Security Analysis of a Privacy-preserving ECC-based Grouping-proof Protocol

Chao Lv\textsuperscript{1}, Hui Li\textsuperscript{1}, Jianfeng Ma\textsuperscript{1}, Ben Niu\textsuperscript{1}, Haiyang Jiang\textsuperscript{2}
\textsuperscript{1}Key Laboratory of Computer Networks and Information Security, Xidian University, China, lyvchao@gmail.com
\textsuperscript{2}Department of Computer Science, Tangshan Normal University, China

Abstract

Batina et al. have proposed a privacy-preserving grouping-proof RFID protocol with colluding tag prevention (CTP) recently which relies exclusively on the use of Elliptic Curve Cryptography (ECC). In this paper, we show that this proposed protocol is not secure against the tracking attack. To make this attack successfully, the adversary needs to execute three phases. Firstly, the attacker just eavesdrops on the messages exchanged between Reader and Tags. Secondly, the attacker impersonates Reader to replay the message which is got from the first phase. Finally, the adversary acts as a man in the middle to tamper the messages exchanged between Reader and Tag B. Then we propose an enhancement and prove that the revision is secure against the tracking attack which keeps other security properties.

Keywords: RFID, Authentication Protocol, ECC, Tracking Attack

1. Introduction

Radio Frequency Identification (RFID) is a wireless Automatic Identification and Data Capture (AIDC) technology for identifying a product, animal or person by using radio signals [1-3]. Due to the wide-spread usage of RFID tags and additionally, the tag limitations in terms of the circuitry (computation power), storage and power consumption, it is a great challenge to design an efficient and secure RFID authentication protocol. A well designed RFID protocol should resist the known privacy and security threats such as tag information leakage, tag location tracking, replay attacks, denial-of-service attacks and server impersonation [4].

For RFID protocols, the following security properties are the most important parts that we should consider during the processes of protocol design and analysis [5].

Mutual Authentication: the tag and the server should be authenticated with each other. The tag-to-server authentication allows the server to verify that an incoming tag is authentic. The server-to-tag authentication allows the tag to verify that it is communicating with an authorized server [6].

Untraceability: it captures the intuitive notion that a tag is untraceable if an adversary cannot distinguish whether he has seen the same tag twice [7].

Desynchronization Resistance: we say that a protocol $P$ is desynchronization resistant, if a tag never loses all its owners with respect to $P$ [8]. It ensures that the protocol runs will never evolve into a state that where is nobody who can successfully execute the protocol with a tag. In a desynchronization attack, the intruder aims to disrupt the key or the identification update, leaving tag and server’s database in a desynchronized state and rendering future authentication impossible [6].

Several theoretical models have been proposed to describe the privacy of RFID systems in the literature [9-12]. In the theoretical framework of [10], there are two characteristics of attackers: wide (or narrow) attacker and strong (or weak) attacker. The wide attacker has the ability of accessing to the result of the verification (accept or reject) in a server. If an attacker is able to extract a tag’s secret and reuse it, he is a strong attacker. Otherwise he is a weak attacker. The protocol is defined as wide-strong privacy-preserving if a protocol is untraceable against a wide-strong attacker who is the most powerful.

Most of RFID authentication protocols are focused on symmetric-key solution or hash functions lies in the common perception that public-key algorithms are time consuming to calculate and they are too complicated for such low-cost environments. However in [13], the authors have shown that Elliptic Curve Cryptography (ECC) is also suitable to be used in the designing of RFID systems. They design
an ECC-based processor which is able to use on RFID tags. Then some ECC based processors are proposed [14, 15]. In [16], the authors design the EC-RAC authentication protocol based on the Elliptic Curve Discrete Logarithm Problem (ECDLP). They claim that EC-RAC is proved secure under the generic group model and has the ability to minimize its computational workload. While in [5, 17], it is shown that EC-RAC is vulnerable to tracking attack and replay attack. The authors in [17] present that the tags can be tracked if one has eavesdropped the same tag twice and a tag can be impersonated by an adversary if it has been passively eavesdropped three times. Then a revision of EC-RAC has been proposed in [18]. However, Deursen and Radomirović [19] prove that the revision of EC-RAC in [18] is still vulnerable to tracking attack for the case that the adversary has the ability of challenging the Server utilizing the linearity of EC group operation.

In [20], the authors have proposed a privacy-preserving ECC-based grouping-proof protocol with colluding tag prevention. A pair of RFID tags (as $A$ and $B$) are able to prove that they are scanned simultaneously through this protocol. However, in this paper we show that this proposed protocol has vulnerability that it can not resist of the tracking attack under the attack of wide-strong attackers.

The rest of this paper is organized as follows. In Section 2, we introduce system model and threat model. The protocol proposed in [20] has been reviewed in Section 3. We show the security analysis in Section 4 and propose the enhancement in Section 5. Finally, we draw our conclusions in Section 6.

2. System Background

2.1. System Model

A typical RFID system architecture [21] consists of three key components: RFID tags, RFID readers and a back-end server which is shown in Fig. 1. The reader sends a radio signal to the tag and listens to the tag’s response. The tag detects this signal and replies with its identification. The reader and the tag communicate with each other through the wireless network. While the communication channel between reader and database can either be wired or wireless. Usually, we assume that the communication between server and reader is secure due to the usage of advanced encryption scheme. The wireless communication channel between reader and tag is not secure and the adversary is able to eavesdrop on it. The adversary is also able to intercept or even modify and inject the communication messages. For the grouping-proof protocol proposed in [20], we only care about the participate agents as Reader and Tags.

2.2. Threat Model

RFID technology has been widely used in numerous applications, ranging from manufacturing, logistics, transportation, warehouse inventory control, and supermarket checkout counters, to many emerging applications [22]. RFID system may face with many threats which are launched by all kinds of attackers. These attackers may be active or passive. The construction of formal RFID security and privacy frameworks is fundamental to the design and analysis of robust RFID systems.

Dolev-Yao intruder model [23] is the classical model used to analyze security protocols. Under this model, the attacker may full control over the network. The adversary is able to eavesdrop on all messages exchanged between reader and tag, modify or block any message sent from reader to tag or...
3. Review of the Protocol

In this section, we review the protocol proposed in [20]. The following notations are used in the rest of this paper. \( P \) is the base point on an Elliptic Curve (EC), and \( y \) and \( Y = yP \) are the trusted verifier’s private-key and public-key pair, where \( yP \) denotes the point derived by the point multiplication operation on the EC group. The notation \( x(T) \) is represented as the \( x \)-coordinate of the point \( T \) on the elliptic curve. \( s_t \) and \( S_j = (x_j, y_j) \) are the tag \( t \)’s private-key and public-key respectively.

The two-party CTP protocol is shown in Fig. 2. The tags and the reader will abort when a timeout occurs, or when they receive the EC point at infinity during the protocol execution. The details of protocol steps are as follows.

<table>
<thead>
<tr>
<th>Tag A</th>
<th>Reader R</th>
<th>Tag B</th>
</tr>
</thead>
<tbody>
<tr>
<td>( s_A, Y )</td>
<td>( \text{“start left”} )</td>
<td>( s_B, Y )</td>
</tr>
<tr>
<td>( r_0 \in \mathbb{Z}, T_{a,1} \leftarrow r_0P )</td>
<td>( r_s \in \mathbb{Z} )</td>
<td>( r_0 \in \mathbb{Z}, T_{b,1} \leftarrow r_0P )</td>
</tr>
<tr>
<td>( T_{a,1} )</td>
<td>( \text{“start right”,} T_{a,1}, r_s )</td>
<td>( T_{b,2} )</td>
</tr>
<tr>
<td>( T_{b,2} )</td>
<td>( \leftarrow (r_0 + x(T_{b,1})s_B)Y )</td>
<td>( T_{a,2} )</td>
</tr>
</tbody>
</table>

\[
T_{a,2} \leftarrow (r_0 + x(T_{a,2})s_a)Y
\]

\[
T_{b,1}, T_{b,2}
\]

**Figure 2.** Review of Two-party Grouping-proof Protocol

1. Reader \( R \) first sends the messages “start left” to Tag \( A \) for indicating the role of the tags in the protocol.
2. Tag \( A \) generates a random number \( r_0 \in \mathbb{Z} \) and calculates the corresponding EC point \( T_{a,1} = r_0P \). Then \( A \) replies with \( T_{a,1} \).
3. Upon receiving \( T_{a,1} \), Reader generates its own random number \( r_s \in \mathbb{Z} \) and challenges Tag \( B \) with “start right”, \( T_{a,1}, r_s \).
4. When receiving the Reader’s challenge, Tag \( B \) chooses a random number \( r_0 \) and calculates \( T_{b,1} = r_0P, \ T_{b,2} = (r_0 + x(T_{a,1})s_B)Y \). Then \( B \) responds \( R \) with \( T_{b,1}, T_{b,2} \).
5. Reader forwards \( T_{b,2} \) to \( A \). Then \( A \) calculates \( T_{a,2} = (r_0 + x(T_{b,2})s_a)Y \) and sends it to Reader.
6. At last, Reader collects the grouping proof \( (T_{a,1}, T_{a,2}, r_s, T_{b,1}, T_{b,2}) \). Then \( R \) verifies the following equations,

\[
s_A P = (y^{-1}T_{a,2} - T_{a,1})x(T_{b,2})^{-1}, \tag{1}
\]

and

\[
s_B P = (y^{-1}T_{b,2} - T_{b,1})x(r_0T_{a,1})^{-1}. \tag{2}
\]
If the public keys of $A$ and $B$ as $S_a$ and $S_b$ are registered in the database of the verifier, the grouping proof is accepted and a timestamp is added.

4. Vulnerability of the Two-party Grouping-proof Protocol

In this section, we perform the vulnerability analysis of the protocol proposed in [20] under the attack of wide-strong attacker. We describe a tracking attack to against the two-party grouping-proof protocol with colluding tag prevention. To make this attack successfully, there are three phases for the construction which is shown in Fig. 3.

(a) Phase I
A normal session takes place. The attacker is able to eavesdrop on the message $T_{a,1}, T_{a,2}, r_s, T_{b,1}, T_{b,2}$ as his knowledge.

(b) Phase II
Then the adversary impersonates Reader to challenge Tag $B$ with message “start right”, $T_{a,1}, r_s$. The elements $T_{a,1}$ and $r_s$ are got from Phase I. In this place, the attacker just replays them. Upon receiving the replayed message, Tag $B$ generates a new random number $r_b'$ and calculates $T'_{b,1} = r_b'P$ and $T'_{b,2} = (r_b' + x(r_s T_{a,1})s_b)Y$. At last, $B$ replies the message $T'_{b,1}, T'_{b,2}$.

(c) Phase III
For the last phase, the adversary carries out as a man in the middle and executes the following steps.

(1) Reader $R$ challenges Tag $A$ with “start left” once more.

(2) Tag $A$ produces a new random number $r_a'$ and calculates $T'_{a,1} = r_a'P$, then replies with $T'_{a,1}$.

(3) Upon receiving $T'_{a,1}$, $R$ generates a random number $r_a'$ and challenges Tag $B$ with $T'_{a,1}, r_a'$.

(4) Then $B$ generates its own new random number $r_b'$ and calculates $T'_{b,1} = r_b'P$ and $T'_{b,2} = (r_b' + x(r_s T_{a,1})s_b)Y$. Then $B$ responds with $T'_{b,1}, T'_{b,2}$.

(5) The attacker blocks this message which is send form Tag $B$ to the Reader. Then the adversary concludes the forged messages as

$$T_{b,1}^* = T_{b,1} + T'_{b,1} - T_{b,1} = (r_b' + r'_b - r_b)P$$

and

$$T_{b,2}^* = T_{b,2} + T'_{b,2} - T_{b,2} = (r_b' + r'_b - r_b + x(r_s T'_{a,1})s_b)Y.$$
Then Attacker sends $\tilde{T}_{b1}^*$ and $\tilde{T}_{b2}^*$ to Reader.

(6) Then Reader forwards $\tilde{T}_{b2}^*$ to Tag $A$.

(7) Upon receiving $\tilde{T}_{b2}^*$, Tag $A$ calculates $T_{a2}^* = (r_a + x(\tilde{T}_{b2}^*)r_a)Y$ and sends it to Reader.

(8) Now Reader has collected all the grouping proof messages tuple $(T_{a1}^*, T_{a2}^*, r_a, \tilde{T}_{b1}^*, \tilde{T}_{b2}^*)$. To verify the grouping proof constructed by $A$ and $B$, Reader first checks that the proof has no been used before and then verifies the following equations,

\[ s_aP = (y^{-1}T_{a2}^* - T_{a1}^*)x(\tilde{T}_{b2}^*)^{-1}, \]

and

\[ s_bP = (y^{-1}\tilde{T}_{b2}^* - \tilde{T}_{b1}^*)x(r_aT_{a1}^*)^{-1}. \]

For a wide attacker, there is one-bit extra information compared to a narrow attacker: the decision of the verifier whether to accept a tag or not [27, 28]. This extra bit information can be used by a wide-strong attacker to perform a tracking attack. For the case that $s_aP$ and $s_bP$ can be verified correctly, Reader will accept this tag. So the wide attacker is able to get this information which induces to the leakage of Tag location.

5. Our Revised Protocol

5.1. The Revision

In this section, we propose the countermeasures to resist tracking attack for this grouping-proof protocol. We revise the protocol as shown in Fig. 4. The messages $T_{b2}$ and $T_{a2}$ are changed as $T_{b2} = (r_b + x(r_aT_{a1})s_y)Y$ and $T_{a2} = (r_a + x(T_{b2}s_a)Y)$. The other message are calculated the same as before.

At last, reader $R$ verifies the following equations,

\[ s_aT_{a1} = (y^{-1}T_{a2} - T_{a1})x(T_{b2})^{-1}, \]

and

\[ s_bT_{b1} = (y^{-1}\tilde{T}_{b2} - \tilde{T}_{b1})x(r_aT_{a1})^{-1}. \]

![Figure 4. Revision of Two-party Grouping-proof Protocol](image-url)
5.2. Security Analysis

In this place, we show security analysis of the revision of two-party grouping proof protocol. The revision keeps all the other security properties and can resist the tracking attack. In the revision of protocol, we change the value of $T_{b,2}$ and $T_{a,2}$. This makes that each part of plus operation in parenthesis changed when a new session takes place. The adversary is not able to execute the attack process successfully as Section 4 because Reader R will verify the equations (7) and (8) failed. So the adversary cannot track Tag’s location.

6. Conclusion

In this paper, we have shown that a privacy-preserving ECC-based grouping proof RFID protocol which is proposed in [20] can not resist the tracking attack under a special designed three phases attack procedure. The attacking method is feasible in practice. We also propose an enhancement method to resist this kind of attack.

7. Acknowledgments

This work is supported by the following projects: National Natural Science Foundation of China grant 60772136 and 61003300, 111 Development Program of China B08038, Doctoral Fund of Ministry of Education of China 20100203110002, Fundamental Research Funds for the Central Universities JY10000901018, JY10000901021, JY10000901032 and JY10000901034.

8. References