*Sensors & Transducers, Vol. 183, Issue 12, December 2014, pp. 162-171*

**Sensors & Transducers**


*http://www.sensorsportal.com*

# An Approach for Prevention of MitM Attack Based on Rogue AP in Wireless Network

**Zhendong Wu, Mengru Cai, Siyu Liang, Jianwu Zhang**
College of Telecommunication Engineering, Hangzhou Dianzi University, Hangzhou 310018, China
E-mail: wzd@hdu.edu.cn

*Received: 11 October 2014 /Accepted: 28 November 2014 /Published: 31 December 2014*

**Abstract:** With the rapid development of WLAN, more and more schools and businesses have begun to provide the WLAN for users. However, WLAN is considerably more susceptible to MitM (man-in-the-middle) attack. To overcome it, we propose a dynamic password technology named Two-way Dynamic Authentication Technology (TDAT). It uses two-factor during the initial authentication, and uses a two-way hash chain during the cross-domain authentication. TDAT effectively protects users' authentication credentials and improves users' experience. In an actual wireless network environment, we implement a MitM attack framework based on Rogue AP. Then we effectively prevent this MitM attack by using TDAT. Moreover we analyze the security of TDAT by using BAN logic. *Copyright © 2014 IFSA Publishing, S. L.*

**Keywords:** Wireless network, MitM, Rogue AP, Dynamic password, Hash chain.

## 1. Introduction

With the rapid development of WLAN, more and more schools and businesses have begun to provide the WLAN for users. The security of WLAN is more important than it was in the past [1]. Open wireless network makes it particularly vulnerable to eavesdropping [2], DoS [3] and MitM [4] attacks. Therefore, a new standard-802.11i is proposed to improve the security of WLAN [5]. The 802.11i standard enhances WLAN security mainly in two aspects. In authentication aspect, port-based access control technology named 802.1X is used, EAP (Extensible Authentication Protocol), TLS encrypted tunnel and dynamic key management mechanisms are included. In encryption aspect, TKIP (Temporal Key Integrity Protocol), CCMP (CTR with CBC-MAC Protocol) is used. Although 802.11i is secure, in actual wireless network environment, correctly configuring the mobile terminal is not an easy mission for non-professional users. It will be more likely to suffer MitM (man-in-the-middle) attack.

In wireless network, the MitM attack based on Rogue AP is common attack. Attackers induce the user to connect the Rogue AP. Then they can monitor data or implement more advanced attacks to steal the user's information. The attack framework is shown in Fig. 1. In this attack framework, the most important thing is to get access to the wireless network. When the wireless network is using low-security authentication methods, such as open or WPA2-PSK model, attackers can get access to a wireless network and then launch a MitM attack using Rogue AP. Attackers forward the data between the Legitimate AP and the victims, meanwhile using sniffer software such as wireshark [6] to dump the plaintext data to filter out the plaintext username/password. This paper proposes an attack framework, which demonstrates that it is possible for the attackers to collect the authentication credentials by using the designed MitM attack based on Rogue AP. Then, we propose a dynamic password technology named Two-way Dynamic Authentication Technology (TDAT). It uses a two-factor authentication during the initial

authentication and a two-way hash chain authentication during the cross-domain authentication to prevent the MitM attack. In the initial authentication phase, we use the challenge, which is transmitted through the telecommunication channel (such as 3G or 4G), and the user's static password as the two factors to generate the dynamic password which is different every time. In the cross-domain authentication phase, both the client and the server maintain two hash chains as the authentication credentials. The authentication process proceeds automatically by the client in the background during cross-domain authentication. This scheme is convenient and secure to non-professional users. It could effectively protect users' authentication credentials and improve users' experience.

This paper continues as follows. In Section 2, we introduce the related works. In Section 3, we implement a MitM attack framework based on Rogue AP in actual wireless network environment. In Section 4, we propose Two-way Dynamic Authentication Technology (TDAT) to prevent the MitM attack. In Section 5, we analyze the security of TDAT by using BAN logic. In Section 6, we summarize the whole paper and put forward a direction for further research.

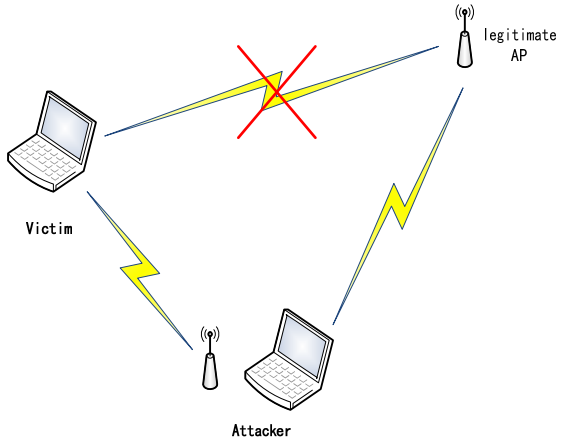


**Fig. 1.** MitM attack framework based on Rogue AP.

## 2. Related Works

### 2.1. 802.1X

The 802.1X standard is a port-based access control protocol, which uses a controlled/ uncontrolled port mode to control the transmission of the data packets [7]. Generally, The 802.1X's authentication mechanism, extensible standard for authenticating users, is implemented by the STA (Station, The wireless terminal) and the AS (Authentication Sever) using EAP data frames. The EAP data frames could pass the 802.1X uncontrolled port transmitting authentication-data. Then, non-EAP data frames would be passed or blocked via the controlled port depending upon the result of authentication.

### 2.2. Rogue AP

In this paper, Rogue AP means a wireless access point that has been created to allow an attacker to conduct a MitM attack. The Rogue AP does not employ mutual authentication and may be used in conjunction with a Rogue RADIUS server, depending on security configuration of the target network. The concept of Rogue AP is not new, but the installation of Rogue AP makes many attacks possible, such as MitM attack [8] and evil twin attack [9]. Rogue AP usually has the same SSID (Service Set Identifier) and configuration as the legitimate AP's. Moreover, Rogue AP must have stronger signals than the legal AP's that STA should connect.

### 2.3. BAN Logic

Burrows-Abadi-Needham logic, known as the BAN logic, is a set of rules for defining and analyzing information exchange protocols. There are three main stages of the analysis of a protocol using BAN logic. The first step is to express the assumptions and goals as formulas (also known as statements) in a symbolic notation, so that the logic can proceed from a known state so as to be able to ascertain whether the goals are in fact reached. The second stage is to transform the protocol steps also into formulas in symbolic notation. Lastly, a set of deduction rules called Postulates are applied. The postulates should lead from the assumptions, via intermediate formulas, to the authentication goals [10].

### 2.4. MitM Attacks and its Prevention in Wireless Network using 802.1X & EAP

Asokan et al. [11] point out that legacy client authentication protocol is not aware whether it is run in protected or unprotected mode. They also show that how to realize the MitM attacks in EAP-PEAP with EAP-AKA in a WLAN access authentication setting. Since 802.11i and 802.1X-2004 standard are not issued when this paper is published, the effect of this type of attack under 802.11i wireless environment is not yet known. Hyunuk Hwang et al. [12] propose the wireless MITM Framework, which can actively prove the vulnerability of MitM by applying the MitM technique in the WLAN, whose security is applied using 802.1X and EAP. This paper proposes that it is possible to collect the authentication information of the unauthorized user using the Wireless MitM-Framework under the EAP-MD5 environment. However, EAP-MD5 fails to meet any of the WLAN-required security claims and EAP-MD5 is rarely used in the actual wireless environment. Jing-Wei Zhou et al. [13] analyze the authentication process of EAP-PEAP in IEEE 802.1X authentication mechanism and propose a wireless

MitM-Framework under the EAP-PEAP environment. In this framework, the attacker forces the user to use a low-security access protocol EAP-MS-CHAPv2, in order to obtain the user's authentication credentials. However, under the actual wireless network environment, it is difficult to perform this kind of attack. In fact, the defaults EAP-methods in the mobile terminal are EAP-PEAP and EAP-TTLS. Thus, the attacker is unable to force the mobile terminal using EAP-MSCHAPv2. So such Wireless MitM-Framework under the EAP-PEAP environment is not valuable. Ding Wang et al. [14] analyze the MitM attack framework in 802.11i wireless network and the conditions of effective attacks then give an attack instance under the EAP-TTLSv0 environment. They conclude that whether EAP-TTLS and EAP-PEAP resist to the MitM attacks depends on the protocols implementation version. When EAP-PEAP and EAP-TTLS use a strong mutual authentication, they are not vulnerable to MitM attacks. However, the paper does not give examples of attack framework in actual wireless network environment, and the effect of this type of attack in actual network environment with mobile terminal is not yet known. In this paper, we implement a new MitM attack under the actual wireless network using EAP-TTLS/MS-CHAPv2.

Many methods were proposed for prevention of MitM Attacks in wireless network, such as 802.11i. But, most of them are inconvenient to use. It makes users less secure in actual networks. Some methods in other areas may enhance the security and give the convenience to non-professional users at the same time, such as Hash-chain. Lamport [15] suggests the use of hash chain as a password protection scheme in an insecure environment. A hash chain is the successive application of a cryptographic hash function to a piece of data. In computer security, a hash chain is a method to produce many one-time keys from a single key or password. Hash chain is often used in RFID authentication to achieve mutual authentication between Tag and Reader [16-18]. But very few hash chains are used in wireless network authenticaiton, considering the weak strength and low fault tolerance of Hash-chain crypto. In this paper, we propose a two-way hash chain protocol in the wireless network, which would use two hash functions and dynamic timestamps to enhance the strength of Hash-chain crypto and tolerate one-step fault in hash-chain mismatch.

## 3. MitM Attack under the EAP-TTLS Environment

### 3.1. EAP-TTLS/MSCHAPv2

EAP-TTLS [19] is an EAP method that encapsulates a TLS session. It provides a functionality beyond what is available in EAP-TLS, which improves the mutual authentication between the STA and the AS. EAP-TTLS extends this authentication negotiation in EAP-TLS by using the secure connection established by the TLS handshake to exchange additional information between STA and AS. EAP-TTLS protocol first established a TLS tunnel authentication. Once the TLS tunnel is completely established, the STA and AS use the legacy password-based authentication protocols to authenticate the STA. This exchange is fully encrypted by using the symmetric key. In this paper, we use the MS-CHAPv2 as the legacy password-based authentication protocol. Fig. 2 shows the process of the protocol. As we can see in Fig. 2, there is really only one unknown in the entire protocol – NTHash, which is used to construct three separate DES keys. Every other element of the protocol is either sent as plaintext, or easily derived from something sent as plaintext. Given that everything else is known, we can try to ignore everything but the core unknown, and see if there are any possibilities available to us. In this protocol we need three DES keys, each 7 bytes long, for a total of 21 bytes. Those keys are drawn from the NTHash which is only 16 bytes. The solution is to simply pad those last five bytes out as zero, effectively making the third DES key two bytes long. Since the third DES key is only 2 bytes long, we can get it in a matter of seconds. We are left trying to find the remaining 14 bytes of the NTHash, but can divide-and-conquer those in two 7 bytes chunks, for a total complexity of $2^{56}$.

### 3.2. Application of MitM Framework in EAP-TTLS/MS-CHAPv2

In EAP-TTLS and EAP-PEAP, STAs must properly validate the RADIUS server first. However, in actual wireless networks, the STAs fail to properly validate the server frequently. The main reasons are:

- Certificates issued by External Certificate Authority are very expensive. Businesses are reluctant to purchase certificates for a free wireless network.
- Certificate binding to the SSID is still a manual process on wireless networks. Some users directly set the configuration to "not validate server certificate", which stop the certification about the server.
- The default behavior of mobile terminals is to prompt users to validate the RADIUS server certificate. This is likely not ideal, since users typically have a hard time distinguishing what the certificate means and whether or not they should proceed.

In the Rogue-AP-based MitM attack framework, clients, especially mobile terminals, may establish a session with a Rogue AP and send its credentials along, which could be cracked. Fig. 3 describes the Rogue-AP-based MitM attack Framework. In business or school, users' wireless network authentication credentials are always binding up with

other systems, such as e-mail systems and internal authentication system. Therefore, the loss of users' authentication credentials will cause great security risk.

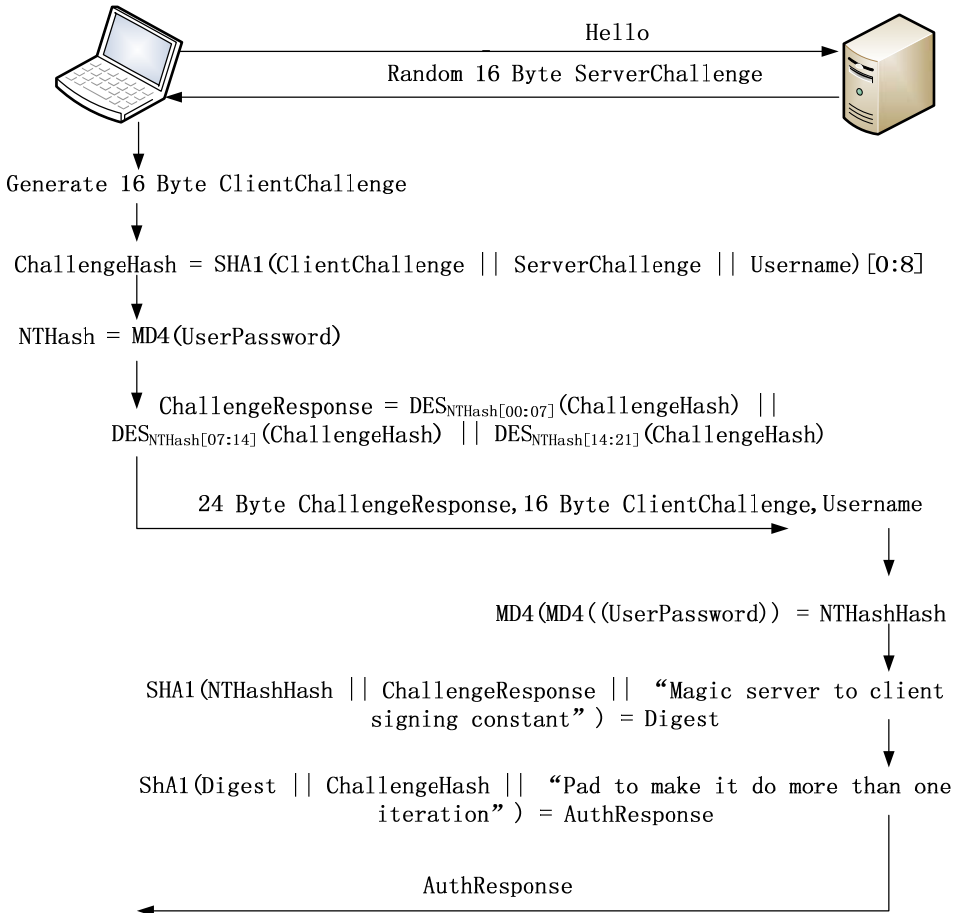


**Fig. 2.** The process of MS-CHAPv2.

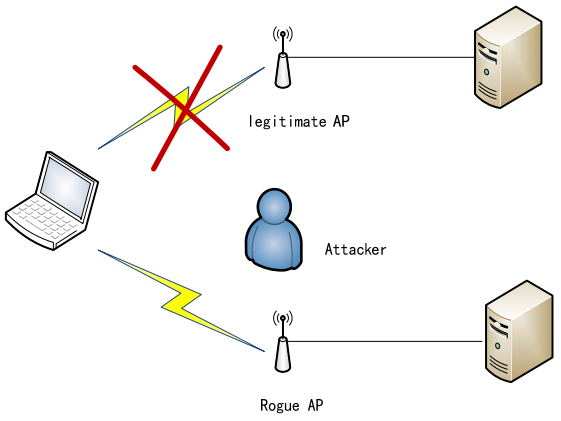


**Fig. 3.** MitM attack Framework based on Rogue AP in EAP-TTLS.

Specific procedures of the Rogue-AP-based MitM framework are as follows:

- The attacker sniffs the network traffic between the AP and the client, obtaining the appropriate information about the AP. Then the attacker configures the Rogue AP to make it perform like a legitimate AP.
- The attacker cuts the connection between client and legitimate AP.
- The client re-connects to the AP with the best signal strength. Therefore it can easily be "tricked" into connecting to the Rogue AP if the Rogue AP is strong enough.
- The client establishes a session with the Rogue AP and completes the authentication process. The attacker gets the MS-CHAPv2 packets and mounts a dictionary attack to crack the MS-CHAPv2.
- The attacker uses the username and the password to connect the legitimate AP.

In the Rogue-AP-based MitM attack Framework mentioned above, the attacker should meet the following conditions:

Condition 1. The attacker should be able to monitor the wireless network, and can cut the connection between STA and legitimate AP.

Condition 2. There should be a Radius server which can get the MS-CHAPv2 packets.

Condition 3. The attacker should be able to crack MS-CHAPv2.

The attacker can use the Aircrack suite [20] to meet Condition 1. Condition 2, attackers can use

Wireless Pwnage Edition [21] of FreeRadius [22] to complete the authentication process. Condition 3, attackers can use asleap [23], mschapv2acc [24] or John the Ripper [25] to crack MS-CHAPv2. Therefore, the attacker can meet all conditions and get users' authentication credentials through the tools mentioned above. Then we will test this MitM Framework in the actual wireless network environment.

### 3.3. MitM Attack Instance in the Actual Network

We are using IOS and Android mobile terminals in the actual network environment to simulate MitM attacks. We select iPhone5s and Mi3, where iPhone5s's username is "iphone", password is "ios"; Mi3's username is "android", password is "android". Legitimate AP is Cisco AIR-LAP1242AG-C-K9 AP, SSID set to "MitM". Legitimate Radius server is configured Windows2003. Rogue AP selects Motorola AP650, and Rogue Radius server is configured FreeRadius-WPE 2.1.12 in CentOS5.5. First, we configure the SSID of the Rogue AP to "MitM". FreeRadius-WPE uses EAP-TTLS, the internal authentication method is MS-CHAPv2. When the iPhone5s first connects the "MitM", the default behavior is to prompt the user to validate the certificate. This is likely not ideal, since users typically have a hard time distinguishing what a certificate means and whether or not they should proceed. Mi3 needs to set the profile when connected to the "MitM" as Fig. 4 shows. As we can see, there is no certificate to choose. Thus it fails to validate the server.

The attacker makes STA disconnect with the legitimate AP. Then the STA is "tricked" into connecting to the Rogue AP and completing the authentication process. After authentication is completed, the FreeRadius-WPE gets the legitimate username, challenge and response. The attacker uses asleap2.2 to obtain the password. Thus the attacker gets the legitimate user's username and password. iPhone5s and Mi3 username and password's crack are shown in Fig. 5 and Fig. 6.

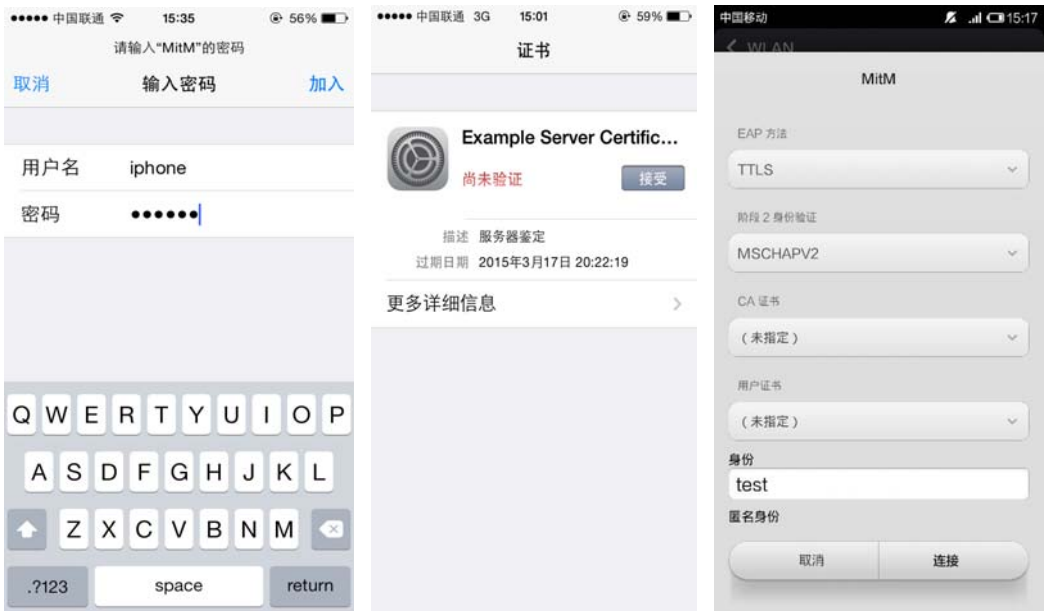


**Fig. 4.** IOS and Android Configuration.


**Fig. 5.** The crack of iphone.

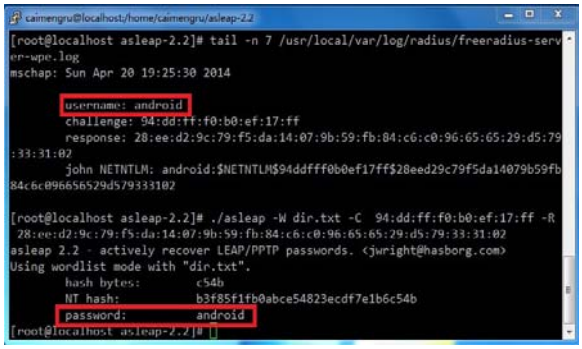


**Fig. 6.** The crack of android.

## 4. Two-way Dynamic Authentication Technology (TDAT)

### 4.1. Two-factor Initial Authentication

Since the mobile client is difficult to validate the server's certificate in a real environment, we propose a dynamic password technology based on two-factor in the second phase of EAP-TTLS to protect the user's authentication credentials. In this technique, we introduce a telecommunication channel, such as 3G or 4G, to enhance the security.

The two-factor include:

1) The unique identifier of the users, devices and the static password;

2) A randomly generated Challenge, which is received from the server via telecommunication channel.

Specific procedures are as follows:

- Users register their mobile phone number, IMEI number and static password when they use for the first time.
- When a user needs to connect to a wireless network, sending a request message containing the username to the server via the telecommunication channel.
- When the server receives a request message, first it would verify the user's mobile phone number and username. If the user is legitimate, the server sends a randomly generated challenge to the user, which is generated by Cryptographically Secure Pseudo-Random Number Generator, and has a length of 8 bytes. The challenge is only valid within a certain time window. In this paper, the time window is 60 s.
- A user enters a username, challenge and static password in the client software. The client software uses SHA256 hash algorithm to generate a dynamic password as the credential for this connection. The dynamic password is as follows:

***Dynamic Password*** = SHA256 (SHA256 (static password) + challenge + IMEI); where + denotes concatenation. The result of the SHA256 is 32-octet. The dynamic password's length is 64 bytes, which is the ASCII hexadecimal digits of the result of SHA256.

- After the EAP-TTLS authentication process has completed, the server receives the dynamic password of users' and uses the same algorithm to generate the dynamic password.

### 4.2. Two-way Cross-domain Authentication

In the cross-domain authentication, the client needs to initiate re-authentication. It is also vulnerable to Rogue-AP-based attack. In this paper we propose an authentication protocol based on a two-way hash chain to prevent the MitM attack and improve the ease-of-use during the cross-domain authentication. In this protocol the client and the authentication server both maintain a hash chain to implement mutual authentication. The authentication process proceeds automatically by the client in the background to improve user experience.

The protocol modifies the second phase of the EAP-TTLS authentication protocol. Authentication server and the client both have two hash functions H and G. Function H is used to update the hash chain nodes, while the function G is used to encrypt the authentication information between the server and the client. For the client, only when the authentication has completed, it updates the value of hash chain nodes. For the authentication server, when the client is authenticated, it updates the value of hash chain nodes.

We use the first32 bytes of Dynamic Password (S1, S2), which is generated during the initial authentication, as the client's initial hash chain value S0 = S1, and the last32 bytes are used as the authentication server's initial hash chain value T0 = S2. The hash chain of client is: $S_1 = H(S_1)$, $S_2 = H^2(S_1)$, ..., $Sj = H^j(S_1)$, ..., $SN = H^N(S_1)$; the hash chain of authentication server is: $T_1 = H(S_2)$, $T_2 = H^2(S_2)$, ..., $T_j = H^j(S_2)$, ..., $TN = HN(S_2)$.

After the first authentication, the client saves (Identity, $S_1$, $T_0$) and the authentication server saves (Identity, $S_0$, $T_1$). This protocol's authentication process is shown in Fig. 7. The i-th authentication process is as follows (Where S represents the authentication server, C represents the client. Si is the client's i-th hash chain node. Ti is the authentication server's i-th hash chain node):

1) S→C: *Transfer* (Random || timestamp1 || Identity Request). The authentication server sends a string of random numbers, a timestamp and an identity request to the client. An identity Request requests the ID of the client.

2) C→S: *Transfer* ($G(S_i$ || Random || timestamp1) || Identity Response || timestamp2). As soon as the client receives the packet. It first determines whether the timestamp is valid. If it is valid, the client calculates G (Si || Random || timestamp1) and sends an Identity Response, and a timestamp to the authentication server.

3) S→C: *Transfer* ($G(T_i$ || Random || timestamp2)). After the authentication server receiving the packet, it first determines whether the timestamp is valid. If the timestamp is valid, the authentication server calculates G ($H(S_j)$ || Random || timestamp1) or G ($S_j$ || Random || timestamp1) (the $S_j$ is the hash chain node of client which is saved by the authentication server and associated with the user identity) and compares G ($H(S_j)$ || Random || timestamp1) with what the value has received. If the values are equal, the authentication server calculates G ($T_i$ || Random || timestamp2) and sends it to client, authentication updates the hash chain node's value of client and authentication server.

4) C: As soon as the client receives the packet, it calculates G ($H(T_j)$ || Random || timestamp2) or G ($H^2(T_j)$ || Random || timestamp2) (the $T_j$ is the hash chain node of authentication server which is saved by

the client and is associated with the user identity) and compares G ($H(T_j)$ || Random || timestamp2) with the value, which is received from the authentication server. If the values are equal, the client updates the hash chain node's value of client and authentication server. The protocol based on a two-way hash chain is used in the second phase of EAP-TTLS, and the process is fully encrypted by using the symmetric key. At the same time, the data between the client and the authentication server are encrypted by the Hash function G. Thus an attacker cannot infer the plaintext according to the hash values. Therefore, even in the MitM attack based on Rogue AP mentioned above, the attacker can only get the hash values, and, cannot get the value of the client's hash chain node. After the authentication has completed, both the client and the server update the value of hash chain node. For the client, only when the authentication is completed, it updates the value of hash chain nodes. For the authentication server, when the client is authenticated, it updates the value of hash chain nodes. Obviously, authentication depends on the correct value of hash chain nodes, and the value of hash chain nodes is changed every time. Hence it is very difficult for attackers to get the right value every time.

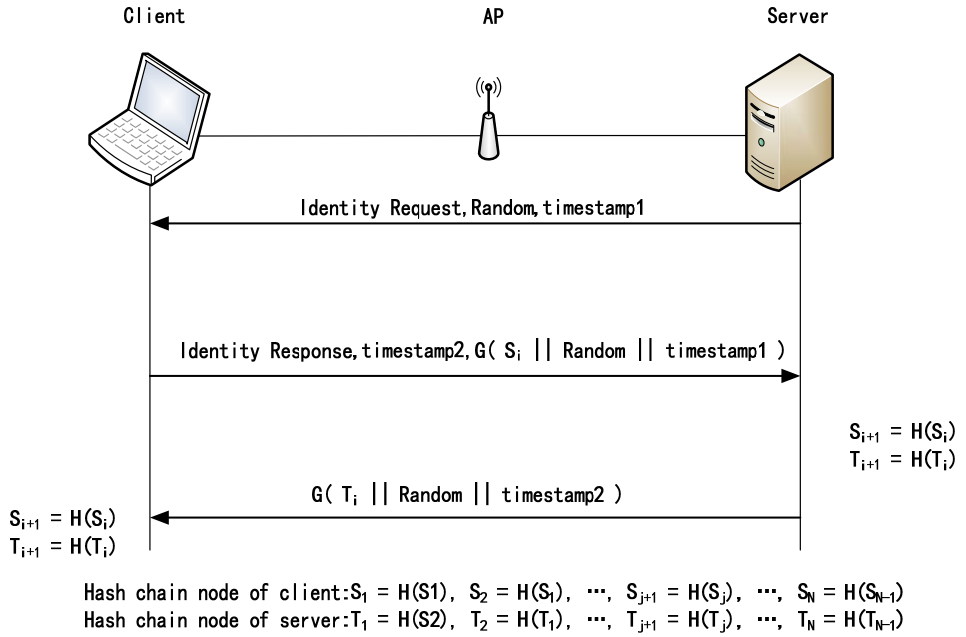


**Fig. 7.** The process of two-way hash chain based protocol.

## 5. Security Analysis

### 5.1. Security Analysis of Two-factor Dynamic Authentication

The security analysis based on two-factor can provide protection against MitM attack based on Rogue AP. The reasons are showed as follows:

- The user's credential is based on two-factor mechanism. In order to obtain the user's authentication credential, the attacker must simultaneously obtain these two factors. It is very difficult. The attacker must monitor the telecommunication channel to get the challenge, and know the user's username and static password at the same time. None of them is easy.
- The client software uses High-security dynamic password generation algorithm based on SHA256 hash algorithm, which is not reversible and has the complexity of 2^256. It is impossible to crack SHA256 through dictionary attacks and brute-force attacks.
- In the MitM attack based on Rogue AP discussed above, the attacker can only obtain the dynamic password through dictionary attack. The dynamic password is only valid within a certain time window (time window << the time to crack MS-CHAPv2).

Now we test this dynamic password technology based on two-factor in the actual wireless network with the iphone5s. The test environment is the same as the experiment mentioned above. The IMEI of the iphone5s is "358778050761484", the challenge received from the server via the 3G network is "a35Vrf9E". The result is shown in Fig. 7. As we can see in Fig. 8, the attack cannot crack the SHA256 to obtain the useful users' credentials. As the dynamic password is only valid within a certain time windows, the attacker cannot use this dynamic password to connect the legitimate AP successfully.

In our two-factor dynamic password technology, the challenge, which is transmitted via the 3G network, is important. In this paper, we use Cryptographically Secure Pseudo-Random Number

Generator to generate 6 bytes, and use base64 to convert 6 bytes to 8 bytes. We connect the AP 7 times to show the Pseudo-Random numbers. The result is shown in Table 1. As we can see in the Table 1, as the challenge changed, the dynamic password is totally different.

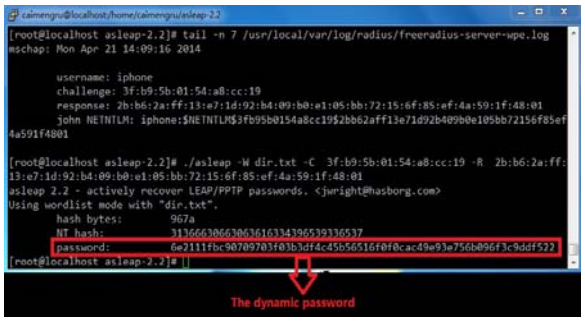


**Fig. 8.** The test result of Two-factor-based dynamic password technology.

**Table 1.** The correspondence table of challenge and dynamic password.

| Challenge | Dynamic Password |
|---|---|
| a35Vrf9E | 6e2111fbc90709703f03b3df4c45b56516f0f0cac49e93e756b096f3c9ddf522 |
| 4rF7Q9na | a04155fd6f40efba2c6b7ddcb2f7fe873d3d061e55c955f55fd490043608d662 |
| Ec68D9m3 | 095d7c63f8ad8b1b8dd9ef3e1f8ce6628dd80aa7977747524ed7560dbf260884 |
| 37dfN89a | 41df1f0f7aa1a73d3e586df7e708a9a1843115b734c6c6f5222345fd562b0e83 |
| Qodf7Cdp | 39f62d0f2916367cb2ee7d65ad6aa777e0f7a2a4f44e56dc8908899b7b35da60 |
| 58asdwql | a0af0632f00b6e671debc09a7ea471dbfbb20e195f93c89281614545ac4217b6 |
| Sdwe45gf | ee278f25c3f35fdfc705770da032a80d13ac36898b610c27ff9f96028bd6bcd6 |

## 5.2. Security Analysis of Two-way Hash Chain

The two-way hash-chain-based authentication protocol ensures the mutual authentication between the client and the authentication server. Now we use the BAN logic to analysis this protocol. The BAN logic has several modal operators and numerous rules of inference for manipulating the protocol to generate a set of beliefs including:

1) P believes X. P is entitled to act as though X is true.

2) P sees X. Someone has sent a message to P containing X so that he can read X and repeat it.

3) P said X. At some time, P uttered a message containing X.

4) P controls X. P is an authority on X and can be trusted on X.

5) *Fresh*(X). It means that X has not been sent before in any run of the protocol.

6) $\{X\}_K$. $X$ is encrypted with key $K$.

7) $P \xleftrightarrow{K} Q$. K is a secret known only to P, Q and possibly some trusted associates.

8) R1

$$\frac{P \ believes \ Q \xleftrightarrow{K} P, P \ sees \ \{X\}_K}{P \ believes \ Q \ said \ X}$$

If P believes key (K, P↔Q), and P sees $\{X\}_K$, then P believes (Q said X).

9) R4

$$\frac{P \ believes \ fresh(X), P \ believes \ Q \ said \ X}{P \ believes \ Q \ believes \ X}$$

If P believes (Q said X) and P believes fresh(X), then P believes (Q believes X).

10) R5

$$\frac{P \, believes \ Q \, controls \, X, P \, believes \ Q \, believes \, X}{P \, believes \ X}$$

If P believes (Q has jurisdiction over X) and P believes (Q believes X), then P believes X.

11)R11

$$\frac{P \ believes \ fresh(X)}{P \ believes \ fresh(X,Y)}$$

If P believes that X is fresh, then P believes that (X,Y) is fresh.

Now we formalize the protocol, where S represents the authentication server, C on behalf of the client, the combination of T represents a random number and timestamp, $ID_C$ represents the hash chain node of client and $ID_S$ represents the hash chain node of the authentication server.

1) S→C: S sends an identity request and T to C.

2) C→S: C sends an identity response and $\{T, ID_C\}_{Ki}$ to S. $K_i$ is the share key between C and S.

3) S: As soon as S receives $\{T, ID_C\}_{Ki}$, it verifies the $ID_C$. If the $ID_C$ is legal, continue to step 4, otherwise the authentication fails.

4) S→C: S sends $\{T, ID_C\}_{Ki}$ to S. $K_i$ is the share key between C and S.

5) C: As soon as S receives $\{T, ID_S\}_{Ki}$, it verifies the $ID_S$. If the $ID_S$ is legal, the authentication is success.

We reserve the relevant part of the protocol security analysis: S sees $\{T, ID_C\}_{Ki}$, C sees $\{T, ID_S\}_{Ki}$. Our safety goal is: S believes $ID_C$, C believes $ID_S$.

Then, we make the following initial assumptions on the protocol:

1) P1 $S \ believes \ S \xleftrightarrow{Ki} C$

2) P2 $C \ believes \ C \xleftrightarrow{Ki} S$

3) P3 $S \ believes \ fresh(T)$

4) P4 $C \ believes \ fresh(T)$

5) P5 $S$ *believes* $C$ *controls* $ID_C$

6) P6 $C$ *believes* $S$ *controls* $ID_S$

Finally, we use analogical reasoning. When S sees $(T, ID_C)_{Ki}$, according to the condition P1 and the rule R1:

$$\frac{P \ believes \ \ Q \xleftrightarrow{K} P, P \ \ sees \ \ \{X\}_K}{P \ \ believes \ \ \ Q \ \ said \ \ \ X}$$

We can know S *believes* C *said* $\{T, ID_C\}$ which also means S *believes* C *said* $ID_A$.

According to the condition P3 and the rule R11:

$$\frac{P \ \ believes \ \ \ \ fresh\,(X)}{P \ \ believes \ \ \ \ fresh\,(X, Y)}$$

We can know S *believes fresh*$(T, ID_C)$ which also means S *believes fresh*$(ID_C)$.

According to the condition P3 and the rule R4:

$$\frac{P \ \ believes \ \ \ fresh\,(X), P \ \ believes \ \ Q \ \ said \ \ X}{P \ \ believes \ \ Q \ \ believes \ \ X}$$

We can know S *believes* C *believes* $ID_C$.

According to the condition P5 and the rule R5:

$$\frac{P \ believes Q \ controls \ X, P \ believes Q \ believes \ X}{P \ believes \ X}$$

We can know S *believes* $ID_C$. In the same way, we can know C *believes* $ID_S$.

According to the analysis above, we know that the protocol fully meets the security objectives S believes $ID_C$, C believes $ID_S$. Therefore, the protocol can meet our demand.

## 6. Conclusions

In this paper, we introduce the MitM framework based on Rogue AP to the wireless environment with EAP-TTLS authentication. We also propose a dynamic password technology based on two-factor and a protocol based on a two-way hash chain to resist to this MitM attack. With the growing demand for Internet and mobile office, we should see WLAN to play a more significant role in the game in a near future. The security of WLAN will become a hot research direction. More advanced network security technologies should appear to meet the demand.

## Acknowledgements

This research is supported by Zhejiang province science and technology project No. 2013C33056.

## References

[1]. C. Yang, G. Gu, Security in wireless local area networks, Wireless Network Security: Theories and Applications, *Springer*, Berlin Heidelberg, 2013, pp. 39-58.

[2]. G. Chen, Y. Zhang, C. Wang, A wireless multi-step attack pattern recognition method for WLAN, *Expert Systems with Applications*, Vol. 41, Issue 16, 2014, pp. 7068-7076.

[3]. J. G. Tang, Wireless network risk assessment model and application, *TELKOMNIKA Indonesian Journal of Electrical Engineering*, Vol. 12, No. 6, 2014, pp. 4639-4647.

[4]. W. L. Chen, Q. Wu, A proof of MITM vulnerability in public WLANs guarded by captive portal, *Proceedings of the Asia-Pacific Advanced Network*, Vol. 30, 2010, pp. 66-70.

[5]. S. Frankel, B. Eydt, L. Owens, et al, Establishing wireless robust security networks: a guide to IEEE 802.11i, *National Institute of Standards and Technology (NIST)*, 2007.

[6]. Wireshark, 2014, (http://www.wireshark.org/).

[7]. H. Zhi, X. Q. Wang, W. Chen, Application of IEEE802. 1x protocol based on EPON system, *Applied Mechanics and Materials*, Vol. 336-338, 2013, pp. 2433-2437.

[8]. I. Kim, J. Seo, T. Shon, et al, A novel approach to detection of mobile rogue access points, *Security and Communication Networks*, Vol. 7, Issue 10, 2013, pp. 1510-1516.

[9]. S. Nikbakhsh, A. B. A. Manaf, M. Zamani, et al, A novel approach for rogue access point detection on the client-side, in *Proceedings of the 26th IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, Fukuoka, Japan, 26-29 March 2012, pp. 684-687.

[10]. D. M. Nessett, A critique of the Burrows, Abadi and Needham logic, *ACM SIGOPS Operating Systems Review*, Vol. 24, Issue 2, 1990, pp. 35-38.

[11]. N. Asokan, V. Niemi, K. Nyberg, Man-in-the-middle in tunnelled authentication protocols. Security Protocols, *Lecture Notes in Computer Science*, Vol. 3364, Springer, Berlin, 2005, pp. 28-48.

[12]. H. Hwang, G. Jung, K. Sohn, et al, A study on MITM (Man in the Middle) vulnerability in wireless network using 802.1 X and EAP, in *Proceedings of the International Conference on Information Science and Security (ICISS'08)*, Seoul, Korea, 10-12 January 2008, pp. 164-170.

[13]. Jing-Wei Zhou, Sheng-Ju Sang, Analysis and improvements of PEAP protocol in WLAN, in *Proceedings of the International Symposium on Information Technology in Medicine and Education*, Hokkaido, Japan, 3-5 August 2012, Vol. 2, pp. 918-922.

[14]. Wang Ding, Ma Chun-Guang, Weng Chen, Jia Chun-Fu, Research of man-in-the middle attack in robust security network, *Journal of Computer Applications*, Vol. 32, Issue 1, 2012, pp. 42-44, 65.

[15]. L. Lamport, Password authentication with insecure communication, *Communications of the ACM*, Vol. 24, Issue 11, 1981, pp. 770-772.

[16]. J. S. Yuan, Y. Hu, Implementation of RFID middleware based on hash chain, *Applied Mechanics and Materials*, Vol. 411, 2013, pp. 12-15.

[17]. C. Lin, S. F. Zhao, S. P. Chen, RFID secure protocol based on time-based hash chain, *Advanced Materials Research*, Vol. 980, 2014, pp. 225-229.

[18]. I. Syamsuddin, T. Dillon, E. Chang, et al, A survey of RFID authentication protocols based on hash-chain method, in *Proceedings of the Third International Conference on Convergence and Hybrid Information*

*Technology (ICCIT'08)*, Busan, Korea, 11-13 November 2008, Vol. 2, pp. 559-564.
[19]. P. Funk, S. Blake-Wilson, Extensible authentication protocol tunneled transport layer security authenticated protocol version 0 (EAP-TTLSv0), *Network Working Group, RFC 5281*, 2008.
[20]. Aircrack, Aircrack-ng, 2012, (http://www.aircrack-ng.org/).
[21]. J. Wright, FreeRADIUS, Wireless Pwnage Edition, 2010, (http://www.willhackforsushi.com/).
[22]. FreeRADIUS Project, FreeRADIUS Radius Server, 2012, (http://freeradius.org/).
[23]. J. Wright, ASLEAP, Exploiting CISCO LEAP, 2008, (http://www.willhackforsushi.com/Asleap.html).
[24]. B. Charles, mschapv2acc, a proof of concept of MS-CHAPv2 auditing and cracking tool, 2010, (http://code.Google.com/p/mschapv2acc/).
[25]. Openwall Project, John the Ripper, 2012, (http://www.openwall.com/john/).