Analysis and Improvement of the 802.11i 4-Way Handshake○

In-Hwan Kim[1] , Sung-Hyun Eum[2], and Hyung-Kee Choi[3]
[1] School of Information and Communication Engineering, Sungkyunkwan University
Suwon, South Korea
[2] School of Information and Communication Engineering, Sungkyunkwan University
Suwon, South Korea
[3]School of Information and Communication Engineering, Sungkyunkwan University
Suwon, South Korea
E-mail: {[1]ihkim, [2]sheum, [3]hkchoi} @ece.skku.ac.kr

Abstract: Wireless LAN (WLAN) is a type of wireless service that has higher data transmission than current networks. The usage is continually increasing. There are many vulnerabilities in wireless network, due to the properties of the wireless environment, regardless of its popularity. IEEE announced the 802.11i security standard to solve these problems. The vulnerable point of messages used in the process of key distribution for 802.11i makes the target node attacked lose memory through continuous messages and blocks the legitimate WLAN service. In this paper, we proposed a new scheme to solve this problem. We analyze this and compare our proposal with the current process.

1. Introduction

Wireless Local Area Network (WLAN) is widely deployed. WLAN has the advantages of mobility and a low installation cost, and it gives reasonable promise of sustainable growth. Despite its current popularity, WLAN inevitably comes under attack. Due to the nature of wireless communication, sensitive data is vulnerable to exposure to a third party. To solve these problems, IEEE has published the 802.1x standard to provide secure communication in WLAN. Personal privacy was secured partially by the 802.1x protocol and its vulnerabilities to man-in-the-middle and Denial-of-Service (DoS) attacks have been identified. IEEE announced the 802.11i [1] security standard to enhance WLAN security. There are many security technologies in the 802.11i standard. These include strong encryption, mutual authentication procedure, and the 4-way handshake that is the session key derivation procedure. The 802.11i 4-way handshake has some weaknesses. One of them may cause DoS attack in the session key derivation procedure of 802.11i. If an attacker sends a number of forged messages to a client, legitimate WLAN services may become unavailable due to depleting client memory.

In this paper, we focus on the 802.11i 4-way handshake. We analyze the security vulnerability of the 4-way handshake and show an attack scenario in 802.11i. Finally, we propose two solutions to prevent the attack and compare performance of the proposed solutions with the 802.11i 4-way handshake.

The paper is organized as follows. Section 2 describes related work for 802.11i and 802.11i 4-way handshake. Section 3 explains the 802.11i 4-way handshake and analyzes its security. Section 4 proposes our improved version of 802.11i 4-way handshake. Performance analysis using simulation and security analysis are given in section 5. Section 6 concludes the paper.

2. Related Work

The IEEE 802.11i standard defines a mutual authentication and session key distribution mechanism in order to enhance the security in wireless LAN. Many researchers analyze 802.11i vulnerabilities to solve these problems.

Chanhua He and his team analyzed possible attacks in 802.11i 4-way handshake and all 802.11i processes. In the 4-way handshake, a client and an AP exchange parameters to derive a temporary session key. The first message in 4-way handshake is not protected by the 802.11i protocol, because the session key is not derived until two messages are exchanged. They proposed a solution to protect from these attacks, to design an efficient and secure 802.11i process [2], [3]. Absence of a key allows a DoS attack to take place. An attacker may transmit a number of the first messages, thereby deceiving a client into receiving the messages, as if they came from legitimate APs.

Hayriye Altunbasak and his team focused on reducing the number of exchange messages in a 4-way handshake in order to reduce latency and computational overhead. They analyzed that role of the last message in the 4-way handshake as minor and proposed a 3-way handshake without the last message. They further proposed a 2-way handshake that protects the first message in the 4-way handshake [4].

Romano Fantacci focused on the handover in the WLAN environment and proposed a fast authentication protocol reusing a secret key acquired from a former authentication process [5].

3. Analysis of the 802.11i 4-Way Handshake

The primary roles of the 802.11i 4-way handshake are to verify existence of the same Pairwise Master Key (PMK) between an AP and a client and to derive the Pairwise Transient Key (PTK). PMK is shared key between the AP and client that generated in the former process of 4-way handshake of 802.11i. PTK is session key between the

○ "This research was supported by the MKE(Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Advancement)" (IITA-2008-C1090-0801-0028)

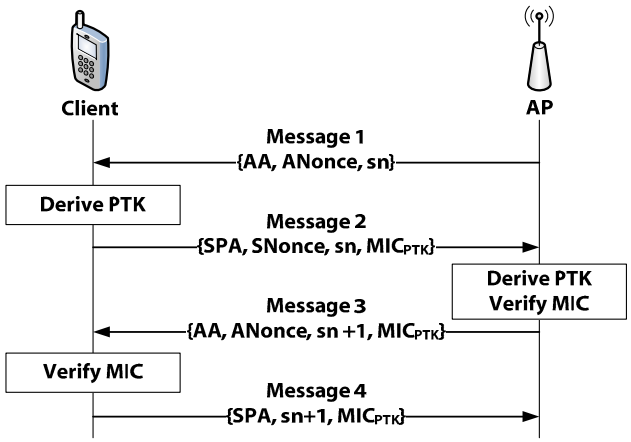

Figure 1. 4-Way Handshake in 802.11i

AP and client. The 4-way handshake consists of four messages, Message 1 to Message 4, as shown in Figure 1.

3.1 Description of the 4-Way Handshake

The 4-way handshake is initiated when an AP sends Message 1 to a client. Message 1 consists of three parameters; the MAC address of the AP (called AA), a random number chosen by the AP (called ANonce), and a counter to prevent a replay attack (called sn).

After receiving Message 1, the client generates two parameters; the MAC address of the client (called SPA) and a random number chosen by the client (called SNonce). The client derives PTK from five parameters, i.e. AA, ANonce, SPA, SNonce, and PMK. The client sends Message 2 that includes SPA, SNonce, sn, and Message Integrity Code (MIC) to the AP.

The AP derives PTK after receiving Message 2 and verifies MIC using PTK. If the integrity of Message 2 is verified, the AP trusts that the client has the same PTK. The AP sends Message 3, which is quite similar to Message 1, except that Message 3 includes MIC.

After receiving Message 3, the client verifies MIC in Message 3 using PTK. If the integrity of Message 3 is verified, the client trusts that the AP has the same PTK. Message 4 simply plays the role of the acknowledgment of Message 3.

3.2 Security Analysis of the 4-Way Handshake

The client must accept another Message 1, even if it has already received Message 1 in the 4-way handshake procedure, because the client is stateless in the 4-way handshake process. The client only checks the duplicate message of the first message. The AP and the client cannot share PTK, before receiving Message 2, so Message 1 cannot be protected by using PTK. If an attacker sends a forged Message 1 involving different ANonce between the legitimate Message 1 and Message 2, as shown in Figure 2, 4-way handshake blocking occurs at the client.

To prevent this attack, the client must generate Temporary PTK (TPTK) for every received Message 1 and install TPTK only after verifying MIC of Message 3. In this case, the client must allocate memory for every Message 1 [3].

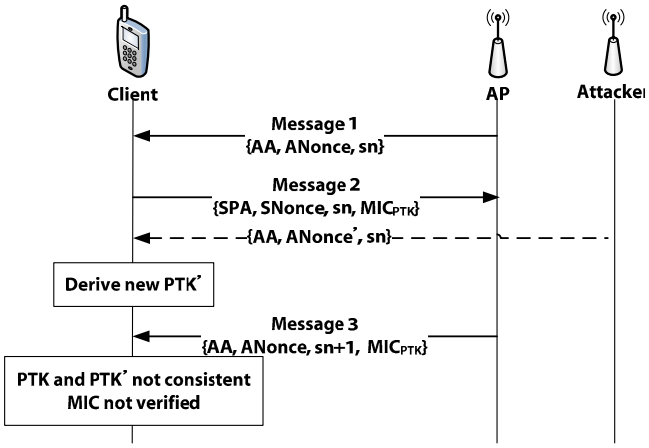

Figure 2. 4-Way Handshake Blocking

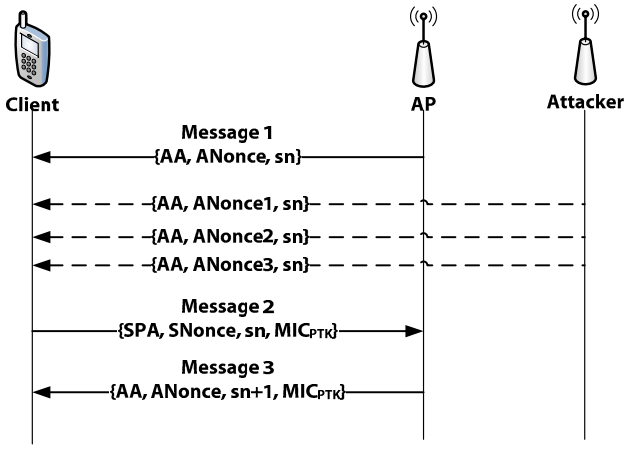

Figure 3. Memory Exhaustion Attack

Thus, if an attacker sends a lot of forged Message 1 involved different ANonce between the legitimate Message 1 and Message 2, as shown in Figure 3, memory exhaustion occurs in the client.

An attacker can easily conduct the memory exhaustion attack, because the number of Message 1s can theoretically be unbounded. If the attack occurs, all of the 802.11i authentication processes prior to the 4-way handshake are cancelled.

4. The Proposed solutions

In this paper, we propose two solutions to prevent the DoS attack on the client. That is, the client does not store both ANonce and PTK in the proposed solutions. We call the first solution SNonce Response, and the second solution, PTK Cookie. We describe the two solutions as follows.

4. 1 SNonce Response

In the SNonce Response the AP sends the modified Message 3 containing SNonce, included in Message 2, as shown in Figure 4.

Initially, the AP sends Message 1 to the client. Message 1 consists of three parameters: AA, ANonce and sn. When the client receives Message 1, it generates two additional parameters: SPA, SNonce. Then the client derives PTK from five parameters, i.e. AA, ANonce, SPA, SNonce, and

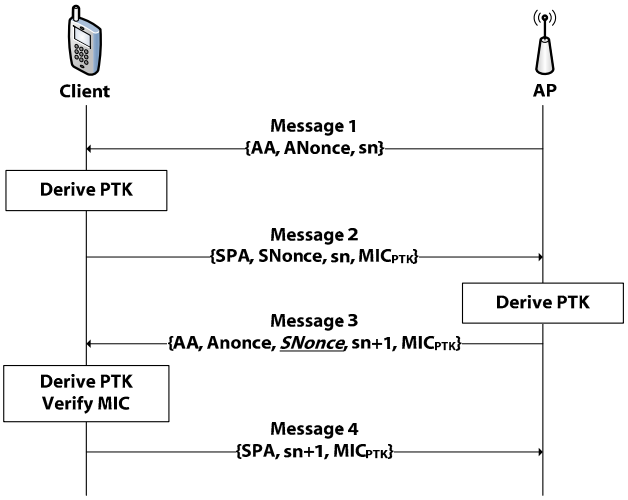

Figure 4. SNonce Response

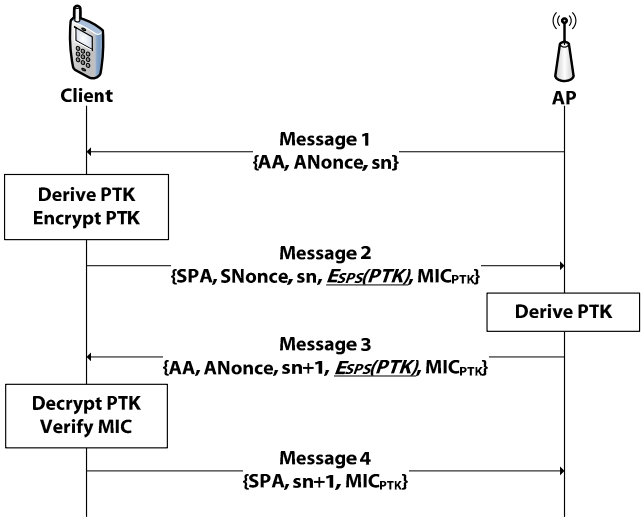

Figure 5. PTK Cookie

PMK. The client sends Message 2 that includes SPA, SNonce, sn, and MIC to the AP. After receiving Message 2, the AP also derives the PTK. By using the MIC in Message 2, the AP can verify that it has the same PTK as the one derived by the client. Then the AP sends the modified Message 3 including AA, Anonce, SNonce, MIC, and sn+1 th the client. When the client receives the modified Message 3, the client can generate PTK, because the message contains the parameters for deriving PTK, i.e. SNonce, ANonce, and AA. The client verifies the MIC of the modified Message 3 using PTK and then installs PTK.

The client does not need to allocate memory for every Message 1, because the client is able to generate PTK from the modified Message 3, without Message 1.

4. 2 PTK Cookie

PTK Cookie uses cookies [6]. Like SNonce Response, this approach does not allocate memory for every Message 1.

After receiving Message 1, the client generates a random secret (called SPS), and encrypts PTK with SPS, as shown in Figure 5. SPS is generated by client and is used by only the client. The length of this secret must be appropriate for encryption. The client sends the modified message 2, including the encrypted PTK, to the AP. After receiving Message 2 from the client, the AP sends the modified Message 3, including the encrypted PTK, received in Message 2 from the client. When client receives the modified Message 3, the client decrypts the encrypted PTK with SPS. The client verifies the MIC of Message 3 using PTK, and then installs PTK. The client does not need to allocate memory for every Message 1, because the client gets PTK from the modified Message 3.

As a result, these solutions prevent the memory exhaustion attack, and finally protect the client from blocking the 4-way handshake.

Table 1. Comparison of performances

Type	Time(ms)	Clock (Mhz)	Time Overhead	Clock Overhead
4-way handshake	2.412	4.173	1	1
SNonce Response	2.491	4.213	1.032	1.009
PTK Cookie	2.484	4.204	1.029	1.007

5. Performance and Security Analysis

The proposed solutions have been implemented in a test bed constructed in the 802.11i environment. All machines are standard x86 processors using hostap[7] and xsupplicant[8]. We modified the source code of hostap and xsupplicant to implement our proposals. We then measured the execution times and clocks of the 4-way handshake on the client. Table 1 shows computational performance. We measured the performance overheads of the proposed solutions based on the original 4-way handshake.

As shown in Table 1, the client experiences an additional 0.079ms latency in SNonce Response and 0.072ms latency in PTK Cookie. These additional overheads are barely noticeable in employing the proposed solutions.

There seems to be a security weakness, like a replay attack, because we insert additional parameter into Message 2 in the 4-way handshake. That is, the client send SNonce in SNonce Response or E_{SPS}(PTK) in PTK Cookie to the AP and the AP send it to the client again. However, they can prevent the replay attack, because both the client and the AP store sn. If they received the message containing sn equal or less than the stored sn, they drop the message. Thus, our proposed solutions prevent the replay attacks as well.

6. Conclusion

802.11i is a well-designed standard, promising to improve the security of WLAN. However, security holes have been identified in the key derivation procedure in 802.11i. We have analyzed and studied the 4-way handshake protocol that is the key derivation procedure in 802.11i. The protocol is vulnerable in using unprotected messages.

In this paper, we showed the DoS attack abusing a vulnerability of the 802.11i 4-way handshake. We proposed two solutions to prevent the DoS attack. The proposed solutions do not allocate memory for Message 1, unlike 802.11i, because the client is able to get PTK after receiving Message 3.

We constructed a WLAN environment, and implemented our two proposals to measure execution times and clocks of the 802.11i 4-way handshake. As a result, we showed that: (a) the proposed solutions are more secure than the original 802.11i 4-way handshake, (b) performance differences between them are barely noticeable.

REFERENCES

[1] IEEE Std 802.11i/D4.1, "Wireless Medium Access Control(MAC) and Physical Layer(PHY) Specifications : Medium Access Control(MAC) Security Enhancements", July 2003

[2] Changhua He, John C. Mitchell. "Security analysis and improvements for IEEE 802.11i", *The 12th Annual Network and Distributed System Security Symposium*, February 2005

[3] Changhua He, John C. Mitchell, "Analysis of the 802.11i 4-Way Handshake", *In Proceedings of the Third ACM International Workshop on Wireless Security*, October 2004

[4] Hayriye Altunbasak, Henry Owen, "Alternative Pair-wise Key Exchange Protocols for Robust Security Networks (IEEE 802.11i) in Wireless LANs", *IEEE SoutheastCon* 2004

[5] Romano Fantacci, Leonardo Maccari, Tommaso Pecorella, " Analysis of Secure Handover for IEEE 802.1x-Based Wireless Ad Hoc Networks", *Wireless Communications*, Volume 14, Issue 5, Page(s):21 – 29, October 2007

[6] RFC 4987, "TCP SYN Flooding Attacks and Common Mitigations", August 2007

[7] http://hostap.epitest.fi

[8] http://open1x.sourceforge.net