Robust Remote Authentication Schemes with Smart Cards

Chun-I Fan, Yung-Cheng Chan, Zhi-Kai Zhang
Department of Computer Science and Engineering
National Sun Yat-sen University, Taiwan


The final publication is available at http://www.sciencedirect.com
Outlines

- Introduction
- Basic Ideas
- The Proposed Scheme
- Discussions
- Implementation
- Three-factor Authentication
A password table:

<table>
<thead>
<tr>
<th>User ID</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>Irene</td>
<td>“HoneyPot”</td>
</tr>
<tr>
<td>Cody</td>
<td>“Rabbit”</td>
</tr>
<tr>
<td>Anita</td>
<td>“Happy1763”</td>
</tr>
</tbody>
</table>

- It cannot be revealed. (Possible solutions: Hashing or Encryption)
- It is inefficient to maintain the password table.

Do not keep the password table
• Long or meaningless passwords are difficult to be memorized by users.

“57t#h!fd0j487w-thkwhwi+odj&a90^854-hqig8je*wgnjcas345”

Allow the users to choose their passwords
The limited computation capabilities of the smart cards should be considered.

- Low cost: XOR, hashing, addition, and multiplication
- Heavy cost: Multi-exponentiation and inverse computation.

Time-consuming operations are not suitable for the smart cards.
The replay attack:
Step 1: Intercept the login messages
Step 2: Re-send these messages
Step 3: Impersonate the legitimate users

Withstand the replay attack
Introduction (5/8)

Timestamp-based schemes

• The problems of clock synchronization and delay-time limitation exist.

• Only suitable for stable LANs

No clock synchronization and delay-time limitation
The offline dictionary attack: (with the card)

Step 1: Intercept the login messages.
(Step 2: Get the information in the smart card.)
Step 3: Guess the user’s password.
Step 4: Examine if the guessed password is correct.

\[ F(..., \text{password}, ...) = G(...) \]

Step 5: If it is false, go to Step 3.

Resist the offline dictionary attack
Server authentication:
  • An illegal system may cheat the users.
  • It may obtain secret information from the users.

*Mutual authentication is required*
Introduction (8/8)

- Lost the smart card
  - Solution: Re-register for a new card without changing the old password.
- Lost both the password and the card
  - Solution: Choose a new password and re-register for a new card.

The lost cards must be revoked.
Basic Ideas (1/5)

Splitting the password table:

• The password record of each user is encrypted and stored in her/his smart card.
• The user sends the encrypted password to the system in the login phase.
• The system decrypts it to check the password.
• No password table is maintained by the system.
Two-factor authentication:

$U_i$: the secrecy kept by user $i$
$C_i$: the secrecy stored in the card of user $i$
$S$: the secrecy kept by the system

$U_i \cap C_i = \emptyset$
$C_i \subset S$
$U_i \not\subset S$

$C_i \Rightarrow$ login failure
$U_i \Rightarrow$ login failure
$U_i \cup C_i \Rightarrow$ login success
Basic Ideas (3/5)

Secure channels:
(1) Protecting the login messages
(2) Public-key cryptosystems
(3) User efficiency
Basic Ideas (4/5)

Lost card revocation:
• To distinguish valid cards from all invalid ones
• The registration table: \((ID_i, Valid\_CID_i)\)
  – Each card identifier is quite short.
  – Do not require keeping these records secret.
  – Cost efficient
Basic Ideas (5/5)

Nonce-based protocols
• Nonce = Use only once.

A

Key: $K$

B

$E_K(r)$

$r$
The Proposed Scheme (1/3)

The registration protocol:

**User** $i$

$ID_i$: the identity of user $i$

$PW_i$: the password of user $i$

Keep {the smart card, $PW_i$} (in a secure manner)

**The System**

$H(\cdot)$: a public one-way hashing

$p, q$: two distinct large primes

$n$: the product of $p$ and $q$

$E_s(\cdot)$: the encryption with key $s$

$CID_i$: the card identity of user $i$

$\wp_i$: a randomly-chosen string

$b_i = E_s(H(PW_i) || H(ID_i) || CID_i || \wp_i)$

The smart card $\supset \{CID_i, ID_i, b_i, n\}$

An identification protocol

{${ID_i, H(PW_i)}$} (in a secure manner)
The Proposed Scheme (2/3)

The registration table:

<table>
<thead>
<tr>
<th>User Identity</th>
<th>Valid Card Identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>$ID_1$</td>
<td>$Valid_CID_1$</td>
</tr>
<tr>
<td>$ID_2$</td>
<td>$Valid_CID_2$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>$ID_i$</td>
<td>$Valid_CID_i$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

It is not necessary to keep the table secret but the integrity must be guaranteed.
The Proposed Scheme (1/3)

The login protocol:

User $i$

Insert the card and input $PW_i$

$L_1 = \{ID_i, (b_i||H(ID_i)||u)^2 \text{ mod } n\}$

($u$: a randomly-chosen string)

The System

$L_1$\[\rightarrow\]

Decrypt:
1. $(b_i||H(ID_i)||u)^2 \text{ mod } n \rightarrow (b_i||H(ID_i)||u)$
2. $b_i \rightarrow (H(PW_i)||H(ID_i)||CID_i||v_i)$

Check if:
1. $H(ID_i)$ is in the result of decrypting $b_i$
2. $ID_i$ exists in the registration table
3. $CID_i$ is equal to $Valid\_CID_i$

Compute:
1. a randomly-chosen string $r$
2. $\alpha = r \oplus u$ and $\beta = H(r||u)$

$L_2 = \{\alpha, \beta\}$\[\leftarrow\]

$L_3 = H(H(PW_i)||r)$

$L_3$\[\rightarrow\]

Check if $L_3$ is equal to $H(H(PW_i)||r)$
Low computation for the smart card:

- 4 hashing computations
- 1 random-number generation
- 1 modular squaring computation
- 1 XOR operation
- Some string concatenations
No password table:

- Split the password table into entries and securely store each entry in the smart card. \( b_i = E_s(H(PW_i)\|H(ID_i)\|CID_i\|v_i) \)

- The registration table
  - It does not need to be kept secret but the integrity must be guaranteed.
  - Much smaller than the password table
  - Easily maintained
Discussions (3/13)

Passwords are chosen by the users:
• Not decided by the system
• Each user can choose any one of her/his favorite strings.
• It can help the users with memorizing their own passwords.
Discussions (4/13)

Not requiring clock synchronization and delay-time limitation:

– Nonces instead of timestamps
– Eliminate the problem of clock synchronization.
– It is not necessary to consider the limitation of delay time.
Discussions (5/13)

Two-factor security:

- $L_1 = \{ID_i, (b_i||H(ID_i)||u)^2 \mod n\}$ cannot be computed without the card.
- $L_3 = H(H(PW_i)||r)$ cannot be computed without the hashed value of the password.
Withstanding the replay attack:

If attackers replay \( L_1 = \{ID_i, (b_i||H(ID_i)||u)^2 \text{ mod } n\} \)

- \((b_i||H(ID_i)||u)^2 \text{ mod } n \rightarrow u\) (It is intractable.)
- \(r \oplus u \rightarrow r\) (It is impossible.)
- Without \(r\); \(L_3 = H(H(PW_i)||r)\) cannot be constructed even if the attackers have \(H(PW_i)\).
Discussions (7/13)

Server authentication:

• The attackers do not have \((p, q)\).
• \((b_i \| H(ID_i) \| u)^2 \mod n \rightarrow u\) (It is intractable.)
• Without \(u\), \(L_2 = \{r \oplus u, H(r \| u)\}\) cannot be constructed correctly.
Discussions (8/13)

Resisting the offline dictionary attack without the smart card:

• The attackers can intercept \( \{L_1, L_2, L_3\} \).
• \((b_i|H(ID_i)||u)^2 \mod n \rightarrow u\) (It is intractable.)
• \(r \oplus u \rightarrow r\) (It is impossible.)
• Without \(r\), the attackers cannot guess the password such that \(L_3 = H(H(PW_i)||r)\).
Resisting the offline dictionary attack WITH the smart card:

• It can resist the offline dictionary attack without the smart card.

• $b_i = E_s(H(PW_i)\|H(ID_i)\|CID_i\|v_i)$ cannot be decrypted without $s$. 

Discussions (9/13)
Discussions (10/13)

Revoking the lost cards without changing the users' identities:

- \( CID_i \) embedded in the new card of user \( i \) and \( Valid\_CID_i \) stored in the system will be increased by 1.
- The system will check if a received \( CID_i \) equals to \( Valid\_CID_i \) in the registration table.
### Discussions (11/13)

<table>
<thead>
<tr>
<th></th>
<th>Ours</th>
<th>[8]</th>
<th>[28]</th>
<th>[15]</th>
<th>[26]</th>
<th>[13]</th>
<th>[7]</th>
<th>[3]</th>
<th>[6]</th>
<th>[23]</th>
<th>[24]</th>
<th>[29]</th>
<th>[25]</th>
<th>[27]</th>
<th>[1]</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>C2</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>C3</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>C4</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>C5</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>C6</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>C7</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>C8</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>C9</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

- C1: User efficiency (Low computation for the smart card)
- C2: No password table
- C3: Passwords chosen by the users themselves
- C4: Not requiring clock synchronization and delay-time limitation
- C5: Withstanding the replay attack
- C6: Server authentication
- C7: Withstanding the offline dictionary attack without the card
- C8: Withstanding the offline dictionary attack with the card
- C9: Revoking the lost cards without changing users’ identities
Discussions (12/13)

<table>
<thead>
<tr>
<th>Ours</th>
<th>[8]</th>
<th>[28]</th>
<th>[15]</th>
<th>[26]</th>
<th>[13]</th>
<th>[7]</th>
<th>[3]</th>
<th>[6]</th>
<th>[23]</th>
<th>[24]</th>
<th>[29]</th>
<th>[25]</th>
<th>[27]</th>
<th>[1]</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>&lt;&lt;T</td>
<td>&lt;&lt;T</td>
<td>T</td>
<td>&lt;&lt;T</td>
<td>&lt;&lt;T</td>
<td>&lt;&lt;T</td>
<td>&lt;&lt;T</td>
<td>&lt;&lt;T</td>
<td>3T</td>
<td>&lt;&lt;T</td>
<td>3T</td>
<td>2T</td>
<td>3T</td>
<td>&lt;&lt;T</td>
</tr>
<tr>
<td>P2</td>
<td>T</td>
<td>&lt;&lt;T</td>
<td>&lt;&lt;T</td>
<td>&lt;&lt;T</td>
<td>&lt;&lt;T</td>
<td>&lt;&lt;T</td>
<td>&lt;&lt;T</td>
<td>&lt;&lt;T</td>
<td>3T</td>
<td>&lt;&lt;T</td>
<td>3T</td>
<td>2T</td>
<td>2T</td>
<td>&lt;&lt;T</td>
</tr>
<tr>
<td>P3</td>
<td>1.85</td>
<td>1.85</td>
<td>&lt;</td>
<td>[n]</td>
<td>&lt;</td>
<td>[n]</td>
<td>&lt;</td>
<td>[n]</td>
<td>&lt;</td>
<td>[n]</td>
<td>&lt;</td>
<td>[n]</td>
<td>&lt;</td>
<td>[n]</td>
</tr>
</tbody>
</table>

P1: The computation cost of the smart card
P2: The computation cost of the system
P3: The communication cost
$T$ = the computation cost of a modular exponentiation computation in $\mathbb{Z}_n^*$
$|n| = 1024$ bits
Discussions (13/13)

Summary:
1. It can resist the offline dictionary attack without or with the smart card.
2. Lost cards can be revoked in the scheme.
Implementation (1/9)

[Image of a software interface]

- 執行狀態: 身分驗證
  Step 1. 請按 [檢驗] 以檢查智慧卡及選票中心的正確性

- 輸出結果:
  選票中心 IP 為: 140.117.176.118
  請記得將智慧卡放入讀卡機!

- 入視窗:

- 候選人名單:
  - 請選擇候選人號碼 -
  - 檢驗
  - 結束
Implementation (2/9)
Implementation (4/9)
Implementation (5/9)
Implementation (6/9)

[Image of a dialog box with options for execution status, output result, input window, and candidate list]

[Text content: Step 2, please input password, and press [Enter] to complete identity verification.]

[Text content: Identity card and election center verification is correct, please input password.]

[Input window]

[Text content: Please select candidate number -]
Implementation (7/9)
Implementation (8/9)
Implementation (9/9)