RFID Distance Bounding Protocols

Yu-Ju Tu
Information Systems and Operations Management
University of Florida, Gainesville, FL 32611, USA
tuyuju@ufl.edu

Selwyn Piramuthu
Information Systems and Operations Management
University of Florida, Gainesville, FL32611, USA
selwyn@ufl.edu

***Abstract* – Almost all existing RFID tag/reader authentication protocols are vulnerable to mafia attacks and/or terrorist attacks from adversaries because of their inability to verify location of the tag. Several protocols have been proposed that purport to alleviate these forms of attacks. However, vulnerabilities have been identified in most of these protocols. We present and evaluate a modified distance bounding protocol.**

I. INTRODUCTION

Relay attack occurs when a valid reader (or, tag) is tricked by an adversary into believing that it is communicating with a valid tag (or, reader). These attacks are especially difficult to prevent because the adversary generally does not alter signals between tag and reader. These attacks also occur without the knowledge of tag or reader involved. These attacks are identified either through measuring signal strength or through distance bounding protocols [1]. A resourceful adversary, with the capacity to appropriately vary signal strength, easily thwarts the former. The latter operates by measuring the round trip distance between tag and reader.

Several protocols have been proposed that purport to alleviate problems due to relay attacks ([1-5]). However, these have been shown to be not completely secure against relay attacks. Relay attacks are further classified into Mafia attacks and terrorist attacks.

Mafia fraud attacks occur when an adversary inserts a rogue tag and reader between honest reader and tag respectively to relay signals between them. Brands and Chaum proposed a distance bounding protocol to prevent mafia attacks essentially through verification of the physical proximity of a tag through a series of rapid challenge-response rounds using bit exchanges ([1]). The round-trip time taken between tag and reader are then calculated. The tag is assumed to be valid if the distance between tag and reader, as calculated from the round-trip times, is within a reasonable range with respect to the speed of light. Mafia fraud attacks occur while the honest tag and reader are unaware of the process. Terrorist fraud attack, on the other hand, occurs when a dishonest tag colludes with an adversary, without sharing its secret information, to trick a reader of its physical proximity.

We briefly review existing literature in this area in the next section, and then present a modified distance bounding protocol that addresses some of the identified vulnerabilities in the following section. We conclude the paper with a brief discussion on the proposed protocol.

II. EXISTING DISTANCE BOUNDING PROTOCOLS

Since these are primarily authentication protocols, they contain an authentication part in addition to the distance measuring part. [3] presents a distance bounding protocol in which a series of rapid challenge-response rounds are implemented, and the distance between tag and reader are then calculated by the reader for each round. The tag is considered valid if the tag-reader distance is calculated to be within expected range.

The probability with which a mafia attack can occur in [3] is bounded by $(3/4)^n$, where n is the number of rapid challenge-response rounds. However, since the authentication and distance-bounding phases are not tightly linked, it is hard to ensure that the parties involved in these are one and the same. An adversary can take advantage of this and, with the help of a colluding tag, mount a terrorist attack. The colluding tag can relay all necessary information for the distance-bounding phase to the adversary, which then successfully impersonates the colluding tag to the reader. The key point to note here is that the adversary cannot infer any secret values from the information shared by the colluding tag.

The protocol presented in [5] addresses this vulnerability by ensuring that a colluding tag does not share its secrets. However, it sends the identity information of tag and reader in plaintext. A passive adversary can track the tag and reader using this information. The probability with which a mafia attack can occur is bounded by $(7/8)^n$, where n is the number of rapid challenge-response rounds.

Whereas the protocols presented in [3] and [5] involve several rapid challenge-response rounds for the distance-bounding phase, the protocol presented in [4] uses only a single-round distance-bounding phase. Similar to [5], this protocol sends the identities of tag and reader in plaintext format, and therefore is vulnerable to being tracked by an adversary. The distance-bounding phase does not involve any secret key information. The tag can, therefore, share all necessary information to the adversary, which can then impersonate the tag to the reader leading to a successful terrorist attack.

Most protocols designed to resist relay attacks measure round trip distance between tag and reader. However, the protocol presented in [2] measures just the one-way distance between tag and reader. They claim that this allows for relatively complex computations at the tag-level since it is not constrained to respond instantaneously. Instantaneous response is paramount in protocols with round-trip distance measurements since ideally the time taken for computation should be less than the travel-time between tag and reader for measurement precision purposes. In their protocol, the sent and reception times of signals at the tag and reader are used to calculate one-way times. The tag is also assumed to have an internal clock. Generally speaking, a passive tag cannot have an internal clock due to unavailability of internal power. This restriction can, however, be alleviated through other means such as by measuring the discharge rate of capacitor(s) on-board the tag. A resourceful adversary can tweak the time taken for computation, by modifying the reception and sent times, to impersonate a valid tag to the reader. Instead of one reader, which is commonly used in such protocols, they consider a set (e.g., 3) of readers to measure distance accurately through triangulation.

III. PROPOSED MODIFIED PROTOCOL

We use some of the principles that were used in [1], [3], and [5] to develop the proposed modified framework. We therefore

(a) include both timed and un-timed phases where the timed phase is used to verify distance between tag and reader and the un-timed phase is used to authenticate tag and reader,

(b) generate fresh nonces for every round of the protocol to prevent replay attacks

(c) use nonces generated by both tag and reader to render it hard to impersonate tag or reader

(d) use expressions in the bit-wise fast challenge-response phase such that these, when known to the adversary, can be used to reveal the secret key.

(e) avoid sending nonces as plaintext.

(f) incorporate periodic (4 times here) verification of reader by the tag to prevent an adversary from retrieving a large part (e.g., half) of the answers to the challenges from the tag by impersonating the valid reader.

We use the following notations to describe the proposed modified protocol.

- r_A, r_B : random n-bit vectors
- x : n-bit shared-secret vector
- h : hash function – $\{0,1\}^* \rightarrow \{0,1\}^n$
- || : concatenation operator
- ⊕ : Exclusive-OR (XOR)
- t_s, t_f : start and finish times respectively

We assume n to be divisible by 4. The reader and tag share a secret key x. The proposed modified protocol is given below.

```
Reader (secret x)                                            Tag (secret x)
$r_B \leftarrow \{0,1\}^n$                --$x \oplus r_B$→            $r_A \leftarrow \{0,1\}^n$
                                           ←$x \oplus r_A$--
$k = h(r_A, x \| r_B)$                                          $k = h(r_A, x \| r_B)$
$c = k \oplus x$                                                $c = k \oplus x$
For $(u,v) = \{(r_A, r_A), (r_B, r_B), (r_B, r_A), (r_A \oplus r_B, r_A \oplus r_B)\}$
    For $i = 1..(n/4)$
        $q_i \leftarrow \{0, 1\}$
        Start clock ($t_s$)                -- $q_i$ →                 If $q_i = 0$, $C_i = k_i$
                                                                      If $q_i = 1$, $C_i = c_i$
        End clock ($t_f$)                  ← $C_i$ --
        Check $C_i$, $\Delta t = (t_f - t_s)/2$
        If $C_i$, $\Delta t$ invalid, abort process.
    End for
    $k_{temp} = h(u, x \| v)$               -- $k_{temp}$ →        verify $k_{temp} = h(u, x \| v)$
                                                                   If invalid, abort process
End for
```

Here, the outer (quarter) for-loop iterates for four different combination values of (u,v) (of $(r_A, r_{A)},$ (r_B, r_B), (r_B, r_A), $(r_A \oplus r_B , r_A \oplus r_B)$) while the inner loop iterates n/4 times. The tag is validated in the inner loop through C_i and Δt. This can be done in the background while the fast bit exchange occurs between tag and reader. Using k_{temp}, the reader is validated by the tag after every iteration of the inner loop. Each of the bits here can also be streamed and clocked to measure distance between tag and reader. The probability with which a mafia attack can occur is bounded by $(9/16)^n$. This is because an adversary, not knowing k_{temp}, can impersonate the reader to the tag and retrieve C_i values for a given q_i (say, 0) until the tag verifies k_{temp} at the end of the first quarter (u,v) loop (i.e., (u,v)= (r_A, r_A)). This can be operationalized by the adversary as follows: capture $x \oplus r_A$ from the tag and hold it while obtaining the C_i values from the tag. Once this has happened, the adversary can relay $x \oplus r_A$ to the reader and wait for the fast bit exchange challenge from the reader.

Since q=0 or 1 on an average of (n/2) times, the adversary can impersonate the tag to the reader with a probability of (1/2) during each iteration. However in the first outer loop, this probability is ¾ since the adversary has the correct values for half the C_i on average and the other half can be guessed correctly half the time on average The probability of correctly guessing any given C_i is therefore ¼*(¾ + ½ + ½ + ½) = 9/16. This process proceeds for n iterations, hence the superscript value of n in the probability expression. This probability calculation ignores the fact that C_i and Δt are both validated during every iteration of the inner loop and the process is aborted if any inconsistency is found. The actual probability, therefore, would be much less than $(9/16)^n$. This probability is less than those for similar published protocols.

We use both timed and untimed phases as in [1], [3], and [5]. We use freshly generated nonces by both tag and reader during every round, and send it XORed with the secret key (x) to avoid revealing the nonces to outside entities. We use two loops within each round to introduce more authentication steps and make it difficult for an adversary. The only purpose of the outer loop is to abort the authentication process if the reader cannot authenticate itself to the tag. We can vary the number of times (here, 4) this is done per round to vary the probability of an adversary learning responses to challenges. Generating C_i from both c_i and k_i prevents terrorist attacks by ensuring that the colluding tag does not share its secrets with an adversary. Here, the secret (x) can be retrieved from simultaneous knowledge of both c and k.

The round trip times (Δt) are used to verify the distance between tag and reader. The authentication process is aborted by the tag as well as the reader. The reader aborts the authentication process when response from the tag (C_i) are found to be invalid or when the round trip time is found to be longer than expected. The tag aborts the authentication process when k_{temp} is found to be invalid. By aborting the process when something is invalid or unusual, the chances of an adversary causing harm is reduced.

The overall probability can be further reduced by splitting k and c into k_a, k_b and c_a, c_b respectively and with the following modifications in the proposed algorithm.

```
$k = k_a \| k_b$                                        $k = k_a \| k_b$
$c = c_a \| c_b$                                        $c = c_a \| c_b$
      ...
Start clock                  -- $q_i$ →       If $q_i = 0$, $C_i = \{k_a\}^i \| \{k_b\}^i$
                                                If $q_i = 1$, $Ci = \{c_a\}^i \| \{c_b\}^i$
End clock                    ← $C_i$ --
```

This would necessitate sending 2 bits (instead of 1) during the rapid challenge-response phase.

IV. Discussion

We presented a modified distance bounding protocol that is resistant to terrorist attacks. We used principles from [1], [3], and [5] as well as identified vulnerabilities in these protocols to develop the proposed protocol. We believe that the proposed modified protocol addresses some of the vulnerability issues in the protocols presented in [1], [3], and [5]. Although we have reduced the probability of mafia attack, vulnerability still remains. I.e., the probability is not zero. We used one reader in the proposed protocol. There is evidence in the literature that use of multiple readers can decrease the probability of relay attacks through triangulation. There is a need to study the underlying dynamics of distance bounding protocols, mafia attacks, and terrorist attacks to develop better protocols that resist all variations of relay attack.

References

[1] S. Brands and D. Chaum. Distance-Bounding Protocols. *Advances in Cryptology - EUROCRYPT'93*, Lecture Notes in Computer Science 765: 344-359, 1993.

[2] S. Capkun and J.-P. Hubaux. Secure Positioning in Wireless Networks. *IEEE Journal on Selected Areas in Communications*, 24(2), February, 2006.

[3] G.P. Hancke and M.G. Kuhn. An RFID Distance Bounding Protocol. *Proceedings of the IEEE/Create-Net SecureComm*, 67-73, 2005.

[4] C. Meadows, R. Poovendran, D. Pavlovic, L.W. Chang, and P. Syverson. Distance Bounding Protocols: Authentication Logic Analysis and Collusion Attacks. *Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks*, 279-298, Springer-Verlag, 2007.

[5] J. Reid, J.M.G. Nieto, T. Tang, and B. Senadji. Detecting Relay Attacks with Timing-Based Protocols. *Proceedings of the 2*nd *ACM Symposium on Information, Computer, and Communications Security*, 204-213, 2007.