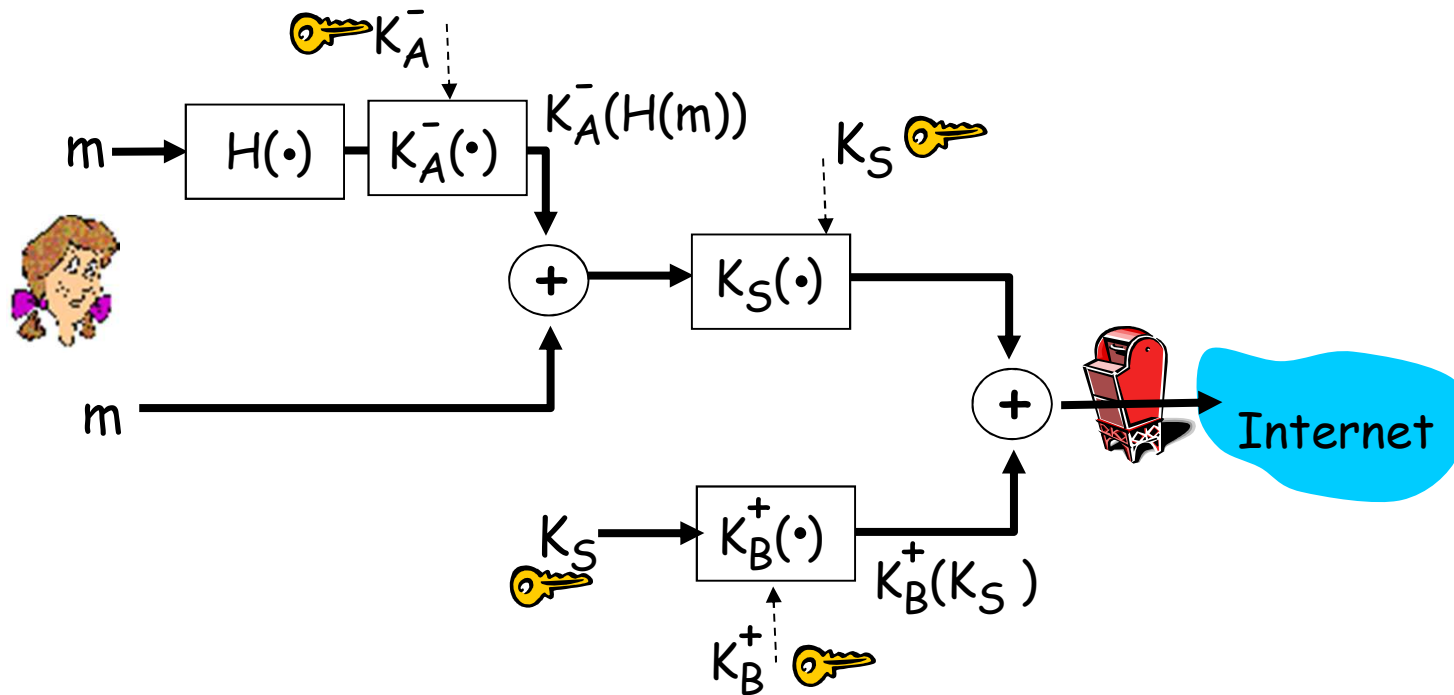
Ứng dụng th/thường



Ứng dụng
với SSL

- SSL cung cấp giao diện lập trình ứng dụng (API) cho ứng dụng
- các thư viện/lớp SSL trong C và Java đã có sẵn

Quá trình làm việc:



- Nhưng cần gửi luồng byte và dữ liệu tương tác
- Cần một bộ các khóa bí mật cho toàn bộ kết nối
- Cần phân trao đổi chứng chỉ của giao thức:
pha bắt-tay

Chương 8 Mục lục

8.1 Bảo mật mạng là gì?

8.2 Các nguyên lý của mật mã

8.3 Toàn vẹn thông điệp

8.4 Bảo vệ email

8.5 Bảo vệ kết nối TCP: SSL

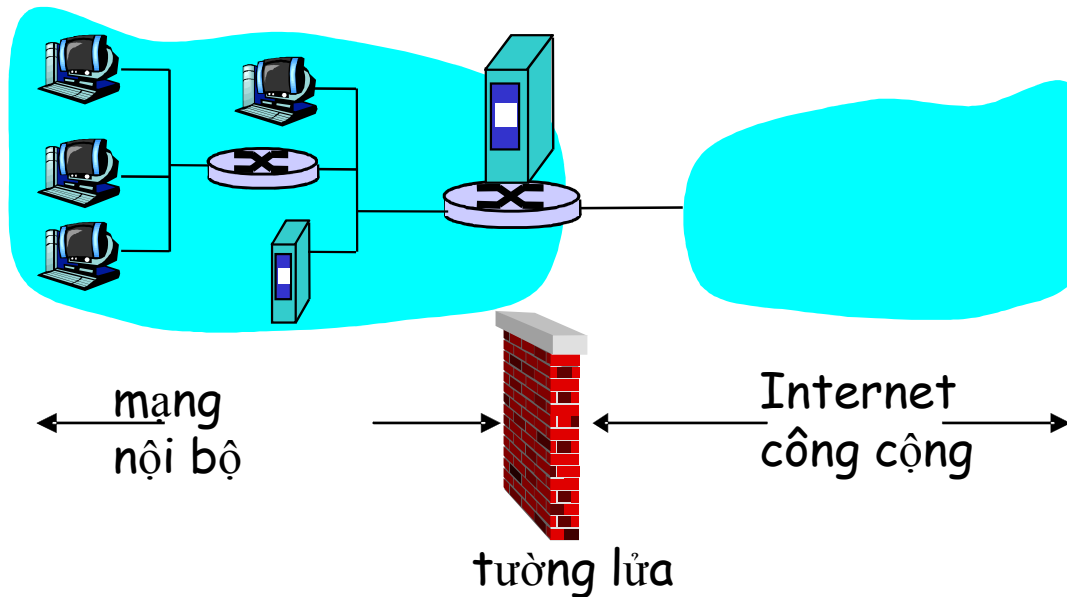
8.6 Bảo mật tầng Mạng: IPsec

8.8 Bảo mật hành vi: tường lửa và IDS

Tường lửa

tường lửa

cách li mạng bên trong tổ chức với mạng Internet, cho phép vài gói tin đi qua, chặn những gói khác.



Tường lửa: Để làm gì?

ngăn chặn tấn công từ chối dịch vụ:

- Sự gửi tràn SYN: kẻ tấn công thiết lập nhiều kết nối TCP giả , ko còn tài nguyên cho những kết nối "thật"

ngăn chặn sự truy cập/thay đổi không hợp pháp vào dữ liệu nội bộ.

- vd: kẻ tấn công thay đổi trang chủ của công ty

chỉ cho phép những truy cập được xác thực vào bên trong mạng (nhóm các n.dùng, máy đã được xác thực)

ba loại tường lửa:

- bộ lọc gói không trạng thái
- bộ lọc gói trạng thái
- cổng kiểm soát ứng dụng

Hệ thống phát hiện xâm nhập

□ sự lọc gói:

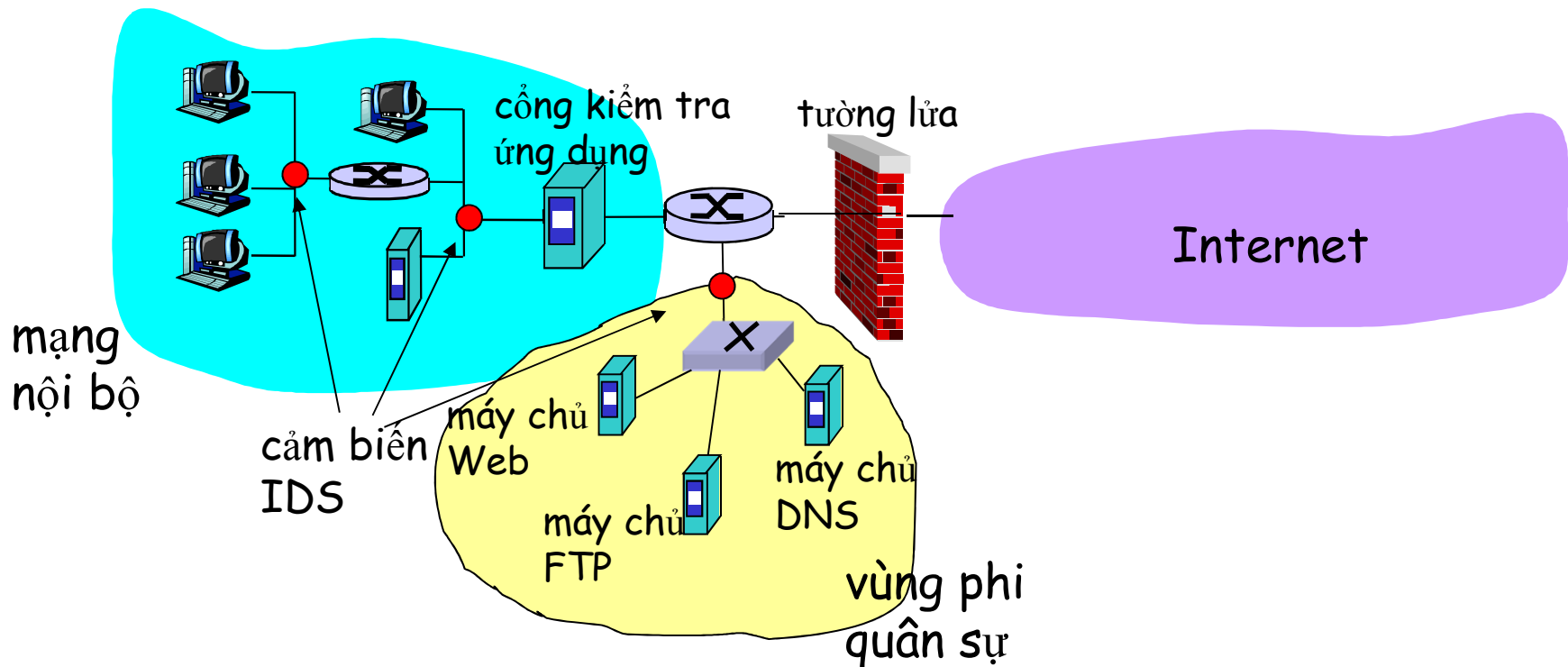
- chỉ làm việc với mào đầu TCP/IP
- ko kiểm tra sự tương quan giữa các phiên

□ *IDS: hệ thống phát hiện xâm nhập* (intrusion detection system)

- *Kiểm tra gói sâu*: xem xét nội dung gói tin (vd: kiểm tra chuỗi kí tự trong gói tin, so sánh với cơ sở dữ liệu của vi-rút, chuỗi tấn công)
- xem xét mối tương quan giữa nhiều gói tin
 - sự dò công
 - ánh xạ mạng
 - tấn công DoS

Hệ thống phát hiện xâm nhập

- nhiều IDS: nhiều loại kiểm tra khác nhau tại nhiều vị trí khác nhau



Bảo mật mạng (tổng kết)

Kỹ thuật cơ bản.....

- mã hóa (đối xứng hoặc công khai)
- toàn vẹn thông điệp
- xác thực đầu cuối

.... sử dụng trong nhiều kịch bản bảo mật

- email an toàn
- truyền tải an toàn (SSL)
- 802.11

Bảo mật hành vi: tường lửa và IDS