A Privacy-Restoring Mechanism for Offline RFID Systems

Gildas Avoine   Iwen Coisel   Tania Martin

Université catholique de Louvain
Belgium

April 16, 2012
Authentication protocol that restores privacy in case of compromised readers in offline RFID systems
### Offline RFID Systems

**Online system**
- Fixed readers
- Always connected to BE
- Readers do not store data to authenticate tags

**Offline system**
- Handheld readers
- Operate without BE
- Readers **must store** all data to authenticate tags
  - i.e. *all tags’ secrets*
Tag corruption

- A steals secrets of the corrupted tag

vs.

Compromised reader in offline RFID systems

- A steals all tags’ secrets stored by reader
Malicious traceability

An adversary $\mathcal{A}$ can distinguish two (challenge) tags over their different protocol executions
Malicious traceability

An adversary $A$ can distinguish two (challenge) tags over their different protocol executions

Tag corruption

- We consider that tags do not share secrets
- $A$ can trace this corrupted tag
Malicious traceability

An adversary $\mathcal{A}$ can distinguish two (challenge) tags over their different protocol executions.

**Tag corruption**

- We consider that tags do not share secrets
- $\mathcal{A}$ can trace this corrupted tag

**Compromised readers in offline RFID systems**

- $\mathcal{A}$ can trace all tags
- More powerful attack than tag corruption
Outline

1. Our Protocol
2. Privacy Analysis
3. Efficiency Analysis
4. Implementation
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle

CORRUPT
Our Protocol: Principle

I can differentiate them!!!

Tag 1
Tag 2
Tag 3
Our Protocol: Principle

What can we do against this problem of traceability?

Solution

- Repair the compromised reader
- Spread this info of repaired reader via tags’ mobility
Our Protocol: Design Choices

- Challenge/response authentication protocol
  - Based on Needham-Schroeder [ACM-Comm-1978]

- Public-key crypto
  - For authentication
    - Cryptosystem (Enc/Dec) for T’s answer
    - Signature scheme (Sign/Verif) for R’s identity
      ⇒ via CR certificate
  - For privacy-restoring mechanism
    - Signature scheme (Sign/Verif) for info about repaired readers
      ⇒ via NewCR/NewCT certificates

- Secret-key crypto to personalize tags’ secrets
  - Unique secret key sTR by pair (T, R)
Our Protocol: Principle

- \((P_{new}^{R}, K_{new}^{R})\)
- \(C_{new}^{R}, v_{new}^{R}\)
- \(\text{Tab}_{new}^{R} = \{ \forall T : (ID_T, s_{TR}^{new}) \}\)
- \(\text{NewC}_{new}^{R}\)
Our Protocol: Principle

- \((P_{new}^R, K_{new}^R)\)
- \(C_{new}^R, v_{new}^R\)
- \(\text{Tab}_{new}^R = \{\forall T : (ID_T, s_{new}^{TR})\}\)
- \(\text{NewC}_{new}^R\)

REPAIR
Our Protocol: Principle

- Picks a nonce $n_R$

- Checks $C_R$
  - $s_{TR} = \text{MAC}(k_T || ID_R || v_R)$
  - $E = \text{Enc}_{PR}(ID_R || n_R || s_{TR})$

- $ID_R || n_R || s_{TR} = \text{Dec}_{K_R}(E)$
  - Authenticates $T$ if $s_{TR} \in \text{Tab}_R$

- Sends $NewC_T$ ←

- Checks $NewC_T$ → Updates its values

- Sends $NewC_R$ if newer than $NewC_T$

- Checks $NewC_R$ → Updates its values
Our Protocol: Principle

- Picks a nonce $n_R$

\[ C_R, n_R \rightarrow \]

- Checks $C_R$
- $s_{TR} = \text{MAC}(k_T || ID_R || v_R)$
- $E = \text{Enc}_{P_R}(ID_R || n_R || s_{TR})$
- Sends $NewC_T$

\[ E \rightarrow \]

- Checks $NewC_T$
→ Updates its values

- Sends $NewC_R$ if newer than $NewC_T$

\[ NewC_R \rightarrow \]

- Checks $NewC_R$
→ Updates its values

\[ UPDA \]
Our Protocol: Principle

- Picks a nonce $n_R$

  - Checks $C_R$
  - $s_{TR} = \text{MAC}(k_T || ID_R || v_R)$
  - $E = \text{Enc}_{PR}(ID_R || n_R || s_{TR})$
  - Sends NewC$_T$

- $ID_R || n_R || s_{TR} = \text{Dec}_{K_R}(E)$
- Authenticates T if $s_{TR} \in \text{Tab}_R$
- Checks NewC$_T$
  - Updates its values
- Sends NewC$_R$ if newer than NewC$_T$

- Checks NewC$_R$
  - Updates its values
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle
Our Protocol: Principle

I cannot differentiate them anymore!!!

G. Avoine, I. Coisel, T. Martin – A Privacy-Restoring Mechanism for Offline RFID Systems
Privacy Analysis

Privacy experiment (from Juels and Weis’ model [Percom-2007])

1. The challenger $C$ initializes the RFID system $S$.
2. $A$ interacts with the whole system.
3. $A$ chooses two challenge tags $T$ and $T'$, and gives them to $C$.
4. $C$ chooses a random bit $b$, and assigns $T_b = T$ and $T_{b \oplus 1} = T'$.
   Then $C$ gives back $T_b$ and $T_{b \oplus 1}$ to $A$.
5. $A$ interacts with the whole system.
6. $A$ outputs a guess bit $b'$.

$A$ wins if $b = b'$.

Adversary classes

- STANDARD [$A$ can corrupt any tag (except challenge tags)]
- FORWARD [$A$ can corrupt any tag]
- CORRUPT [$A$ can corrupt any reader]
  - CORRUPT is composable with STANDARD and FORWARD
  \[ \Rightarrow \] 4 possible adversaries
Privacy Analysis

When the system is stable

- FORWARD-privacy
- CORRUPT-STANDARD-privacy

During the system update

- We define the average probability $\tau(t)$ to trace 1 tag
- When $t \uparrow$ then $\tau(t) \downarrow$

\[
\tau(t) = \left(\frac{1}{2} + \epsilon(s)\right) \left(\frac{u(t)}{n}\right) \left(\frac{u(t) - 1}{n-1}\right) \\
+ \left(1 - \frac{u(t)}{n}\right) \left(1 - \frac{u(t)}{n-1}\right) + 2 \left(\frac{u(t)}{n-1}\right) \left(1 - \frac{u(t)}{n}\right)
\]

where $u(t) =$ number of updated tags at time $t$
Case Study: 3-Day Automobile Race

Goal
Analyze in practice our privacy-restoring mechanism

Experimental conditions

- 55 readers spread all over the area
- 102 110 tags
- 1 reader has been compromised and repaired
Case Study: Tracing 1 Tag During the Event

Advantage = $|2\,\tau(t) - 1|$

- Curves depend on the update start time
- Influenced by the 1-day tickets

![Graph showing advantage over time and days]

Time $t$

- 1st day, 6AM
- 1st day, 12PM
- 2nd day, 6AM
- 2nd day, 12PM
- 3rd day, 6AM
- 3rd day, 12PM

Advantage to trace one tag
Outline

1. Our Protocol
2. Privacy Analysis
3. Efficiency Analysis
4. Implementation
Implementation

Consider the 3-day sport event data with
- 55 readers
- 10 compromised readers (at most)

<table>
<thead>
<tr>
<th></th>
<th>Our Protocol</th>
<th>⇒</th>
<th>JavaCard</th>
</tr>
</thead>
<tbody>
<tr>
<td>EEPROM</td>
<td>0.8 KB</td>
<td>⇒</td>
<td>72 KB</td>
</tr>
<tr>
<td>Transmission</td>
<td>5953 bits</td>
<td>⇒</td>
<td>68.04ms</td>
</tr>
<tr>
<td>Tag computation</td>
<td>1 PK encryption + 2 certif verifs</td>
<td>⇒</td>
<td>331.7ms</td>
</tr>
</tbody>
</table>
Conclusion

- Privacy-restoring mechanism
  - Can face the problem of compromised readers in offline systems
  - Via tags’ mobility

- Efficient protocol in a real case study
  - When attack detected at the beginning of the event
    ⇒ 99.5% of tags with a restored privacy

- Protocol deployable in practice
  - Tested and operable on JavaCard
Conclusion

- Privacy-restoring mechanism
  - Can face the problem of compromised readers in offline systems
  - Via tags’ mobility

- Efficient protocol in a real case study
  - When attack detected at the beginning of the event
    ⇒ 99.5% of tags with a restored privacy

- Protocol deployable in practice
  - Tested and operable on JavaCard

Thank You!