HB and Related Lightweight Authentication Protocols for Secure RFID Tag/Reader Authentication*

Selwyn Piramuthu
Decision and Information Sciences
University of Florida, Gainesville, Florida 32611-7169.
Email: *selwyn@ufl.edu*

Abstract

Lightweight authentication protocols are necessary in Radio-Frequency Identification (RFID) applications due to tag-level constraints. Over the past few years, several such protocols have been proposed and analyzed. We focus on the HB protocol and its variants. We show the vulnerability of some of these to attacks on tags, where the adversary pretends to be a valid reader, and propose a modified protocol that avoids this type of attack.

1 Introduction

Radio-Frequency Identification (RFID) tags are poised to supplant barcodes in the very near future. Advantages of RFID tags over barcodes are many including their capacity to store more information, and the ease with which they can be read since they don't require line of sight (Finkenzeller, 2002). The primary impediment to their widespread use is their cost. Privacy and security issues also play a major role in the success of RFID tag implementations due to the ease with which the object they are attached to can be identified and/or tracked by an adversary.

When dealing with privacy/security issues in RFID tag implementations, their processing power and memory constraints dictate lightweight authentication protocols. Several researchers have proposed and evaluated protocols that fit the bill of being lightweight and at the same time being secure to a reasonable extent (Avoine and Oechslin, 2005; Dimitriou, 2005; Weis et al, 2004).

One such is the HB protocol that was proposed by Hopper and Blum (2001). Although this protocol works well under most circumstances where passive adversaries exist, an active adversary can break its secureness. Juels and Weis

*CollECTeR Europe Conference, Basel, Switzerland, 9-10 June 2006.

(2005) modified HB to include protection against active attacks from adversaries. However, this (HB^+) too was not completely secure under certain circumstances (Gilbert et al. 2005). Bringer et al. (2006) later modified HB^+ to secure it against active attacks from adversaries as described in Gilbert et al. (2005). We show that the protocols (HB^{++} and HB^{++} [first attempt]) presented in Bringer et al. (2006) are vulnerable to active attacks, and present a modified solution.

This paper is organized as follows: we provide a brief overview of HB and its variants in the next section. This is followed by suggested modifications to a recent variant of the HB protocol (HB^{++}). Section 3 includes brief security analysis of the proposed modifications, and Section 4 concludes the paper.

2 HB and its variants

We briefly describe and evaluate HB and its variants in this section. After providing a brief introduction to the protocols, we consider some security violations that may occur and discuss possible remedies that were provided in the literature. We then propose modifications to HB^{++}, a recent variant of HB.
Notations Used:

- a, b: random k-bit binary vectors
- x, x', y, y': k-bit secret key vectors
- ν: noise bit (=1 with probability $\eta \in [0, \frac{1}{2}]$)

2.1 HB

Tag (secret x) $\nu \in \{0,1 \mid \mathcal{P}(\nu = 1) = \eta\}$		**Reader (secret x)**
	$\xleftarrow{a}$	Generate challenge $a \in_R \{0,1\}^k$
Compute $z = a \cdot x \oplus \nu$	$\xrightarrow{z}$	Check $a \cdot x \approx z$

Figure 1: A round of HB protocol (Hopper and Blum, 2001)

An overview of a round of HB protocol is given in Figure 1. Here, $a \cdot x$ and $a \oplus x$ represent scalar product and exclusive-or (XOR) of k-bit binary vectors a and x respectively. The HB protocol relies on the computational hardness of Learning Parity with Noise (LPN) problem, and not on classical symmetric key cryptography solutions (Bringer et al., 2006). It is meant only to be secure against passive attacks, and it is not secure against active attacks. The round given in Figure 1 is repeated r times and the tag is authenticated if check on the reader's side fails at most ηr times. A simple active attack where an adversary

pretending to be the reader transmits a fixed a to the tag several times can retrieve the value of x.

2.2 HB$^+$

Tag (secret x, y**)** $\nu \in \{0,1 \mid \mathcal{P}(\nu = 1) = \eta\}$		**Reader (secret** x, y**)**
Generate blinding vector $b \in_R \{0,1\}^k$	$\xrightarrow{b}$	
	$\xleftarrow{a}$	Generate challenge $a \in_R \{0,1\}^k$
Compute $z = a \cdot x \oplus b \cdot y \oplus \nu$	$\xrightarrow{z}$	Check $a \cdot x \oplus b \cdot y \approx z$

Figure 2: A round of HB$^+$ protocol (Juels and Weis, 2005)

Juels and Weis (2005) modified the HB protocol and showed the modified protocol (HB$^+$) to be secure against active attacks. A round of HB$^+$ is given in Figure 2. They introduced another k-bit vector secret key (y) that is shared between the reader and a tag. They also modified the HB protocol such that the tag, and not the reader, initiates the authentication process. The tag first transmits a k-bit blinding vector to the reader. The other modification is in the way z is computed. A scalar product of the newly introduced secret key (y) and the blinding vector (b) is XOR-ed with the z in HB.

Tag (secret x, y**)** $\nu \in \{0,1 \mid \mathcal{P}(\nu = 1) = \eta\}$		**Reader (secret** x, y**)**
Generate blinding vector $b \in_R \{0,1\}^k$	$\xrightarrow{b}$	
	$\xleftarrow{a' = a \oplus \delta} \cdots \xleftarrow{a}$	Generate challenge $a \in_R \{0,1\}^k$
Compute $z' = a' \cdot x \oplus b \cdot y \oplus \nu$	$\xrightarrow{z'}$	Check $a \cdot x \oplus b \cdot y \approx z'$

Figure 3: The attack on a round of HB$^+$ protocol (Gilbert at al., 2005)

Although Juels and Weis (2005) showed HB$^+$ to be secure against active attacks, Gilbert et al. (2005) showed that HB$^+$ is not secure against a simple man-in-the-middle attack that was not considered in former. A description of this attack is given in Figure 3. Here, the adversary is assumed to be capable of manipulating challenges sent by a legitimate reader to a legitimate tag during the authentication process. The adversary is also assumed to have the capability to recognize when an authentication procedure succeeds or fails. The core of the attack consists of manipulating the challenge sent by the reader (a) by sending the XOR of a and a constant k-bit vector δ to the tag on all r rounds of the

authentication process. If the authentication succeeds, $\delta \cdot x = 0$ with a high probability. If the authentication fails, $\delta \cdot x = 1$ with a high probability. Here, one can manipulate δ to reveal each bit of the secret key x one by one. The protocol can be repeated k times to retrieve all bits of the secret key k.

Once x is identified, the adversary can impersonate the tag and send a given blinding vector b to the reader. In return to the reader's response, the adversary can transmit $a \cdot x$ to the reader. When authentication succeeds, the adversary knows that $b \cdot y = 0$ with high probability. On the other hand, if authentication fails the adversary knows that $b \cdot y = 1$ with high probability. Using these, the adversary is able to derive the other secret key y. When both the secret keys (x, y) are revealed to the adversary, the privacy of this tag is under threat.

The HB$^+$ protocol is not secure against another form of attack from an adversary who pretends to be a valid tag reader. Here, the adversary is assumed to have the capability to intercept all communication between a tag and a reader, and the ability to block transmission of any communication from the tag to the reader. The adversary is also assumed to be capable of transmitting a to the tag before the tag can initiate the next round with a new b value. I.e., the reader repeatedly transmits a to the tag, keeping it busy computing zs. The following should also work with different bs during different rounds of the protocol.

When the authentication process begins as the tag sends b, the adversary intercepts it and transmits $a = 0$ to the tag. The tag then computes $z(= b \cdot y \oplus \nu)$ and transmits it to the reader, which in this case is the adversary. The process can be repeated enough number of times until $b \cdot y$ is retrieved. Since b is known to the adversary, y can be inferred. Knowing a, b, and y, the process can be repeated until x is identified. The adversary takes advantage of the fact that communication between the tag and the reader is controlled by the reader - i.e., when the authentication succeeds, the protocol ends. Since the adversary is the reader during this attack, it can continue with as many rounds of the protocol as is necessary until it succeeds in indentifying the secrets (x, y).

In the effective attack against HB$^+$ proposed by Gilbert et al. (2005), the tag and reader are authenticated by the time the secrets are known to the adversary. It is possible that this tag, once authenticated, may leave the "system" and would no longer interact with any reader. Under these circumstances, the attack presented in the previous two paragraphs has an edge since it happens without any interaction between tag and reader.

2.3 HB^{++}[first attempt] and HB^{++}

In response to Gilbert et al.'s (2005) attack on HB$^+$, Bringer et al. (2006) proposed two protocols (HB^{++}[first attempt] (Figure 4) and HB^{++} (figure 5)) that secures against such man-in-the-middle attacks. However, these protocols are still not immune to attacks from adversary that pretends to be an authentic reader. HB^{++}[first attempt] was shown to not be immune to attacks in Bringer et al. (2006).

Another vulnerability arises from the fact that the protocols HB^{++} and HB^{++}[first attempt] contain z as in HB$^+$, and this z can be used to identify

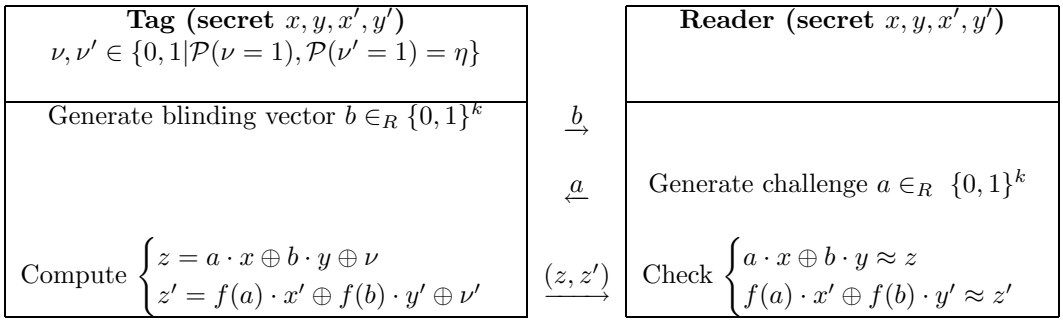

Figure 4: A [first attempt] round of HB^{++} protocol (Bringer at al., 2006)

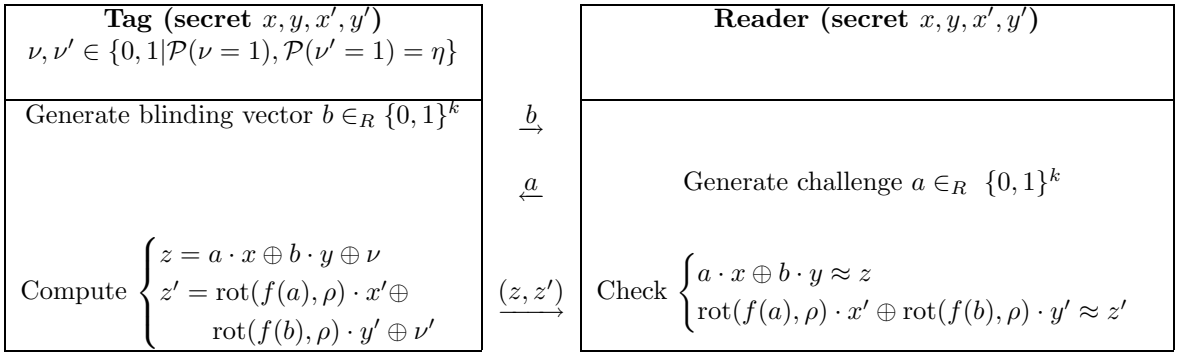

Figure 5: A round of HB^{++} protocol (Bringer at al., 2006)

the secrets x and y. This is a vulnerability since the adversary can track the tag knowing only z, which creates a "side channel" (Phan and Yen, 2006). The adversary can easily compute z using a man-in-the-middle attack (Gilbert et al., 2005) or by pretending to be a valid reader as discussed earlier in this section.

Ignoring the possibility of this "side channel" attack, the adversary still needs to identify the other two secrets (x', y'). Here, the adversary is assumed to have the capability to receive transmissions from the tag and block these transmissions from reaching the reader. The adversary can manipulate a to its advantage to retrieve the secret keys (x', y'). Initially, the adversary can transmit $a = 0$ to determine $f(b) \cdot y'$. Once this is accomplished, the adversary can set $a = 1$ to determine $1 \cdot x' \oplus f(b) \cdot y'$. The adversary can then identify the secret x' from $f(b) \cdot y'$ and $1 \cdot x' \oplus f(b) \cdot y'$.

During the next several rounds of the protocol, the adversary can assign $a = b$ and determine $f(b) \cdot (x' \oplus y')$. Knowing this, $f(b) \cdot y'$, and x', the adversary can determine y'.

Bringer et al (2006) modified HB^{++}[first attempt] to rectify a vulnerability they identify and propose HB^{++} (Figure 5) that has protection against these vulnerabilities. However, HB^{++} too is prone to a similar attack where an adversary pretends to be a valid reader. Here, again, z can be used to retrieve

the secrets x and y as in HB$^+$. To retrieve the other two secrets (x', y'), we make the following assumption: ρ is updated only once during a round, and a round for this purpose is defined as beginning with transmission of b by the tag and ending with the checking of z and z' by the reader. Updates to ρ probably occurs at the beginning of each round (Bringer et al., 2006). However, a fast adversary can communicate with the tag several times in-between two successive transmissions of b by the tag.

As long as b and ρ remain constant when the adversary is communicating with the tag, the following would help reveal the other two secrets (x' and y'). The adversary is assumed to have the capability to intercept b transmitted by the tag and to prevent b from reaching the reader. The adversary can manipulate a to its advantage to retrieve the secret keys (x', y'). Initially, the adversary can transmit $a = 0$ to determine $\text{rot}(f(b), \rho) \cdot y'$. Once this is accomplished, the adversary can set $a = 1$ to determine $1 \cdot x' \oplus \text{rot}(f(b), \rho) \cdot y'$. The adversary can then identify the secret x' from $\text{rot}(f(b), \rho) \cdot y'$ and $1 \cdot x' \oplus \text{rot}(f(b), \rho) \cdot y'$.

During the next several rounds of the protocol, the adversary can assign $a = b$ and determine $\text{rot}(f(b), \rho) \cdot (x' \oplus y')$. Knowing this, $\text{rot}(f(b), \rho) \cdot y'$, and x', the adversary can determine y'.

2.4 Proposed modifications to HB^{++}

Tag (secret x', y')		**Reader (secret x', y')**
$\nu' \in \{0,1 \mid \mathcal{P}(\nu' = 1) = \eta\}$ initialize $\rho = 0$		
Generate blinding vector $b \in_R \{0,1\}^k$	$\xrightarrow{b}$	
	$\xleftarrow{a}$	Generate challenge $a \in_R \{0,1\}^k$
$\rho \longleftarrow \rho + 1$ Compute $z' = \text{rot}(f(a), \rho) \cdot x' \oplus$ $\text{rot}(f(b), \rho) \cdot y' \oplus \nu'$	$\xrightarrow{(z')}$	Check $\text{rot}(f(a), \rho) \cdot x' \oplus$ $\text{rot}(f(b), \rho) \cdot y' \approx z'$

Figure 6: A round of Modified HB^{++} protocol

In order to maintain the proofs of security for the adversary model presented in Juels and Weis (2005), the proposed modifications keep the crux of HB$^+$ intact. The main modifications are as follows:

- Removal of z and related vectors (x, y) and ν. This is to prevent "side channel" attack mentioned earlier. A side effect of this keeps the protocol 'lightweight.'
- Update ρ every time z is computed. This is to prevent usage of the same ρ before initiation of the next round from the tag's end.

3 Security Analysis

Following Dimitriou (2005), we present a brief security analysis on the proposed modifications to HB^{++}.

Attack on a tag. This type of attack refers to the scenario where an adversary pretends to be the reader. Since the protocol is based on avoiding exactly this type of attack, it is hoped that the adversary will not be able to succeed.

Attack on the reader. Here, the adversary pretends to be a valid tag. This type of attack will not succeed because of the shared secret keys (x' and y').

Attack on the communication between tag and reader. An adversary can block messages between the reader and a tag. When this happens, the authentication process is broken and it doesn't succeed. Repeated interactions, either blocking transmissions or through man-in-the-middle attacks, leaves the door open for an adversary to learn the secret keys.

Attack on user privacy. Since no 'private' information is transmitted during validation, this is of no concern here.

Attack on location privacy. This is of concern since the secrets do not change over different runs of the protocol.

Attack against the key. This happens when an attacker listens in on the transaction and tries to identify the key values. Again, if the keys are selected appropriately (e.g., Lenstra and Verheul, 2001), this is not of concern.

Attack against implementation. Provided the keys and the random numbers are genereted with caution, this is not of concern.

Disassembling the tags. These tags are clearly not tamper-resistant, and can be disassembled to retrieve the structure of z' as well as the secret keys (x', y').

4 Conclusion

We discussed and evaluated HB and its variants for RFID tag/reader authentication. In addition to the security compromises that have been mentioned in the literature, we provided yet another security compromise that can occur due to an adversary pretending to be a valid reader. This threat is worse since the valid reader is not involved when the adversary interacts with the tag to identify the secret keys. We also showed the vulnerability of a recent variant of HB (HB^{++}), and presented a means to avoid this vulnerability. Although the proposed method is not secure against all types of attack by an adversary, it

is reasonably secure against those that were considered while maintaining the 'lightweight' characteristic of the protocol.

References

[1] G. Avoine and P. Oechslin. "RFID Traceability: A Multilayer Problem," *Financial Cryptography - FC'05*, LNCS, Springer, 2005.

[2] J. Bringer, H. Chabanne, and E. Dottax. "HB^{++}: a Lightweight Authentication Protocol Secure Against Some Attacks," *IEEE International Conference on Pervasive Services, Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing – SecPerU*, 2006.

[3] T. Dimitriou. "A Lightweight RFID Protocol to Protect Against Traceability and Cloning Attacks," *Proceedings of the IEEE International Conference on Security and Privacy for Emerging Areas in Communication Networks - SECURECOMM*, 2005.

[4] K. Finkenzeller. *RFID Handbook*, second edition, Wiley & Sons, 2002.

[5] H. Gilbert, M. Robshaw, and H. Sibert. "An Active Attack Against HB^{+} - A Provably Secure Lightweight Protocol." *Cryptology ePrint Archive, Report 2005/237, 2005. http://eprint.iacr.org.*

[6] N.J. Hopper and M. Blum. "Secure Human Identification Protocols." in C. Boyd (ed.) *Advances in Cryptology - ASIACRYPT 2001*, Volume 2248, *Lecture Notes in Computer Science*, pp. 52-66, Springer-Verlag, 2001.

[7] A. Juels and S. Weis. "Authenticating Pervasive Devices with Human Protocols," in V. Shoup (ed.) *Advanced in Cryptology - CRYPTO'05*, Volume 3126, *Lecture Notes in Computer Science*, pp. 293-308, Springer-Verlag, 2005.

[8] A. Lenstra and E. Verheul. "Selecting Cryptographic Key Sizes," *Journal of Cryptography*, 14(4), pp. 255-293, 2001.

[9] R. C.-W. Phan and S.-M. Yen. "Amplifying Side-Channel Attacks with Techniques from Block Cipher Cryptanalysis," *Proceedings of the 7th Smart Card Research and Advanced Application IFIP Conference - CARDIS'06*, 2006.

[10] S. Weis, S. Sarma, R. Rivest, and D. Engels. "Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems," *Proceedings of the 1st Security in Pervasive Computing*, LNCS, volume 2802, pp. 201-212, 2004.