15 User Authentication

Ch15 in textbook

Yanwei Yu

E-mail: ywyu@ustc.edu.cn

Outline

- Objectives, Applications, Classification of Identification
- Passwords
- Challenge-response Identification
- Customized and Zero-knowledge Identification Protocols
- Attacks on Identification Protocols

Outline

- Objectives, Applications, Classification of Identification
- Passwords
- Challenge-response Identification
- Customized and Zero-knowledge Identification Protocols
- Attacks on Identification Protocols

Definition of Identification

- **Entity authentication: is the process whereby one party(*verifier*) is assured of the identity of a second party(*claimant*) involved in a protocol, and that the second has actually participated.**

Identification vs. Message Authentication

Identification	**Message authentication**
• **Verify** a claimant's identity through actual communications in real-time	• **Verify** original of message which itself provides no timeliness(及时) guarantees with respect to when a message was created
• typically involves no meaningful message	• involves meaningful messages

Identification vs. Digital signature

Identification

- The semantics of the message are essentially fixed.
- The claim is either corroborated(证实) or rejected immediately, with associated privileges or access either granted or denied in real time.

Digital Signature

- involve a variable message
- typically provide a non-repudiation allowing disputes to be resolved by judges after the fact

In some cases, identification schemes may also be converted to signature schemes.

Objectives of identification protocols

- For honest parties A and B, A is able to successfully authenticate(证明) itself to B, i.e., B will complete the protocol having accepted A’s identity
- (No transferability) B cannot reuse an identification exchange with A so as to successfully impersonate A to a third party C.
- (No impersonation) The probability of successful impersonation is negligible.

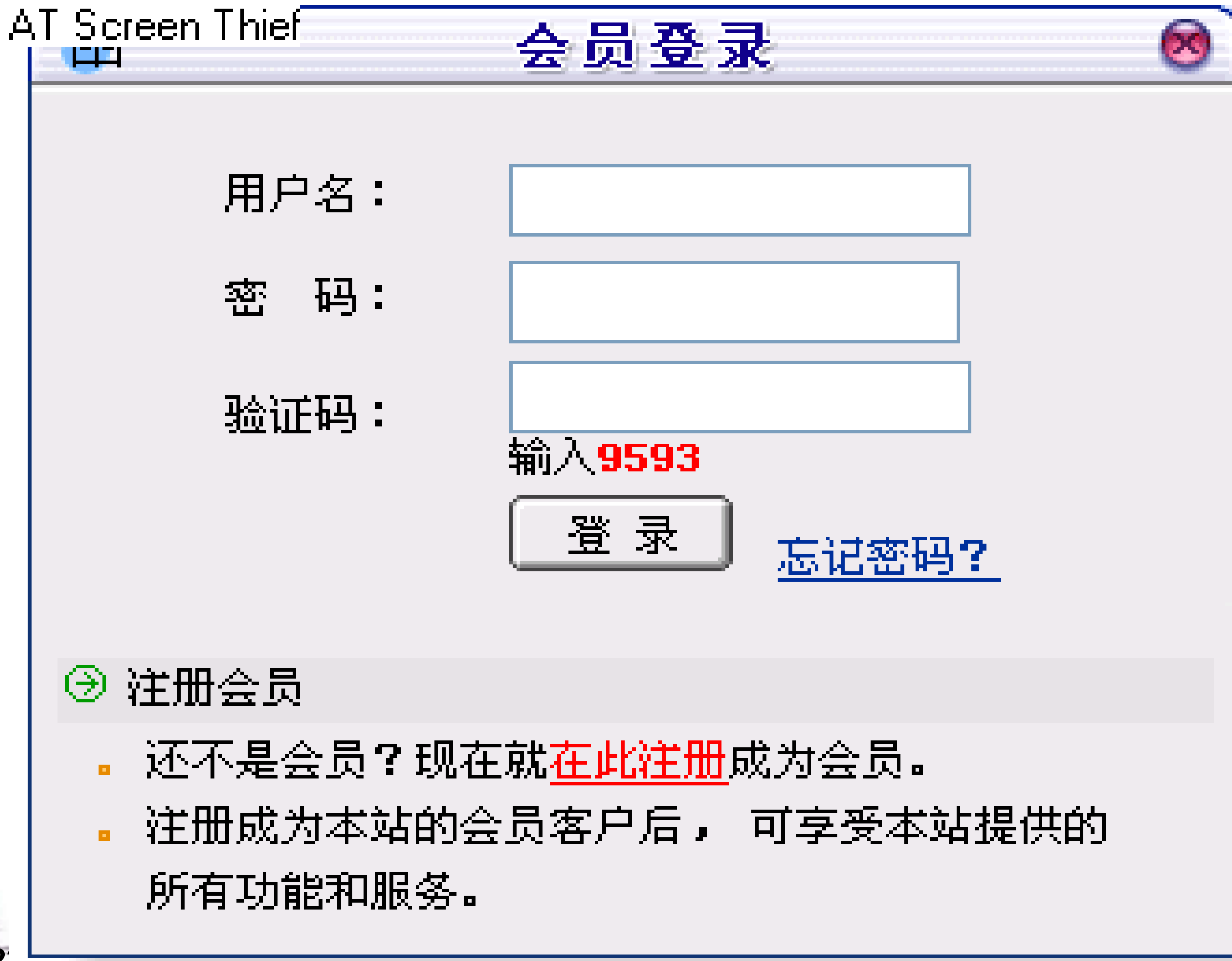
Applications of Identification

- **facilitate access control to a restricted resource, E.g., <u>access control matrix</u>**
- **track identified entities**
  - **E.g., billing of cellular telephony**
- **Used for key establishment protocols**

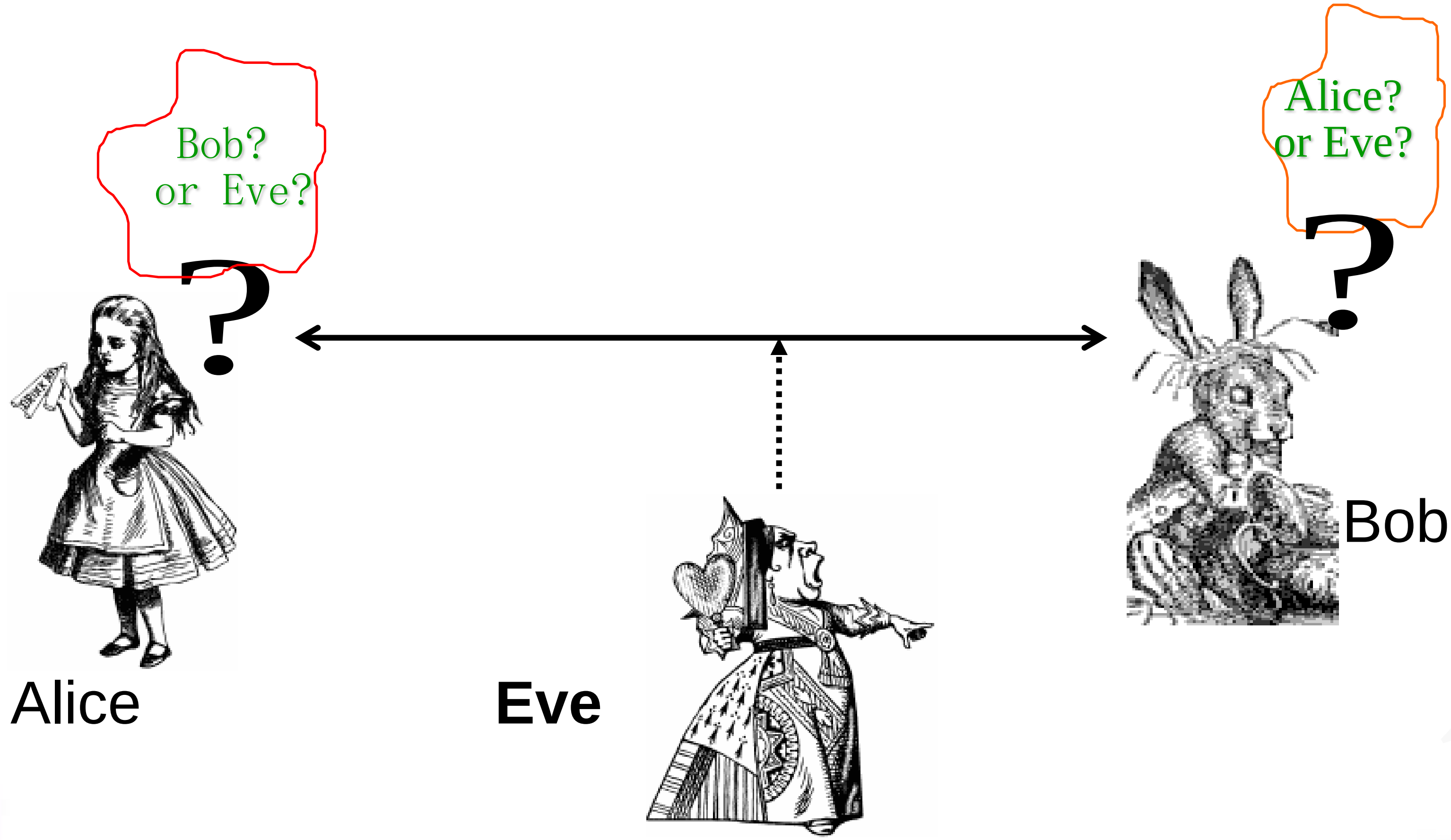
	Program1	. . .	SegmentA	SegmentB
Process1	Read Execute		Read Write	
Process2				Read
⋮				

(a) Access matrix

Example 1: Web Login



Example 2: Secure Communication



Basis of Identification

- **something known.**
  - E.g., passwords, Personal Identification Numbers (PINs), and the secret or private keys.
- **something possessed:** typically a physical accessory.
  - E.g., magnetic-striped cards, *chipcards*, and hand-held customized calculators (*password generators*) which provide time-variant passwords.
- **something inherent** (to a human individual): use of human physical characteristics and involuntary actions (*biometrics*)
  - E.g., handwritten signatures, fingerprints, voice, retinal patterns, hand geometries, and dynamic keyboarding characteristics. (not discussed further here)

Classification of Identification

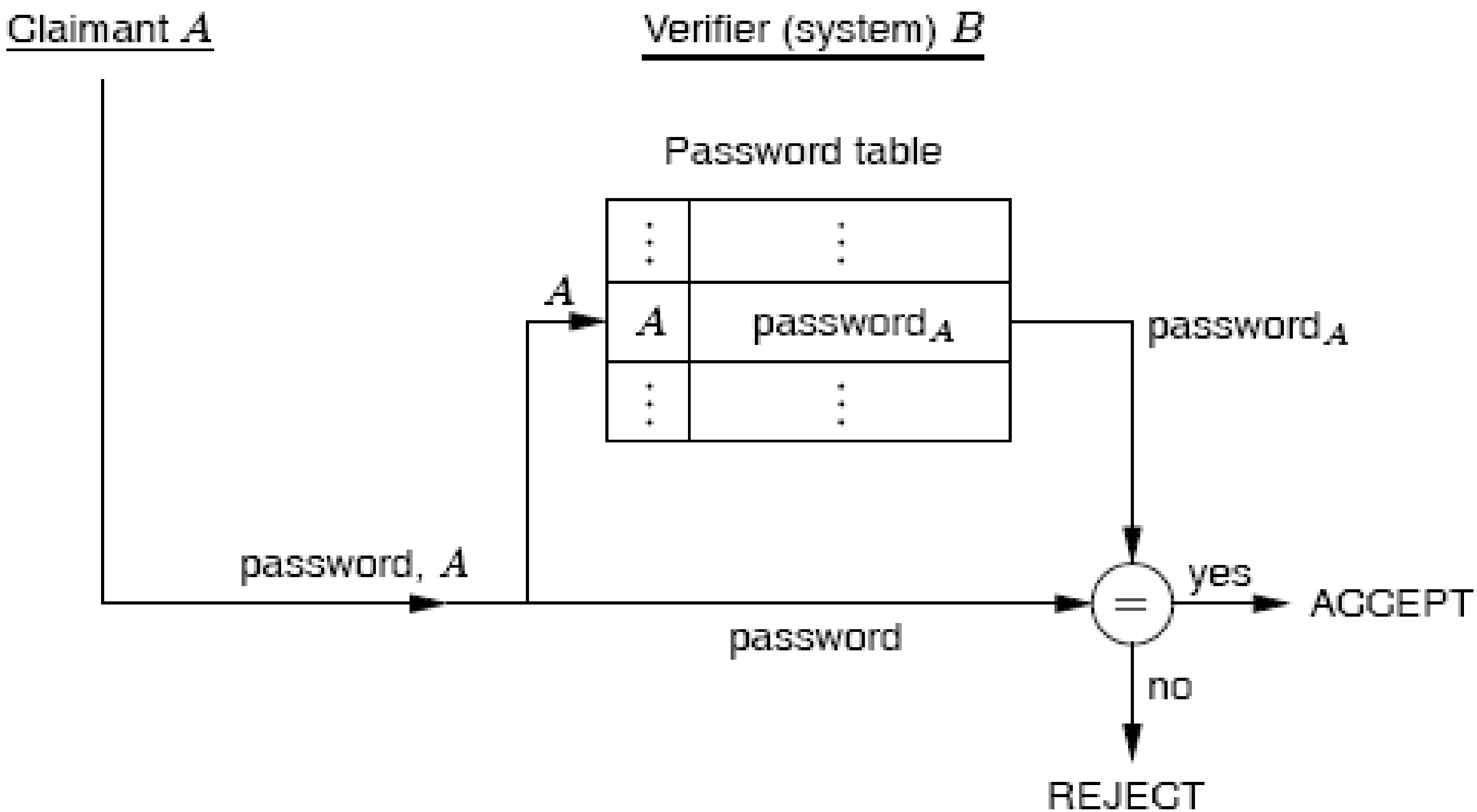
- **Passwords (weak authentication):** The system checks Whether the password matches corresponding data for that userid for access control of the resource.
  - userid is a claim of identity
  - password is the evidence supporting the claim.
- **Strong authentication:** The claimant proves its identity to the verifier by demonstrating knowledge of a secret known to be associated with that claimant, without revealing the secret itself to the verifier during the protocol.
  - **Challenge-response identification: using encryption**
  - **Customized and Zero-knowledge Identification Protocols: using zero-knowledge**

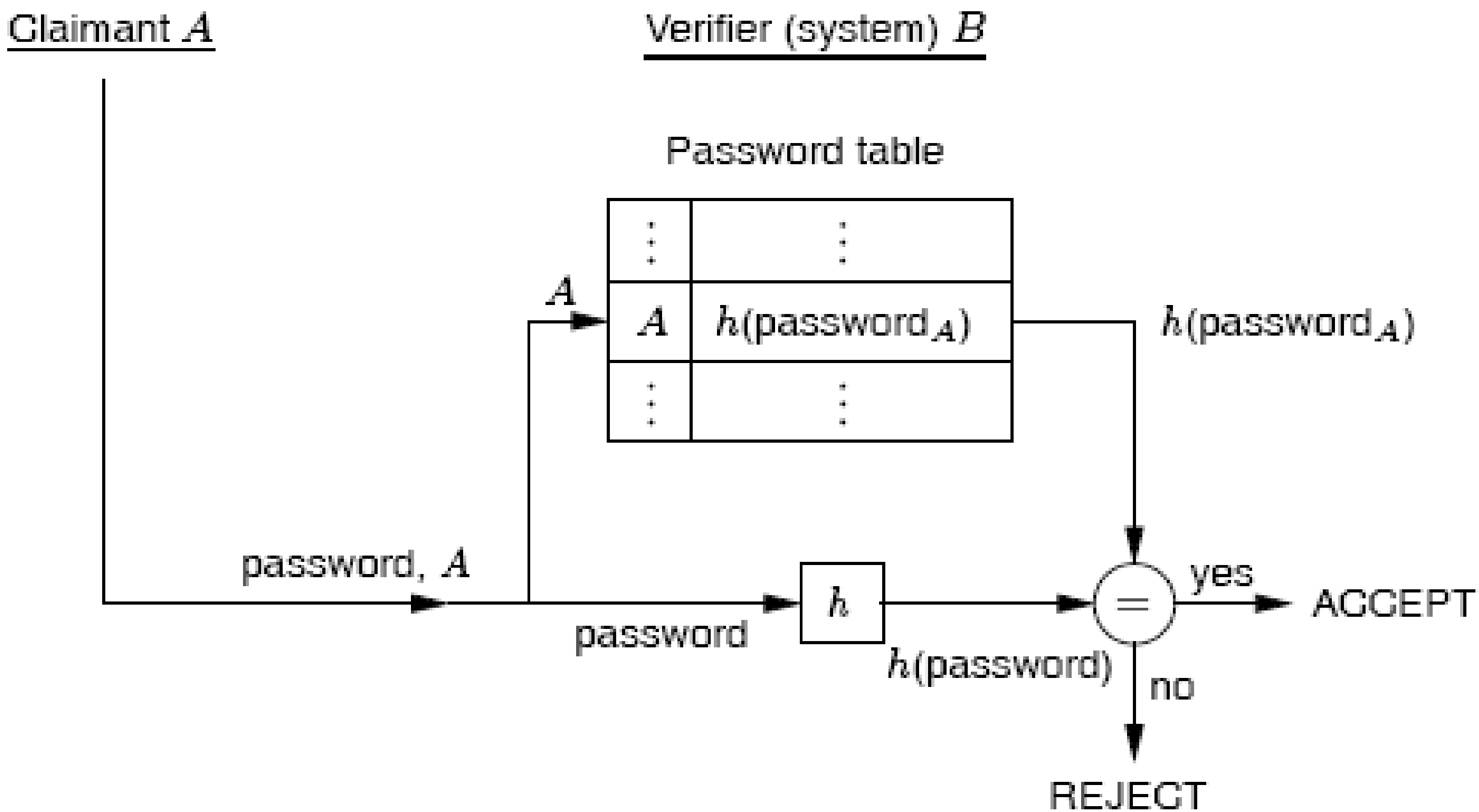
Outline

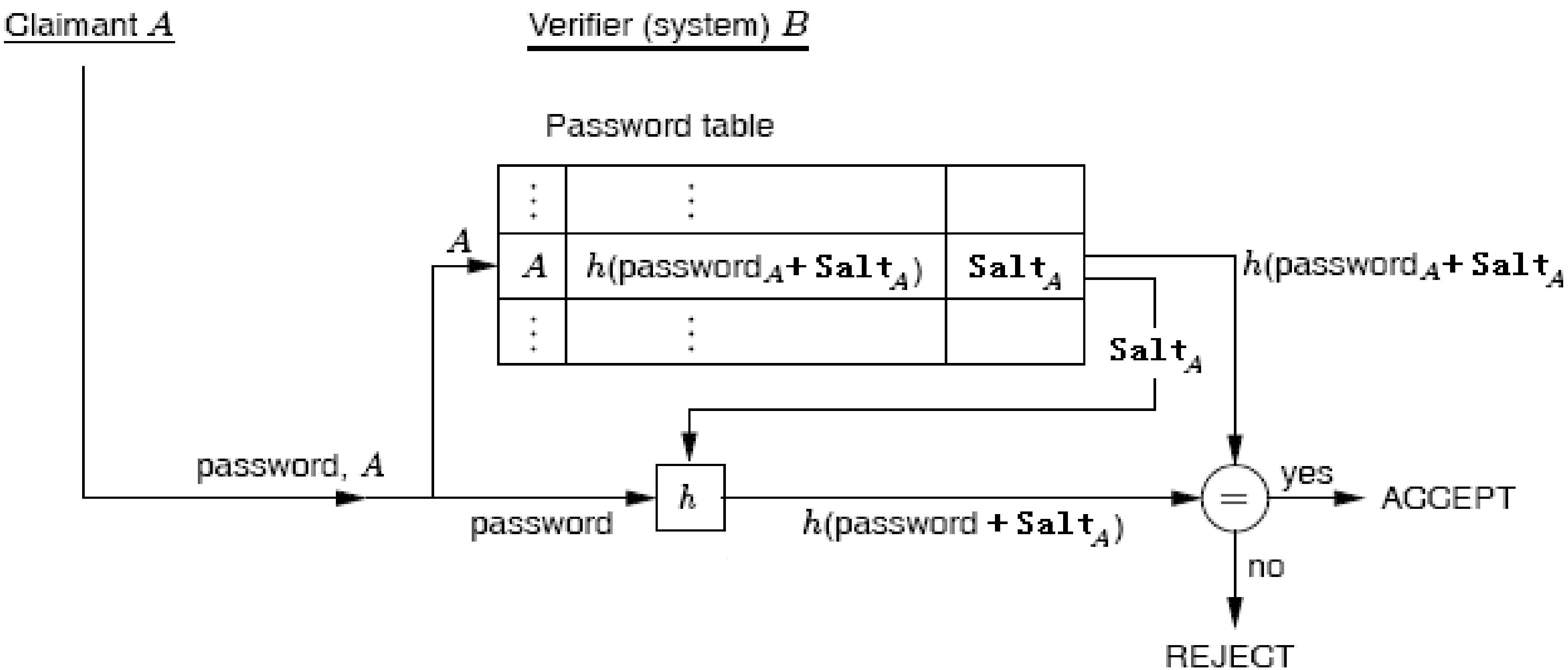
- Objectives, Applications, Classification of Identification
- Passwords
- Challenge-response Identification
- Customized and Zero-knowledge Identification Protocols
- Attacks on Identification Protocols

Fixed Password

- Stored password files
  - store user passwords clear-text
  - system password file needs both read- and write-protected (e.g., via operating system access control privileges).
- “Encrypted” password files
  - Store a one-way function of each user password
  - the password file need now only be write-protected


Claimant A
Verifier (system) B
Password table
A
A
password$_A$
password$_A$
password, A
password
yes
ACCEPT
no
REJECT


Claimant A
Verifier (system) B
Password table
A
A
$h(\text{password}_A)$
$h(\text{password}_A)$
password, A
password
h
$h(\text{password})$
=
yes
ACCEPT
no
REJECT


Claimant A
Verifier (system) B
Password table
A
A
h(password$_A$+ Salt$_A$)
Salt$_A$
h(password$_A$+ Salt$_A$)
Salt$_A$
password, A
password
h
h(password + Salt$_A$)
=
yes
ACCEPT
no
REJECT

Attacks and Countermeasures of Fixed Password

Attack

- Replay of fixed passwords
- Exhaustive password search
- Password-guessing and dictionary attacks

Countermeasures

- Password rules
  - Long password
  - Not using “weak” passwords
- password aging
- Slowing down the password mapping
- Salting passwords
- Passphrases
- restrict the number of times guessing password

数缘社区 » 注册

必填信息	
用户名* 不能有空格，可以是中文，长度控制在 3 - 12 字节以内	检查用户名
密码* 英文字母或数字等不少于6位	
确认密码*	
认证码*	87 50 请将图片中的数字或英文字母填入到文本框中
Email*	
论坛防恶意注册* 请输入答案： 妈妈的弟弟叫什么？	
注册原因*	

选填信息	
QQ	

提 交

Case Study – UNIX Passwords

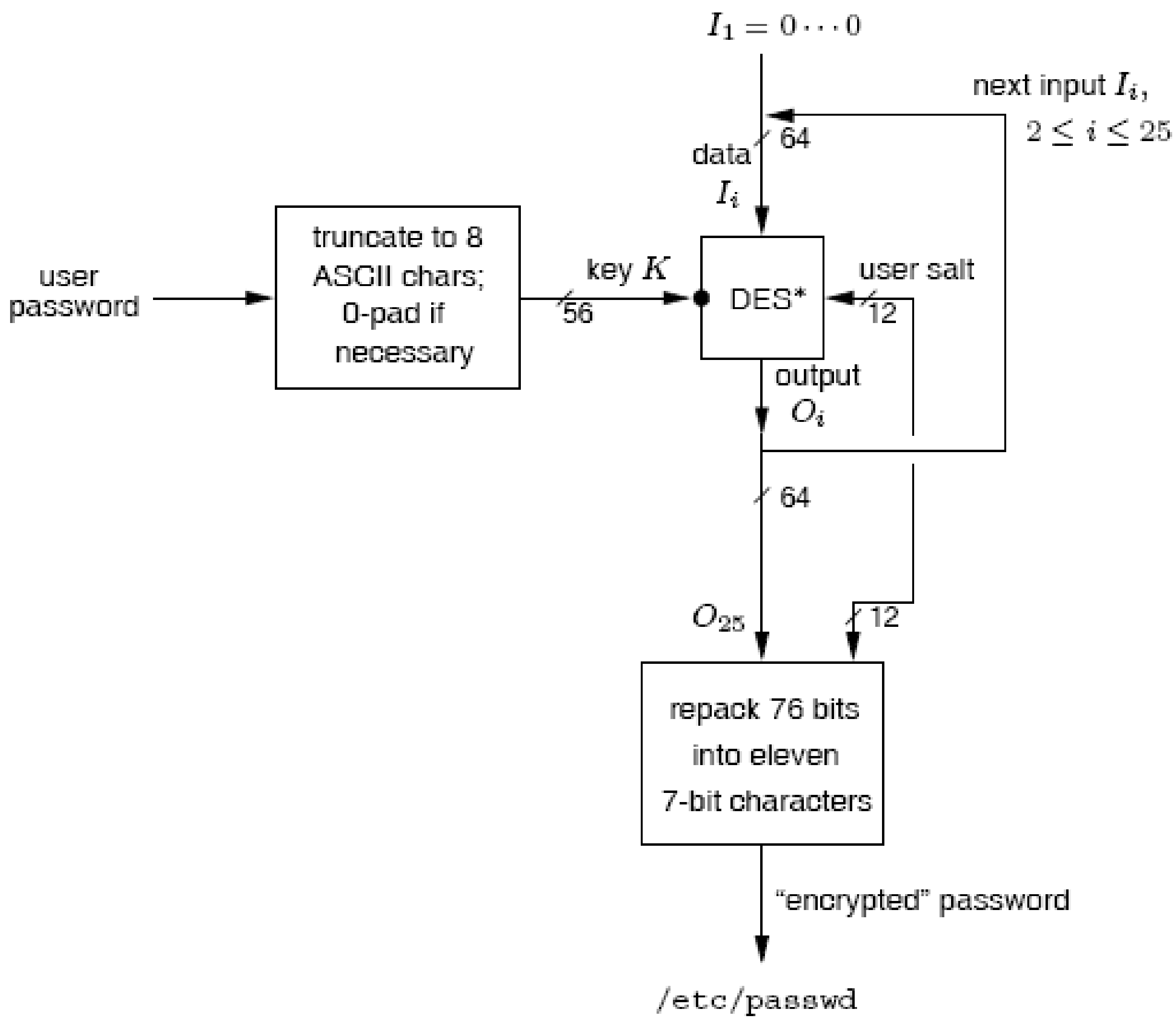
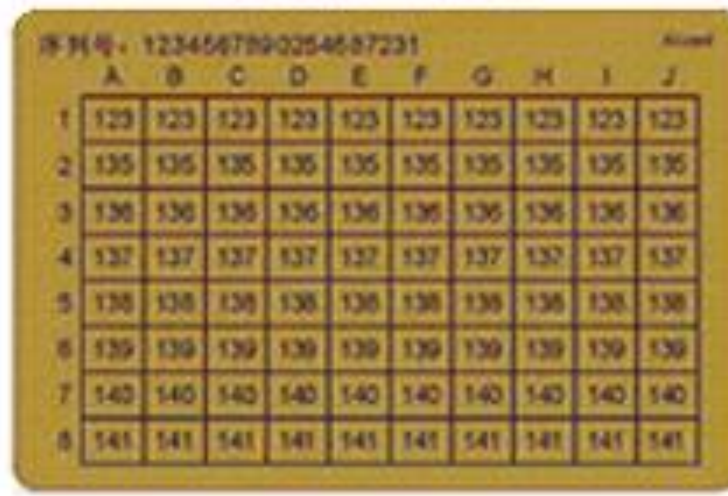


Figure 10.2: UNIX crypt *password mapping. DES* indicates DES with the expansion mapping E modified by a 12-bit salt.*

PINs

- PINs: Personal identification numbers
- PINs + “something possessed” (such as a plastic banking card with a magnetic stripe, or a chipcard)
- PINs are typically short (relative to fixed password schemes) and numeric, e.g., 4 to 8 digits.
- Restrict the number of incorrect entry of successive PINs

电子银行口令卡正面



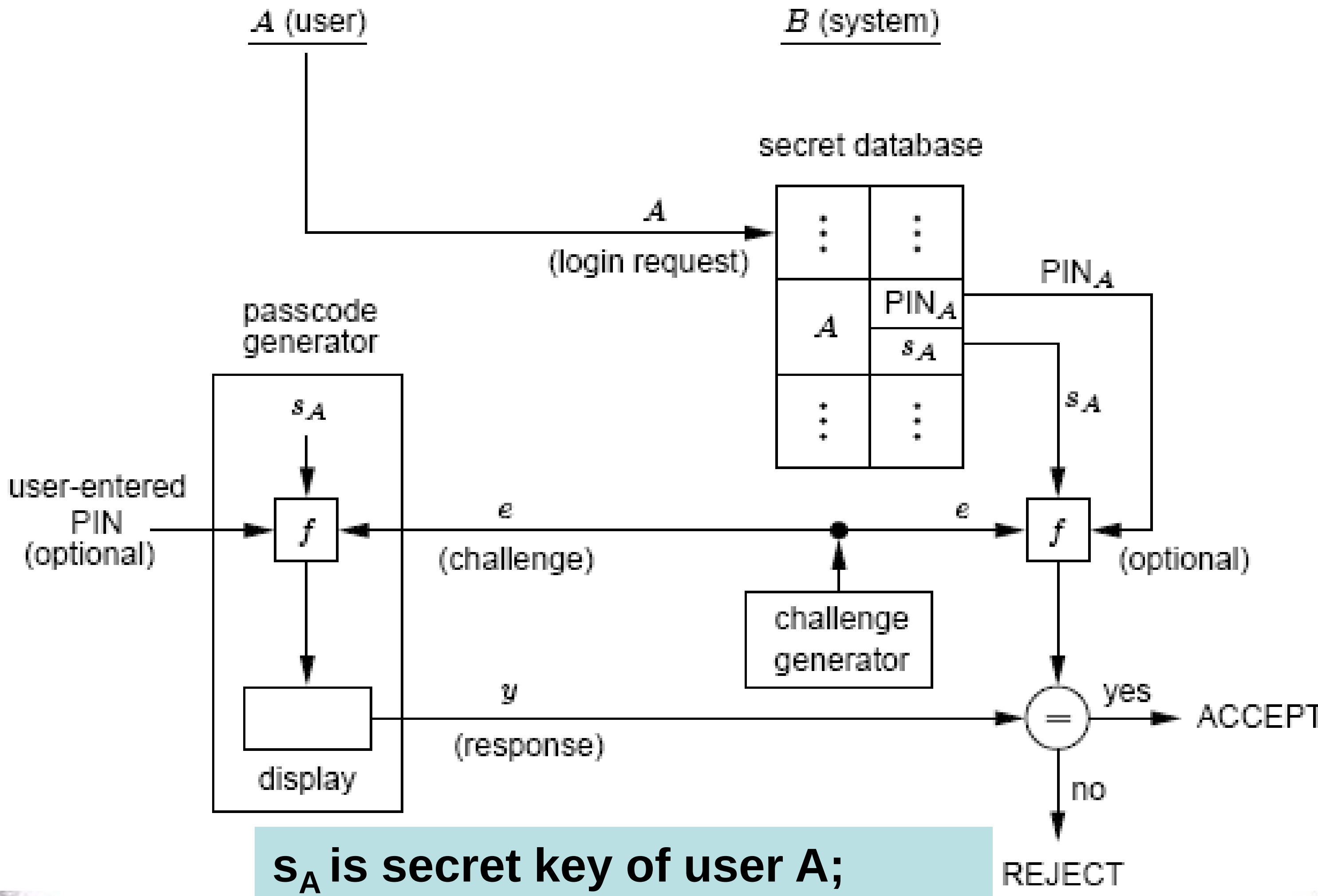
电子银行口令卡背面（覆膜刮开后的示意图）


U盾——网银安全的卫士
taoq.com
U盾


Microfotos


输入 PIN
登入"ICBC Token"
输入 PIN:
最短 PIN 长度 6 字节
最长 PIN 长度 8 字节
确定(O)
取消(C)
工商银行
COMMERCIAL BANK OF CHINA
定，交易成功

Case Study: Hand-held Passcode Generators



s_A is secret key of user A;

F is one-way function

One-time Passwords (Towards Strong Authentication)

- **each password is used only once.**
- **Prevent later attempt impersonation**
- **Variations include:**
  - **shared lists of one-time passwords.**
  - **sequentially updated one-time passwords.**
  - **one-time password sequences based on a one-way function.**
    - **w,H(w),H(H(w)), …**

Outline

- Objectives, Applications, Classification of Identification
- Passwords
- Challenge-response Identification
- Customized and Zero-knowledge Identification Protocols
- Attacks on Identification Protocols

- **Two roles:**
  - ***Prover/Claimant*** **is assured of the identity of a second party(*Verifier*)**
- **Challenge-response:**
  - **Challenge: *Prover/Claimant* ← *Verifier* *(not necessary)***
  - **Response: *Prover/Claimant* → *Verifier***
    - **demonstrating knowledge of a secret known to be associated with that claimant**
    - **without revealing the secret itself to the verifier during the protocol.**
    - **<u>Replay attack resistance</u>: using time-variant parameters**

Background on time-variant parameters

- <u>Time-variant parameters</u>
  - also called nonce, unique numbers, or non-repeating values
  - is a value used no more than once for the same purpose
  - typically serves to prevent (undetectable) replay.
- Three main classes of <u>time-variant parameters</u>
  - random numbers
  - sequence numbers
  - timestamps

Challenge-response by Symmetric-key Techniques

$$A \to B : E_K(t_A, B^*) \quad (1)$$

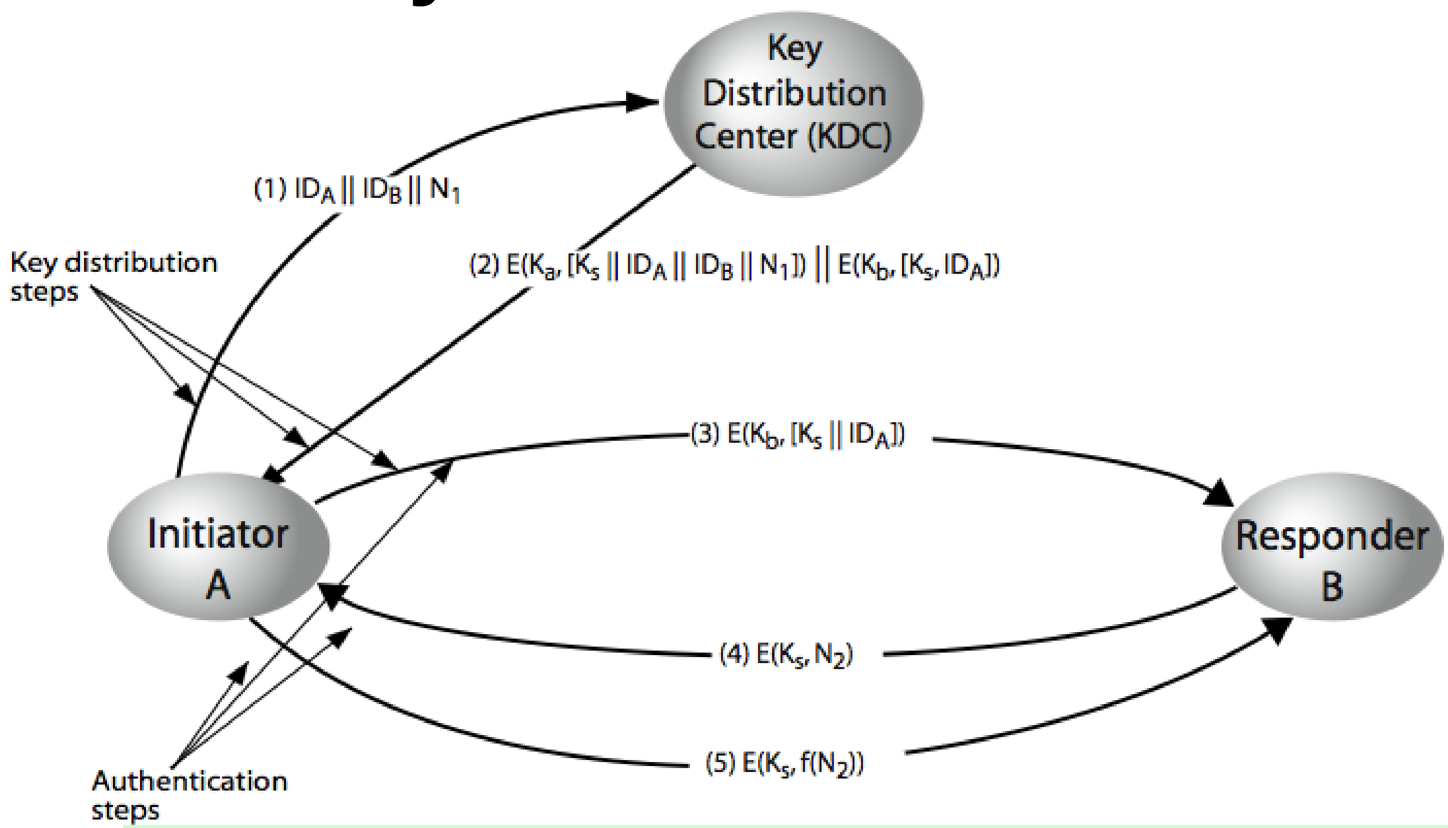
$$
\begin{aligned}
A \leftarrow B &: r_B && (1)\\
A \to B &: E_K(r_B, B^*) && (2)
\end{aligned}
$$

E_k is symmetric-key encryption;
t_A is timestamps;
r_A, r_B is random number;
B* is about identity

$$
\begin{aligned}
A \leftarrow B &: r_B && (1)\\
A \to B &: E_K(r_A, r_B, B^*) && (2)\\
A \leftarrow B &: E_K(r_B, r_A) && (3)
\end{aligned}
$$

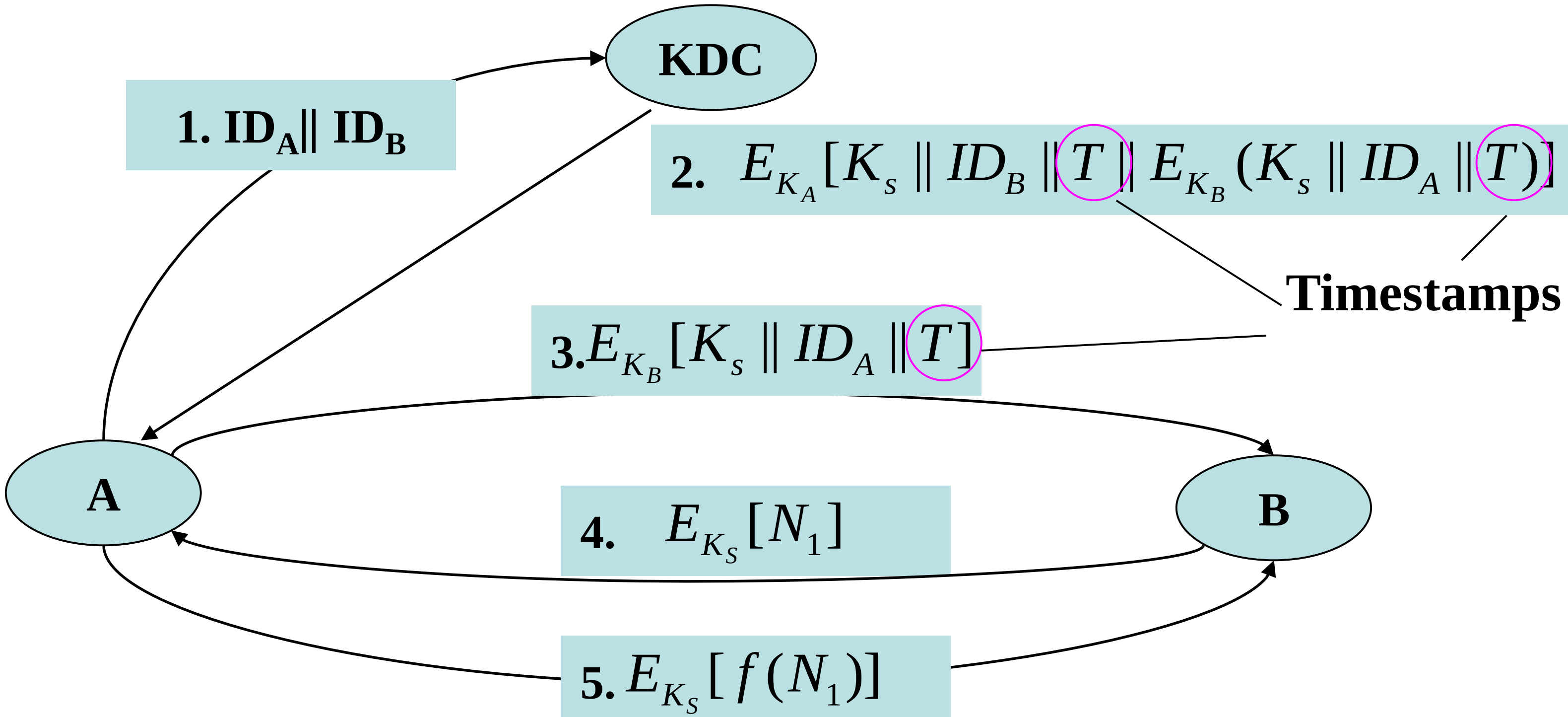
$$
\begin{aligned}
A \leftarrow B &: \quad r_B && (1)\\
A \to B &: \quad r_A,\ h_K(r_A, r_B, B) && (2)\\
A \leftarrow B &: \quad h_K(r_B, r_A, A) && (3)
\end{aligned}
$$

Case Study: Needham-Schroeder Shared-key Protocol



any party knowing an old session key Ks may both resend message (3) and compute a correct message (5) to impersonate A to B.

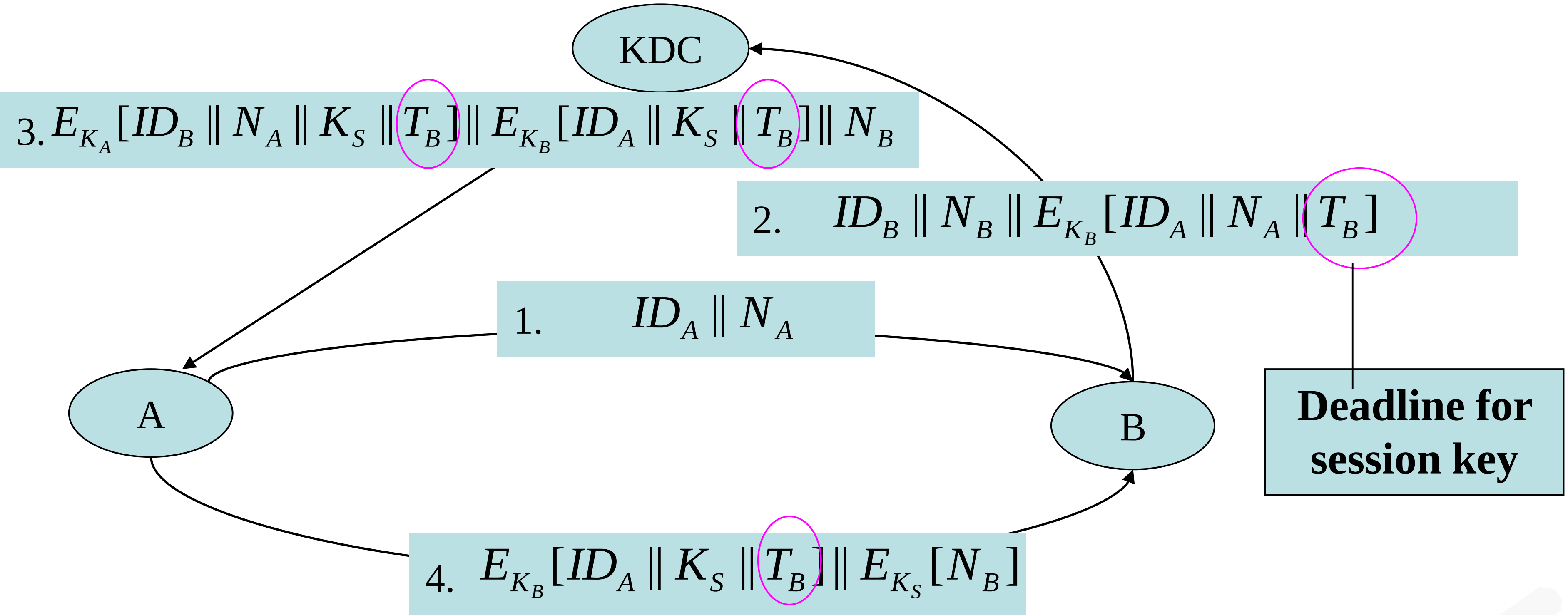
Needham-Schroeder Improvement (1)



Synchronization is necessary to clocks of each party

$|Clock - T| < \Delta t_1 + \Delta t_2$
Clock: For host;
Δt_1: estimated difference between hosts and KDC;
Δt_2: network delay;

Needham-Schroeder Improvement (2)



Needham-Schroeder Improvement (2)

In the valid lifetime of Ks, need not authentication of KDC

A → B: 1. $E_{K_B}[ID_A \| K_S \| T_B], N_A^{'}$

B → A: 2. $N_B^{'}, E_{K_S}[N_A^{'}]$

A → B: 3. $E_{K_S}[N_B^{'}]$

Challenge-response by Public-key Techniques

- A (claimant) demonstrating knowledge of its private key in one of two ways:
  - the claimant decrypts a challenge encrypted under its public key;
  - the claimant digitally signs a challenge.

Challenge-response Based on Public-key Decryption

- unilateral authentication:

$$
\begin{aligned}
A \leftarrow B: &\quad h(r), B, P_A(r, B) \qquad (1)\\
A \rightarrow B: &\quad r \qquad (2)
\end{aligned}
$$

- mutual authentication :

$$
\begin{aligned}
A \rightarrow B: &\quad P_B(r_1, A) \qquad (1)\\
A \leftarrow B: &\quad P_A(r_1, r_2) \qquad (2)\\
A \rightarrow B: &\quad r_2 \qquad (3)
\end{aligned}
$$

r, r_1, r_2 is random number;
P_A is encryption algorithm by using A's public key

Challenge-response Based on Digital Signatures

$$A \rightarrow B : cert_A, t_A, B, S_A(t_A, B) \quad (1)$$

$$A \leftarrow B : r_B \quad (1)$$
$$A \rightarrow B : cert_A, r_A, B, S_A(r_A, r_B, B) \quad (2)$$

$$A \leftarrow B : r_B \quad (1)$$
$$A \rightarrow B : cert_A, r_A, B, S_A(r_A, r_B, B) \quad (2)$$
$$A \leftarrow B : cert_B, A, S_B(r_B, r_A, A) \quad (3)$$

$cert_A$ is public-key certificate for A;
S_A denotes A's signature mechanism;
t_A is timestamps;r_A, r_B is random number;

Case Study: Needham-Schroeder public key protocol

- **Needham-Schroeder public key protocol**

1. $A \rightarrow B$: $E_B(N_A, A)$
2. $B \rightarrow A$: $E_A(N_A, N_B)$
3. $A \rightarrow B$: $E_B(N_B)$

improve

1. $A \rightarrow B$: $E_B(N_A, A)$
2. $B \rightarrow A$: $E_A(N_A, N_B, B)$
3. $A \rightarrow B$: $E_B(N_B)$

1\. $A \rightarrow C$: $E_C(N_A, A)$

1′. $C_A \rightarrow B$: $E_B(N_A, A)$

2′. $B \rightarrow C_A$: $E_A(N_A, N_B)$

2\. $C \rightarrow A$: $E_A(N_A, N_B)$

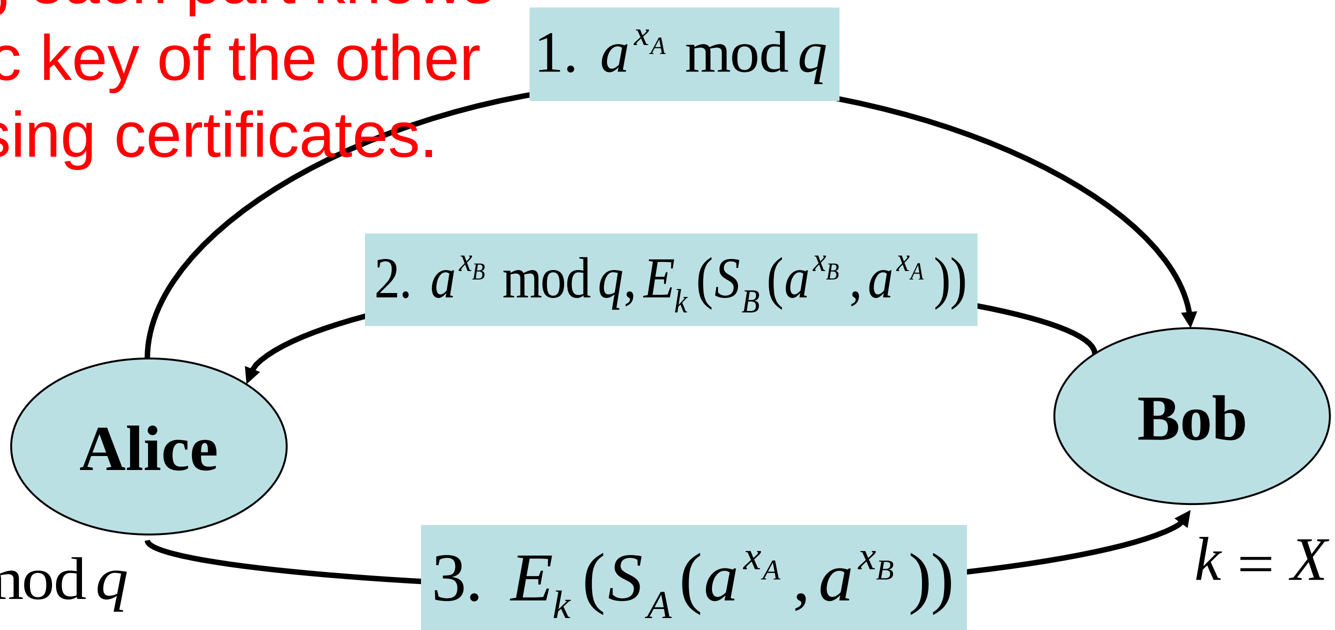
3\. $A \rightarrow C$: $E_C(N_B)$

3′. $C_A \rightarrow B$: $E_B(N_B)$

Interleaving attack:
$A \leftrightarrow C(A) \leftrightarrow B$

Review: Station to Station (STS)

Assuming each part knows real public key of the other part by using certificates.



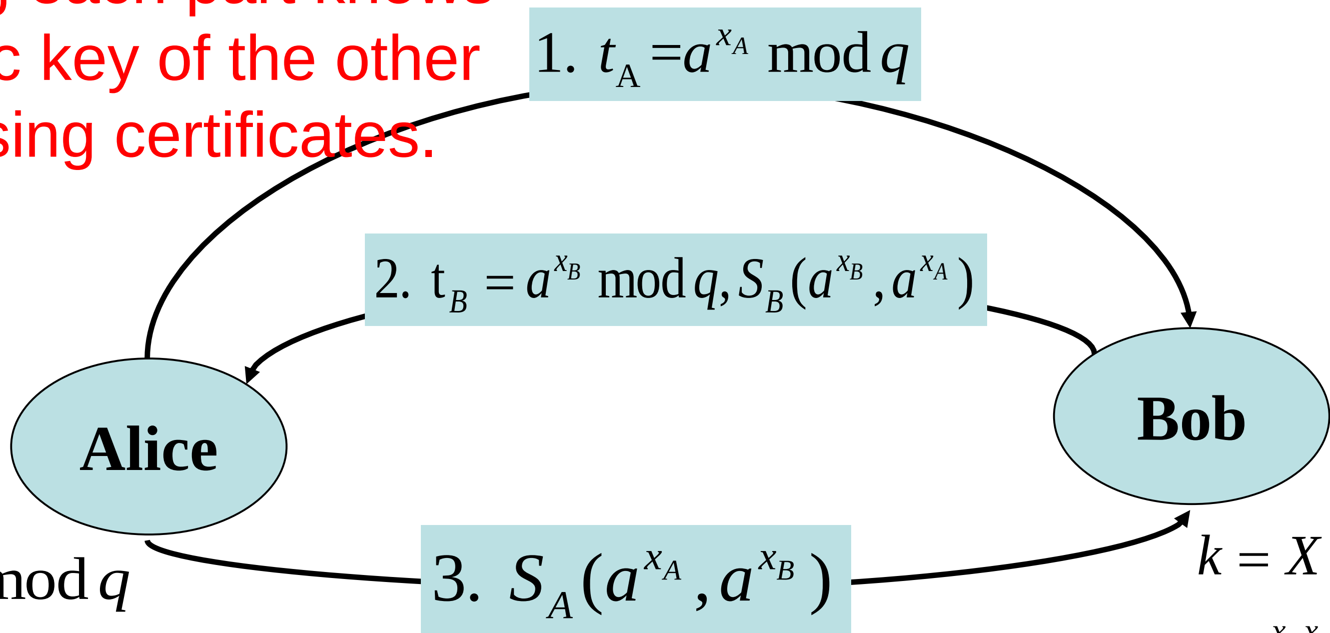
$$k = Y^{x_A} \bmod q = a^{x_A x_B} \bmod q$$

$$k = X^{x_B} \bmod q = a^{x_A x_B} \bmod q$$

one-time secret keys: x_A for Alice, x_B for Bob.
Private key: PR_A for Alice, PR_B for Bob
S_A denotes A’s signature mechanism;
$S_A(m)=E_{PR_A}(H(m))$

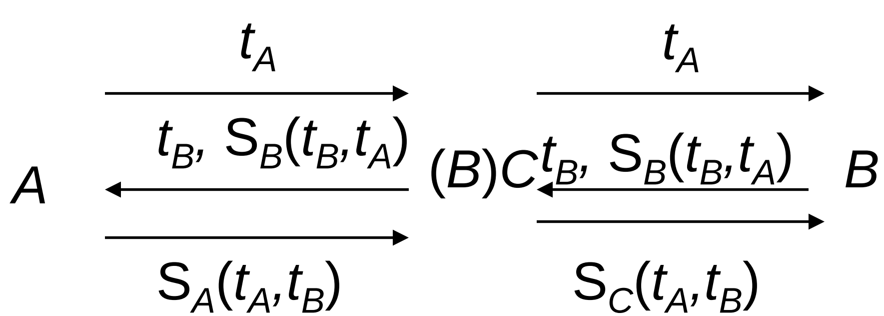
Case Study: STS without encryption

Assuming each part knows real public key of the other part by using certificates.



$$k = Y^{x_A} \bmod q$$
$$= a^{x_A x_B} \bmod q$$

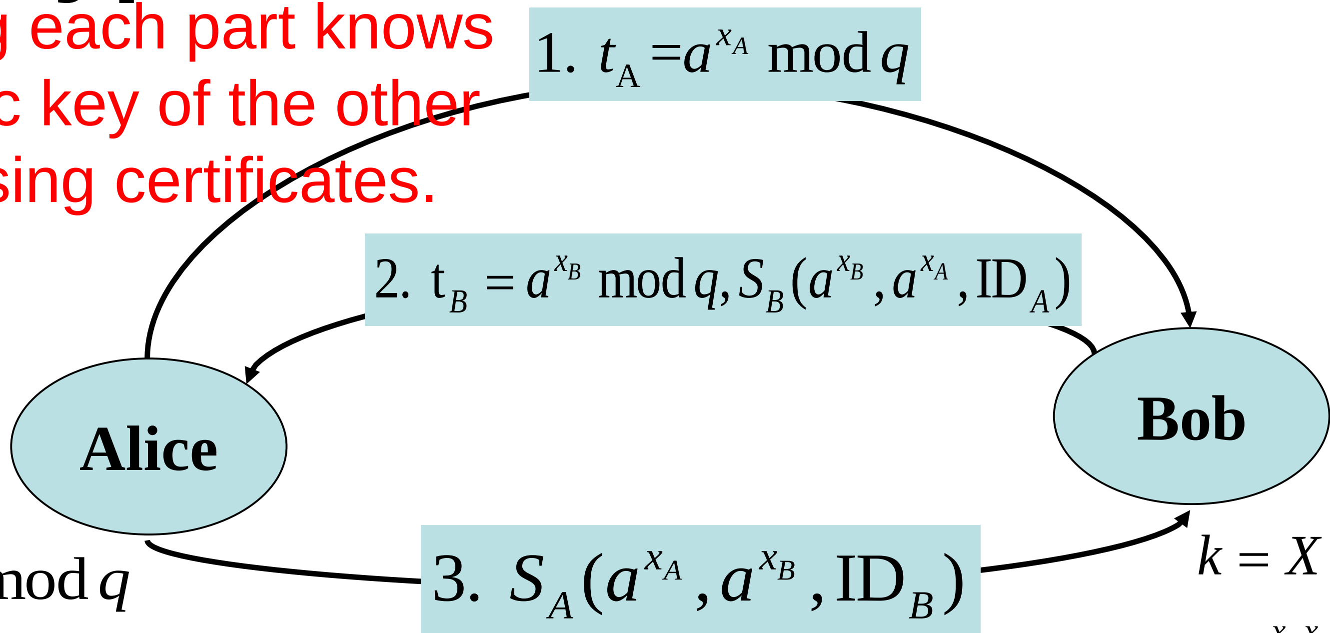
$$k = X^{x_B} \bmod q$$
$$= a^{x_A x_B} \bmod q$$



Interleaving attack

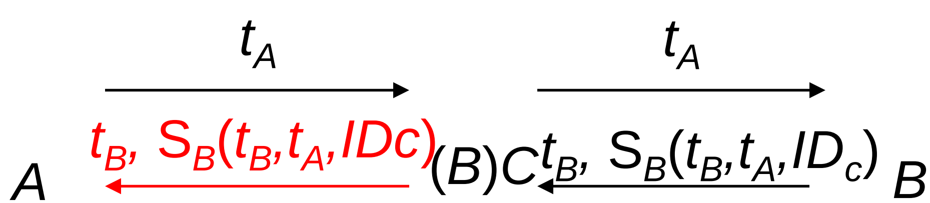
Impovement for STS without encryption

Assuming each part knows real public key of the other part by using certificates.



$$k = Y^{x_A} \bmod q = a^{x_A x_B} \bmod q$$

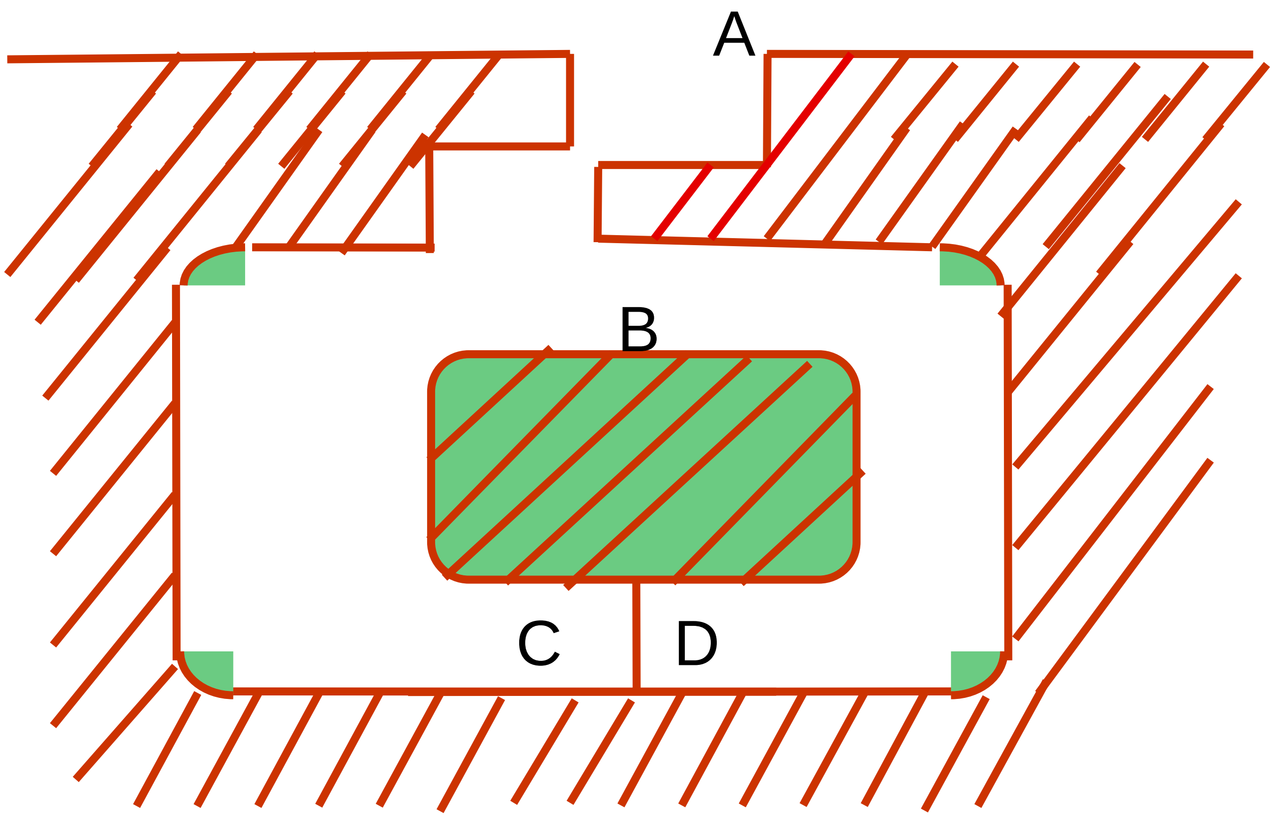
$$k = X^{x_B} \bmod q = a^{x_A x_B} \bmod q$$



Outline

- Objectives, Applications, Classification of Identification
- Passwords
- Challenge-response Identification
- Customized and Zero-knowledge Identification Protocols
- Attacks on Identification Protocols

zero-knowledge proof of Alibaba



Zero-knowledge Identification Protocols

- Use asymmetric techniques but do not rely on digital signatures or public-key encryption
- Use random number as challenge but do not use block ciphers, sequence numbers, and timestamps.
- General structure of zero-knowledge protocols
  - $A \rightarrow B$: (public) witness computed from a random element
  - $A \leftarrow B$: challenge by selecting one question.
  - $A \rightarrow B$: response by answering question (and further B judges by checking for its correctness).

Fiat-Shamir identification protocol (basic version)

1. One-time setup.

(a) A trusted center T selects and publishes an RSA-like modulus $n = p \times q$ but keeps primes p and q secret.

(b) Each claimant A selects a secret s co-prime to n, $1 \leq s \leq n - 1$, computes $v = s^2 \bmod n$, and registers (v, n) with T as its public key.

2. **Protocol messages.** Each of t rounds has three messages with form as follows.

- $A \rightarrow B$： $x = r^2 \bmod n$
- $A \leftarrow B$： $e \in \{0,1\}$
- $A \rightarrow B$： $y = r \times s^e \bmod n$

3. Protocol actions. The following steps are iterated t times (sequentially and independently). B accepts the proof if all t rounds succeed.

(a) A chooses a random number r, $1 \le r \le n - 1$, and sends (the *witness*) $x = r^2 \bmod n$ to B.

(b) B randomly selects a (*challenge*) bit $e = 0$ or $e = 1$, and sends e to A.

(c) A computes and sends to B (the *response*) y

- $y = r$ (if $e = 0$)
- $y = r \times s \bmod n$ (if $e = 1$).

(d) B judges.

- rejects the proof if $y = 0$, (Note checking for $y = 0$ precludes the case $r = 0$)
- and otherwise accepts upon verifying $y^2 \equiv x \cdot v^e \pmod n$. (Depending on e, $y^2 = x$ or $y^2 = xv \bmod n$, since $v = s^2 \bmod n$.

Feige-Fiat-Shamir Identification Protocol

1. One-time setup.

(a) Trusted center T selects and publishes modulus $n = p \times q$ but keeps primes p and q secret.

(b) The claimant A selects k random secret integers $s_1, s_2, \ldots, s_k$ (s.t. $1 \le s_i \le n - 1$ and s_i co-prime to n)

(c) A selects k random bits $b_1, \ldots, b_k$ and computes $v_i=(-1)^{b_i} \times (s_i^2)^{-1} \bmod n$, and registers $(v_1, v_2, \ldots v_k; n)$ with T as its public key.

2. Protocol messages. Each of t rounds has three messages with form as follows.

$$
\begin{array}{llr}
A \rightarrow B: & x \ (= \pm r^2 \bmod n) & (1) \\
A \leftarrow B: & (e_1, \ldots, e_k), \ e_i \in \{0, 1\} & (2) \\
A \rightarrow B: & y \ (= r \cdot \prod_{e_j=1} s_j \bmod n) & (3)
\end{array}
$$

3. Protocol actions. **The following steps are iterated t times (sequentially and independently). B accepts the proof if all t rounds succeed.**

(a) A chooses a random integer r, $1 \le r \le n-1$, and a random bit b; computes $x = (-1)^b \cdot r^2 \bmod n$; and sends x (the *witness*) to B.

(b) B sends to A (the *challenge*,) a random k-bit vector $(e_1, \ldots, e_k)$.

(c) A computes and sends to B (the *response*): $y = r \cdot \prod_{j=1}^{k} s_j^{e_j} \bmod n$ (the product of r and those s_j specified by the challenge).

(d) B computes $z = y^2 \cdot \prod_{j=1}^{k} v_j^{e_j} \bmod n$, and verifies that $z = \pm x$ and $z \neq 0$. (The latter precludes an adversary succeeding by choosing $r = 0$.)

Converting Identification to Digital Signature Scheme

- replace the random challenge e of the verifier by the one-way hash $e = h(x||m)$
  - x: witness;
  - m: message to be signed
  - h: hash function
- the bitsize of the challenge e must typically be increased to preclude off-line attacks on the hash function.

Outline

- Objectives, Applications, Classification of Identification
- Passwords (Weak Authentication)
- Challenge-response Identification (Strong Authentication)
- Customized and Zero-knowledge Identification Protocols
- Attacks on Identification Protocols

Attacks

- **Impersonation:**
  - a deception whereby one entity purports(声称) to be another.
- **Replay attack**
  - an impersonation or other deception involving use of information from a single previous protocol execution,
- **Interleaving attack**
  - an impersonation or other deception involving selective combination of information from one or more previous or simultaneously ongoing protocol executions (*parallel sessions*), including possible origination of one or more protocol executions by an adversary itself.

- **reflection attack**
  - an interleaving attack involving sending information from an ongoing protocol execution back to the originator of such information.
- **forced delay**
- **chosen-text attack**
  - an attack on a challenge-response protocol wherein an adversary strategically chooses challenges in an attempt to extract information about the claimant's long-term key.

Case Study: problem 14.2 in textbook *P471*

- the protocol is as follows. Each node N of the network has been assigned a unique secret key Kn. This key is used to secure communication between the node and a trusted server. That is, all the keys are stored also on the server. User A, wishing to send a secret message M to user B, initiates the following protocol:
  - A generates a random number R and sends to the server his name A, destination B, and E(Ka, R).
  - Server responds by sending to E(Kb, R) to A.
  - A sends E(R, M) together with E(Kb, R) to B.
  - B knows Kb, thus decrypts E(Kb, R) to get R and will subsequently use R to decrypt E(R, M) to get M.
- Analysis: The protocol isn't secure because the server doesn't authenticate users who send him a request. Apparently designers of the protocol have believed that sending E(Kx, R) implicitly authenticates user X as the sender, as only X (and the server) knows Kx But you know that E(Kx, R) can be intercepted and later replayed.
- Most likely An attacker works as follows. After intercepting E(Ka, R) and E(R, M) (see steps 1 and 3 of the protocol), the man, let's denote him as Z, will continue by pretending to be A and ...

Replay attack

Case Study: problem 9.15 in textbook *P311*

The following protocol for communication between two parties A and B, where user A wishing to send message M to user B: (messages exchanged are in the format (sender's name, text, receiver's name).

1. A sends B the block: (A, E(PU_b, M), B).
2. B acknowledges receipt by sending to A the block: (B, E(PU_a, M), A).

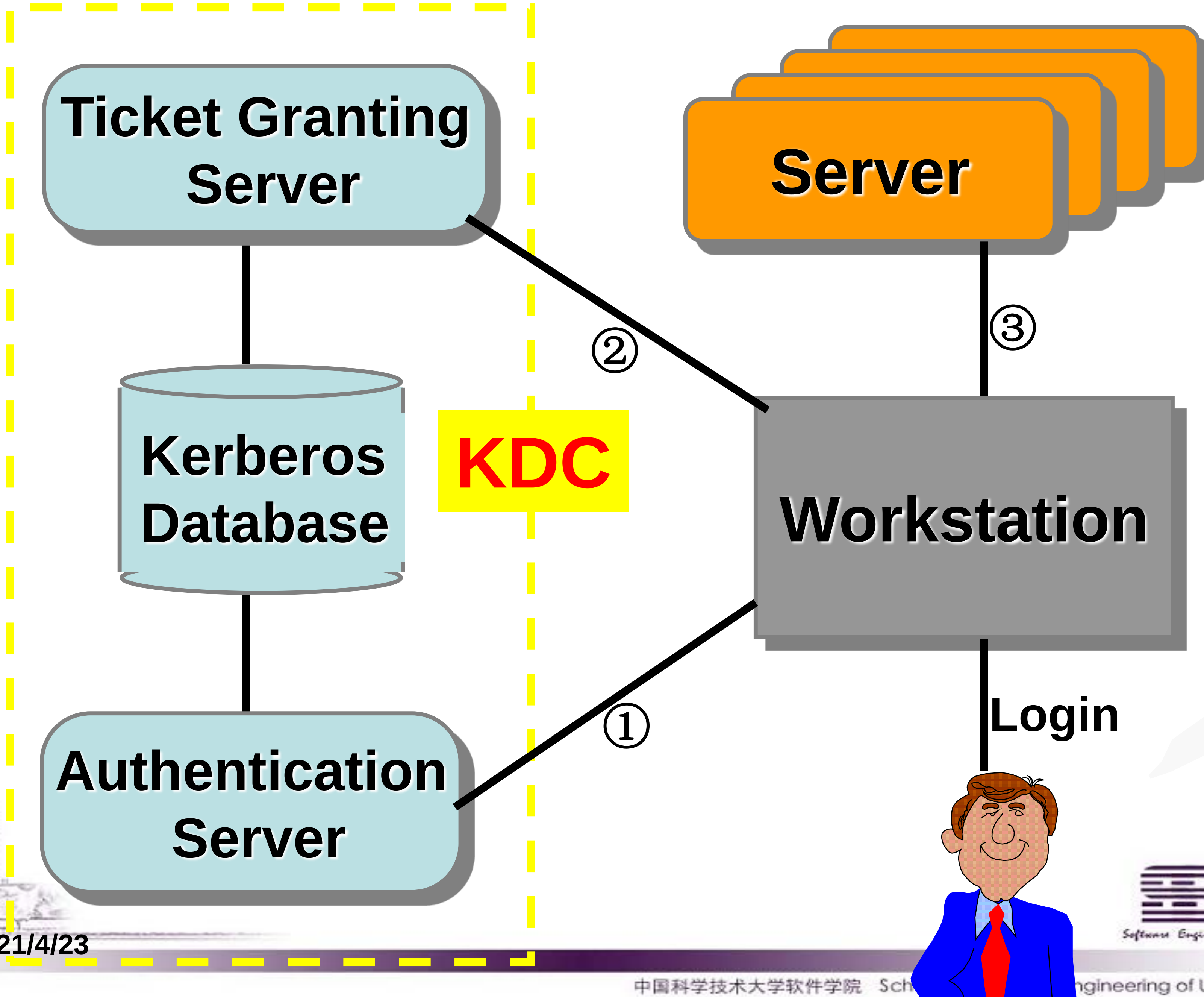
Data Confidentiality？

Replay attack

improve

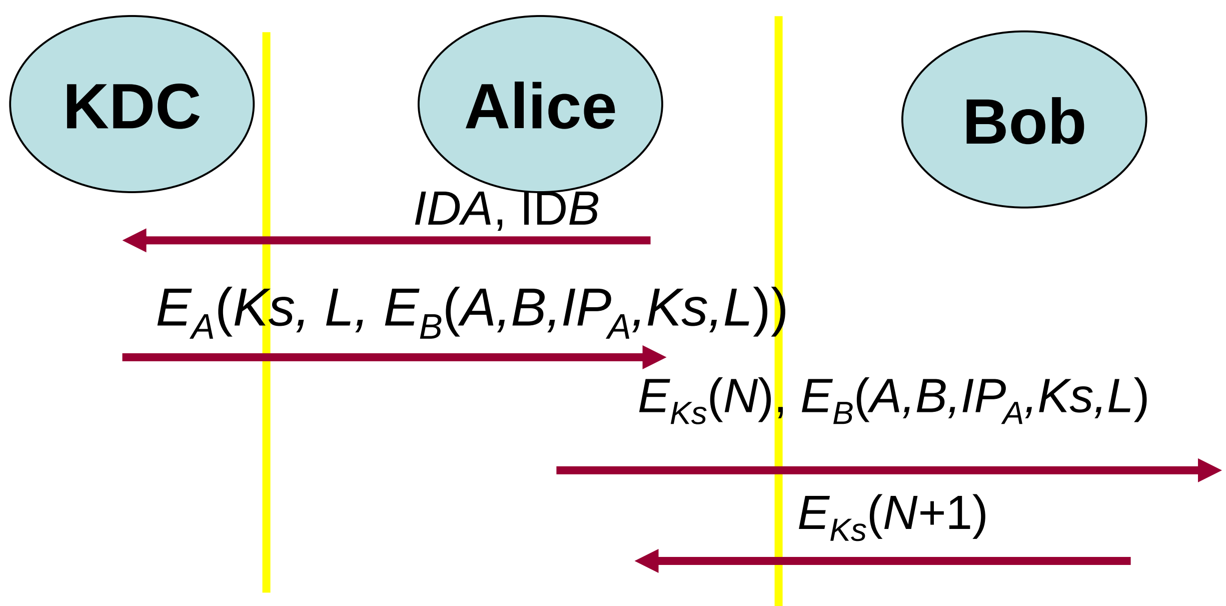
1. A sends B the following block: (A, E(PU_b, [M, A]), B).
2. B acknowledges receipt by sending to A the following block: (B, E(PU_a, [M, B]), A).

Case Study: Kerberos key distribution service



Basic Principle of Kerberos

- **Main target： entity authentication**
- **Additional results: shared secret key distribution**



- $E_A(Ks, L, E_B(A,B,IP_A,Ks,L))$
  - L: lifetime of session key K

Encryption by master key between KDC and B

session key

Ticket:
login name
TGS name
net address
TGS session key

Encryption by master key between KDC and A

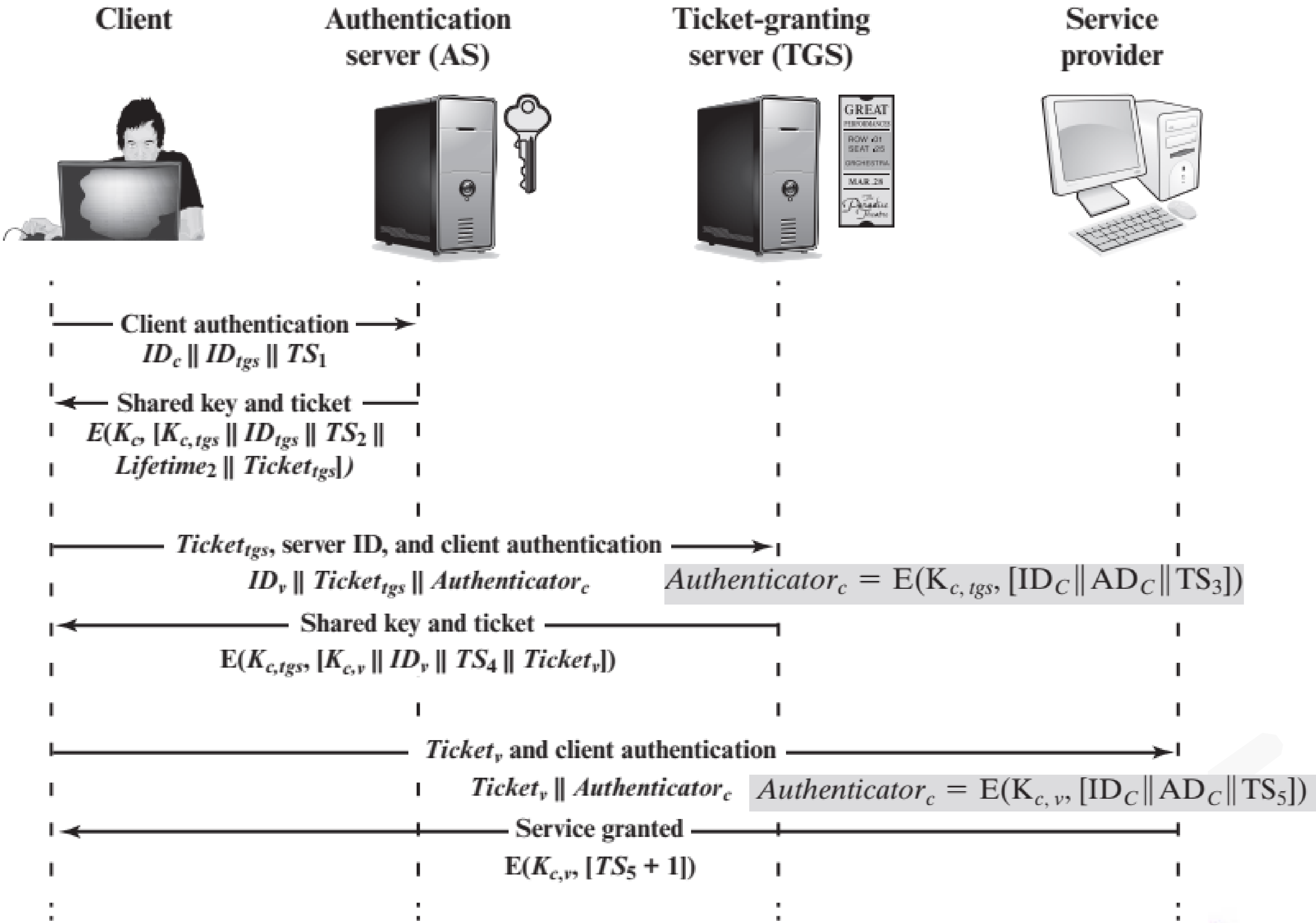
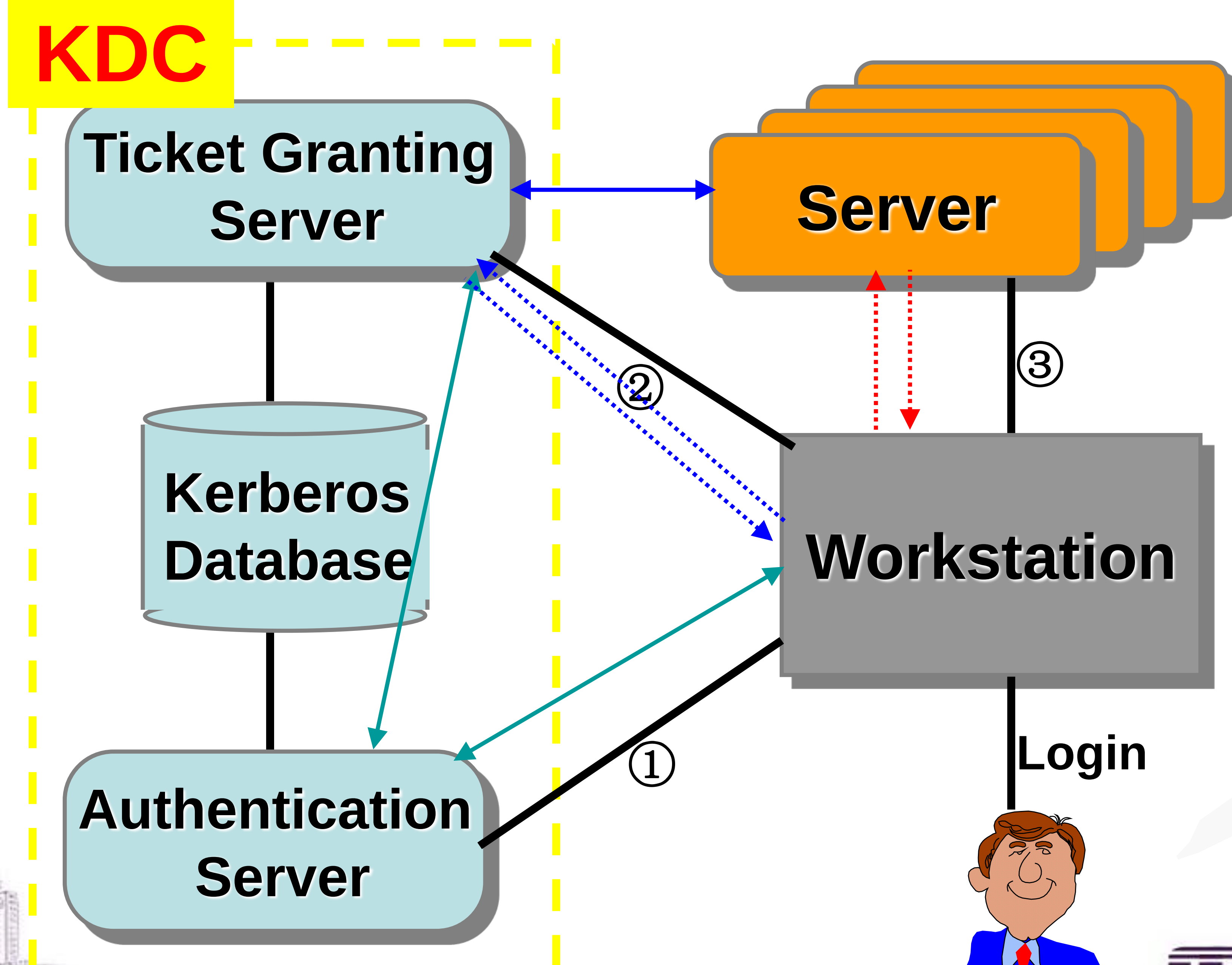


Figure 15.3 Kerberos Exchanges

$Ticket_{tgs} = E(K_{tgs}, [K_{c,tgs} \parallel ID_C \parallel AD_C \parallel ID_{tgs} \parallel TS_2 \parallel Lifetime_2])$

$Ticket_v = E(K_v, [K_{c,v} \parallel ID_C \parallel AD_C \parallel ID_v \parallel TS_4 \parallel Lifetime_4])$



Kerberos **key distribution service**

Attacks vs. Countermeasures

Type of attack	Principles to avoid attack
replay	use of challenge-response techniques; use of nonces; embed target identity in response
interleaving	linking together all messages from a protocol run (e.g., using chained nonces)
reflection	embed identifier of target party in challenge responses; construct protocols with each message of different form (avoid message symmetries); use of uni-directional keys
chosen-text	use of zero-knowledge techniques; embed in each challenge response a self-chosen random number (*confounder*)
forced delay	combined use of random numbers with short response time-outs; timestamps plus appropriate additional techniques

Table 10.3: *Identification protocol attacks and counter-measures.*

Review Questions

- **15.2 List three general approaches to dealing with replay attacks.**
- **15.4 What problem was Kerberos designed to address?**

- **List at least three general means of authenticating a user's identity.**
- **It is more secure using salted hash value stored in the server. Why?**

Problems

- **14.1**
- **14.2**
- **15.10 In Kerberos, when Bob receives a Ticket from Alice, how does he know it is not genuine?**
- **15.11 In Kerberos, how does Bob know that the received token is not corresponding to Alice's?**
- **15.12 In Kerberos, how does Alice know that a reply to an earlier message is from Bob?**

Thanks!