Security Analysis of WAPI Authentication and Key Exchange Protocol *

Liufei Wu[1]，[2] Yuqing Zhang[2] FengjiaoWang[2]

[1]Communication Engineering Institute，Xidian University，Xi'an 710071

[2]National Computer Network Intrusion Protection Center,GUCAS, Beijing100049

Abstract

We first do an in-depth security analysis of the authenticated key exchange protocol WAI in WAPI (WALN Authentication Privacy Infrastructure), point out its flaws and improve it. Next, we give the security proof of this new protocol WAI' in CK security model, which indicates that WAI' has the corresponding security attributes in CK security model, and satisfies the requirements of WAPI.

Key words: *WAI protocol; CK security model; Security analysis*

1. Introduction

In order to adapt to the rapid development of wireless network communication, China has developed the national standard of the wireless LAN standard (GB 15629.11-2003) WAPI [1]. WAPI consists of Wireless Authentication Infrastructure WAI) and Wireless Privacy Infrastructure (WPI). WAI is responsible for authentication and key management, it is the main study. As the first wireless LAN national standard that develops by China herself, it will have enormous and far-reaching influence to the development and strengthen of the information industry of China. Therefore, whether WAPI standard could meet the security goals in wireless communication, such as the identity authentication, integrality of the data and confidentiality, seems particularly important. To the authentication model in GB 15629.11-2003, literature [2] utilized CK security model to analyze it and pointed out its existing flaws, and then a wireless authentication protocol to solve security problems existing in WAPI authentication and key exchange part was given as an enhancement. Literature [3] used BAN logic to analyze and prove the WAPI authentication process in GB 15629.11-2003 and GB 15629.1102-2003. Although the two results are different, both of them have important theoretical significance.

This paper analyzes the WAPI access and authentication process in GB /XG 15629.11-20031-2006, pointes out its flaws and improves it correspondingly. Analysis indicates that the protocol WAI' can offer the formal proof under CK security model.

2. WAPI Access and Authentication Process

2.1. WAPI access and authentication process [1]

WAPI access and authentication process consists of three sub-modules: certificate authentication process, unicast key agreement process and multicast / station key notification process, shown as Figure 1. When the Authentication Supplicant Entity (ASUE) related or related to the Authentication Entity (AE), AE and ASUE need mutual certificate authentication. Only after authentication succeeds, AE allows ASUE to access, at the same time ASUE allows to receive and dispatch the data through this AE. Authentication Service Entity (ASE) is responsible for certificate authentication of AE and ASUE.

* This work is supported by National Natural Science Foundation of China (grant Nos. 60573048, 60373040).

[1]Corresponding author: Tel: +86-010-68860988, Fax: +86-010-68860988.

E-mail address: wlf0701@hotmail.com

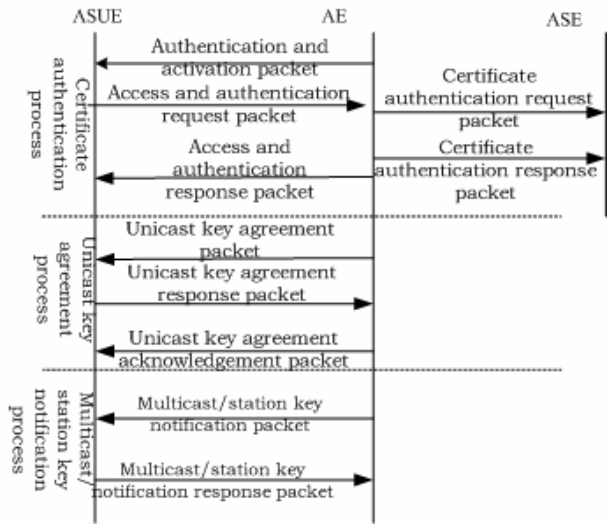

Figure1 WAPI access and authentication process

The concrete process of access and authentication is as follows:

AE sends an authentication and activation packet to start the entire authentication process. On receiving the authentication and activation packet that AE sends, ASUE checks and distinguishes each word section first. If the requirements are met, ASUE produces an access and authentication request and sends it back to AE. After that, AE sends certificate authentication packet to ASE, and puts the response of the certificate authentication received from ASE to an authentication response packet and sends it to ASUE. ASUE checks the state of the authentication response packet and the result of AE's certificate authentication, then determines whether to access this AE.

After the certificate is successfully authenticated, AE sends unicast key agreement packet and begins to unicast key process with ASUE. On receiving the unicast key agreement packet ASUE checks the present state and calculates the local unicast session key, and then ASUE constructs unicast key agreement response packet to AE. After the successful execution of unicast key agreement, AE sends multicast / station key packet to begin multicast / station key process.

Multicast / station key process makes use of the unicast session key for encryption. It uses the key transmission mechanism, hence its key security depends on unicast session key quality, which is out of the scope of this paper. Our main research of WAPI access authentication is certificate authentication and unicat key process. First of all, we formalize WAPI access authentication process.

2.2. Formal Description of WAI Protocol

WAI protocol is authentication and key exchange protocol based on digital certificate exchange, as Figure 2 shows. We only keep the correlated message word section while formalizing, and the content which is omited will not influence the function of the protocol. It is especially pointed out that ASE does not participate in the operation of the protocol except that it distinguishes the certificates of ASUE and AE. Therefore, we omit the ASE certificate authentication process while analyzing.

Premise : AE and ASUE have one's own public key certificates $Cert_{AE}$ and $Cert_{ASUE}$ respectively, P_{AE} and P_{ASUE} denote the identities of AE and ASUE. Assuming that all parties have globally agreed upon a nonsingular high elliptic curve, $E\left(F_q\right)$, where $P \in E\left(F_q\right)$, and n is its order, such as would be defined by an appropriate standards body. Where s is session identifier, R_{AE}, R_{ASUE} denote the challenges of AE and ASUE respectively, N_{AE}, N_{ASUE} is random number for

AE and ASUE to calculate session keys, $ADDID = MAC_{AE} \| MAC_{ASUE}$

Goal: AE and ASUE realize mutual authentication and key exchange.

1. AE $\rightarrow$ ASUE: $s, Cert_{AE}$

2. ASUE $\rightarrow$ AE: $s, R_{ASUE}, \alpha = x \bullet P, P_{AE}, Cert_{ASUE}, SIG_{ASUE}\left(s, R_{ASUE}, \alpha, P_{AE}, Cert_{ASUE}\right)$, where $x \leftarrow_R [1, n-1]$

3. AE $\rightarrow$ ASUE: $R_{ASUE}, R_{AE}, \beta = y \bullet P, \alpha, P_{AE}, P_{ASUE}, SIG_{AE}\left(R_{ASUE}, R_{AE}, \beta, \alpha, P_{AE}, P_{ASUE}\right)$, where $y \leftarrow_R [1, n-1]$;

4. AE $\rightarrow$ ASUE: $ADDID, N_{AE}$

5. ASUE $\rightarrow$ AE: $ADDID, N_{ASUE}, N_{AE}, HMAC_{BK}(ADDID, N_{ASUE}, N_{AE})$

6. AE $\rightarrow$ ASUE: $ADDID, N_{ASUE}, HMAC_{BK}(ADDID, N_{ASUE})$

Figure 2 Protocol WAI

The initial three-step of the protocol is for certificate authenticating, AE and ASUE exchange through ECDH and utilize key export function f to calculate the base key $BK = f\left(x \bullet y \bullet P, R_{AE} \| R_{ASUE}\right)$. In the pre-shared key mode, AE and ASUE lead out the base key by regarding the shared keys as the seed. It should be explained that the key seed $x \bullet y \bullet P$ of ECDH algorithm can not be infinite points. The following 3 steps conduct unicast key agreement process. AE exchange a random number with ASUE respectively, and lead out the unicast session key $K = f\left(BK, ADDID \| N_{ASUE} \| N_{AE}\right)$ by utilizing function f and base key BK.

2.3. Security Analysis of WAI Protocol

The goal of WAI protocol is to realize mutual authentication and key exchange, but the flaws exist in the protocol make the goal cannot be realized. To facilitate the description, we discuss the certificate authentication process and unicast key agreement process of WAI respectively.

2.3.1. Security Flaws in WAI Certificate Authentication Process

WAI certificate authentication process is intended to achieve mutual authentication, but the flaws existed make this protocol insecure against passive opponents, see Figure 3.

1. AE $\rightarrow$ ASUE: $s, Cert_{AE}$

2. ASUE $\rightarrow$ AE: $s, R_{ASUE}, \alpha = x \bullet P, P_{AE}, Cert_{ASUE}, SIG_{ASUE}\left(s, R_{ASUE}, \alpha, P_{AE}, Cert_{ASUE}\right)$

1'. AE $\rightarrow$ $\mathcal{E}$ (ASUE): $s, Cert_{AE}$

2'. $\mathcal{E}$ (ASUE) $\rightarrow$ AE: $s, R_{ASUE}, \alpha, P_{AE}, Cert_{ASUE}, SIG_{ASUE}\left(s, R_{ASUE}, \alpha, P_{AE}, Cert_{ASUE}\right)$

Figure 3 Attack on WAI certificate authentication process

Since the authentication identifier s used by AE and ASUE while updating the base key does not change all the time, in this way, opponents can intercept and capture the access request packet of certificate authentication process and send to AE again. Then AE Will produce a random number y after it receives this message, even they can still calculate different main key seed $x \bullet y \bullet P$ with multiplying α like this, so AE can not perceive this attack here. Due to the freshness of the public key is not assured, the protocol is liable to message replaying attack. The consequence is that AE thinks it has updated the base key with ASUE, in fact ASUE has not

participated in this protocol. Then the base key fails to update.

2.3.2. Security Flaws in WAI Unicast Key Process

We present an attack about the unicast key process of WAI, which is similar to the reflecting attack given in literature [4], see Figure 4. Note that in the base service set mode, unicast key process is initiated by AE only. However, in the independent base service set (IBSS) mode, two STAs should shake hands as AE and ASUE separately. Thus, the premise that we point out the attack is rational.

Premise: opponents $\mathcal{E}$ of the operation launched by the STA play ASUE protocol, STA received the first of this news, but also to play AE launched operation of the protocol.

4. STA $\rightarrow \mathcal{E}$ (ASUE): $ADDID, N_{STA}$

4'. $\mathcal{E}$ (AE) $\rightarrow$ STA: $ADDID, N_{STA}$

5'. STA $\rightarrow \mathcal{E}$ (AE): $ADDID, N_{STA}^{'}, N_{STA}, HMAC_{BK}(ADDID, N_{STA}^{'}, N_{STA})$

5. $\mathcal{E}$ (ASUE) $\rightarrow$ STA: $ADDID, N_{STA}^{'}, N_{STA}, HMAC_{BK}(ADDID, N_{STA}^{'}, N_{STA})$

6. STA $\rightarrow \mathcal{E}$ (ASUE): $ADDID, N_{STA}^{'}, HMAC_{BK}(ADDID, N_{STA}^{'})$

Consequences: SAT thinks ASUE is true in the operation of the current protocol, and in fact, ASUE has not participated in that operation at all.

Figure4 Attacks on unicast agreement process of WAI protocol

Opponents $\mathcal{E}$ has carried out the reflecting attack twice: message 4' is the reflection of message 4, news 5 is the reflection of message 5'. After honest subject STA receiving the message 5 and 5', they cannot detect any improper. First, the random packet that STA receives in message 4' is the random number that STA produce in message 4. Then STA only checks whether it is the first time that receives the random number N_{STA} or not, so the random number N_{STA} will pass freshness inspection smoothly. Similarly, the message that STA receives in message 5 is the message that it produce and sent out in message 5', and the random number $N_{STA}^{'}$ can pass successfully. Therefore, STA can not be aware of the attack.

The basic reason that the attack exists is the misapplication of the cryptography service of WAI protocol, which enable the adversary to get the oracle service by utilizing the legal participant.

3. Improvement of WAI Protocol

Aiming at the above mentioned security defects exist in WAI, protocol WAI' is given by doing some improvements to WAI.

Premise, goal: With WAI protocol.

1. AE $\rightarrow$ ASUE: $s, Cert_{AE}, \alpha = x \bullet P$

2. ASUE $\rightarrow$ AE: $s, R_{ASUE}, \beta = y \bullet P, Cert_{ASUE}, SIG_{ASUE}\left(Cert_{ASUE}, s, R_{ASUE}, \beta, \alpha, P_{AE}\right)$

3. AE $\rightarrow$ ASUE: $s, R_{ASUE}, R_{AE}, P_{AE}, SIG_{AE}\left(P_{AE}, s, R_{ASUE}, R_{AE}, \alpha, \beta, P_{ASUE}\right)$

4. AE $\rightarrow$ ASUE: s, MAC_{AE}, N_{AE}

5. ASUE $\rightarrow$ AE: $s, N_{ASUE}, MAC_{ASUE}, HMAC_{BK}(MAC_{AE}, N_{AE}, s, N_{ASUE})$

6. AE $\rightarrow$ ASUE: $s, MAC_{AE}, HMAC_{BK}(MAC_{ASUE}, N_{ASUE}, s, N_{AE})$

Fig 5 Protocol WAI'

The implementation of the protocol is similar to WAI. Note that the basic design philosophy of WAI' protocol is the same as WAI. The difference only lies in: every message joins the identifier s, which guarantees the freshness of the cryptography operation. WAI' produced $\alpha = x \bullet P$ in message first in the certificate authentication. And WAI' changes *ADDID* into one's own *MAC* addresses in unicast key agreement process. These small changes seem to be insignificant, which can produce the unexpected result. The improved protocol can resist the attack that we provided and have other relevant security attributes. The concrete analysis is as follows.

4. Security Proof of Protocol WAI'

In this section we will prove that protocol WAI' is secure under CK model. For the sake of clarity, we respectively record the certificate authentication as WAI'1 and unicast key agreement process as WAI'2. We will refer a brief introduce to CK model before proving.

4.1. CK Security Model

CK security model [8] presents definition of SK-security, allows for modular design and analysis of key exchange protocol, which simplifies the difficulty of design and analysis of security protocol. The security definition is based on the concept of indistinguishability

The attacker model follows the unauthenticated-links model (UM) that the attacker is a (probabilistic) polynomial-time machine with full control of the communication lines between parties. In addition, the attacker can have access to secret information via session exposure attacks of three types: session-state reveal, session-key queries, and party corruption. The first type of attack is directed at a single session which is incomplete and the result is that the attacker learns the session state of that particular session. A session-key query can be performed against an individual session after completion and the result is that the attacker learns the corresponding session-key. Finally, party corruption means that the attacker learns all information in the memory of that party; in addition, from the moment a party is corrupted all its actions are totally controlled by the attacker.

Sessions can be expired in the model of CK. From the time a session is expired the attacker is not allowed to perform a session-key query or a state-reveal attack against the session, but is allowed to corrupt the party that holds the session. Protocols that ensure that expired sessions are protected even in case of party corruption are said to enjoy "perfect forward secrecy".

For defining the security of a KE protocol, CK follows the indistinguishability style of definitions that the "success" of an attacker is measured via its ability to distinguish the real values of session keys from independent random values. When the attacker chooses the test session it is provided with a value υ which is chosen as follows: a random bit b is tossed, if $b = 0$ then υ is the real value of the output session-key, otherwise υ is a random value chosen under the same distribution of session-keys produced by the protocol, but independent of the value of the real session key. After receiving υ, the attacker may proceed with the regular actions against the protocol; at the end of its run the attacker outputs a bit b'. The attacker succeeds in its attack if (1) the test session is not exposed, and (2) the probability that $b = b'$ is significantly larger than $1/2$. Note that the attacker is allowed to corrupt a party to the test session once the test expires at that party (this captures perfect forward secrecy).

An adversarial model called authenticated-links model (AM) is defined in a way that is identical to the UM with one fundamental difference: the attacker is restricted to only deliver messages truly generated by the parties without any change or addition to them. Then the notion of "emulation" is introduced in order to capture the equivalence of functionality between protocols in different adversarial models, in particular between the UM and AM.

The resultant security notion for KE protocols is called SK-security and is stated as follows:

Definition 1. (SK-security) An attacker with the above capabilities is called an SK-attacker. A key-exchange protocol π is called SK-secure if for all SK-attacker $\mathcal{A}$ running against π it holds:

1. If two uncorrupted parties complete matching sessions in a run of protocol π under

attacker $\mathcal{A}$ then, except for a negligible probability, the session key output in these sessions is the same.

2. $\mathcal{A}$ succeeds in its test-session distinguishing attack with probability not more than $1/2$ plus a negligible fraction.

Definition 2. (SK-security without PFS) We say that a KE protocol without PFS if it enjoys SK-security relative to any KE-adversary in the UM that is not allowed to expire keys. (Similarly, if the above holds for any such adversaries in the AM them we say that π is SK-secure without PFS in the AM.)

Theorem 1. Let π be a SK-secure key-exchange protocol in the AM with PFS (resp., without PFS) and let λ be an MT-authenticator. Then $\pi' = C_\lambda(\pi)$ is a SK-secure key-exchange protocol in the UM with PFS (resp., without PFS).

4.2. Security Proof of WAI'1 Protocol

We first demonstrate that under the Elliptic Curve Decisional Diffie-Hellman (ECDDH) assumption the classic two-move Elliptic Curve Diffie-Hellman key-exchange protocol designed to work against an eavesdropper only is SK-secure in the AM. We denote this protocol by ECDH and describe it in Figure 6. Using Theorem 1 we can apply an appropriate authenticator to this protocol to obtain a secure Elliptic Curve Diffie-Hellman exchange against realistic UM attackers.

Premise：prime $P \in E(F_q)$ of order n 。

goal：AE and ASUE share a session key: $bk = x \bullet y \bullet P$

1. AE $\rightarrow$ ASUE： $s, P_{AE}, \alpha = x \bullet P$

2. ASUE $\rightarrow$ AE： $s, P_{ASUE}, \beta = y \bullet P$

Figure 6 ECDH protocol

The Elliptic Curve Decisional Diffie-Hellman (ECDDH) assumption is as follows:

ECDDH Assumption Let E be a nonsingular high Elliptic Curve on finite field F_q, Let $P \in E(F_q)$ be of order n, $x, y, z \in_R [1, n-1]$. Then the probability $\mathcal{D}$ distributes quintuples $Q_0 = \langle P, x \bullet P, y \bullet P, x \bullet y \bullet P \rangle$ and $Q_1 = \langle P, x \bullet P, y \bullet P, z \bullet P \rangle$ is computationally indistinguishable.

Theorem 2 If ECDDH assumption holds, protocol ECDH is SK-secure in the AM.

Proof: To see that the first requirement of Definition 1 is satisfied, note that if both parties are uncorrupted during the exchange of the key and both complete the protocol then they both establish the same key (which is $bk = bk' = x \bullet y \bullet P$). Note that the session identifier s uniquely binds the values of α and β to these particular matching sessions and differentiates them from other exponentials that the parties may exchange in other sessions.

We show that the second requirement of Definition 1 is also satisfied by protocol ECDH. Assume to the contrary that there is a KE-adversary $\mathcal{A}$ in the AM against protocol ECDH that has a non-negligible advantage ε in guessing correctly whether the response to a test-query is real or random. Out of this attacker $\mathcal{A}$, we construct an algorithm $\mathcal{D}$ that distinguishes between the distributions Q_0 and Q_1 with non-negligible probability, thus reaching a contradiction with Assumption 1. Algorithm $\mathcal{D}$ uses adversary $\mathcal{A}$ as a subroutine and is described in Figure 7.

Proceed as follows, on input $\langle q, P, \alpha^*, \beta^*, \gamma^* \rangle$:

1. Choose $r \leftarrow_R \{1 \ldots l\}$.
2. .Invoke $\mathcal{A}$ on a simulated interaction in the AM with parties running ECDH. Hand $\mathcal{A}$ the values, q , P as the public parameters for the protocol execution.
3. Whenever $\mathcal{A}$ activates a party to establish a new session (except for the r-th session) or to receive a

message, follow the instructions of ECDH on behalf of that party. When a session is expired at a player erase the corresponding session key from that player's memory. When a party is corrupted or a session (other than the r-th session) is exposed, hand $\mathcal{A}$ all the information corresponding to that party or session as in a real interaction.

4. When the r-th session, say (P_S, P_C, s), is invoked with P_S to exchange a key with P_C, let P_S send the message (P_S, s, α^*) to P_C.
5. When P_C is invoked to receive (P_S, s, α^*), let P_C send the message (P_C, s, β^*) to P_S.
6. If session (P_S, P_C, s) is chosen by $\mathcal{A}$ as the test-session, then provide $\mathcal{A}$ with γ^* as the answer to this query.
7. If the r-th session (P_S, P_C, s) is ever exposed, or if a session different than the r-th session is chosen as the test-session, or if $\mathcal{A}$ halts without choosing a test-session then $\mathcal{D}$ output $b' \leftarrow_R \{0,1\}$ and halts.
8. If $\mathcal{A}$ halts and outputs a bit b', then $\mathcal{D}$ halts and output b' too.

Figure 7 Distinguisher $\mathcal{D}$

First note that the run of $\mathcal{A}$ by $\mathcal{D}$ is identical to a normal run of $\mathcal{A}$ against protocol ECDH.

Consider the case in which the test session coincides with the r-th session, and then the response to the test-query by $\mathcal{A}$ is γ^*. In addition, input to $\mathcal{D}$ was chosen with probability that $1/2$ from Q_0 and Q_1, and the advantage that $\mathcal{A}$ guesses correctly whether the test value was "real" or "random" is ε. Thus the distinguisher $\mathcal{D}$ guesses correctly the input distribution Q_0 or Q_1 with the same probability $1/2 + \varepsilon$ as $\mathcal{A}$ did.

Now consider the case in which the r-th session is not chosen as a test-session. In this case $\mathcal{D}$ always ends outputting a random bit, and thus its probability to guess correctly the input distribution is $1/2$.

Since the first case happens with probability $1/l$ while the other case happens with probability $1-1/l$ we get that the overall probability of $\mathcal{D}$ succeeds in distinguishing Q_0 from Q_1 with non-negligible advantage.

1. AE $\rightarrow$ ASUE: m
2. ASUE $\rightarrow$ AE: m, N_{ASUE}
3. AE $\rightarrow$ ASUE: $m, SIGN_{AE}(m, N_{ASUE}, P_{ASUE})$

Figure 8 Signature-based MT-authenticator

Applying the signature-based authenticator in Figure 8 to each of the flows in ECDH protocol and joining (piggy-baking) the common flows, then we can get protocol NAKE in UM. Follows from Theorems 1 and 2, NAKE is a SK-secure protocol under UM. □

4.3. Security Proof of WAI'2 Protocol

Similar to the proof method of WAI'1 protocol, we provide the secure REKEY protocol [5] under AM first, see Figure. 9. REKEY protocol only meets SK-security without PFS under AM. The security of REKEY protocol is based on the existing of pseudo-random function. Due to the space limitations, proof is not provided in detail.

Premise: AE and ASUE share BK and pseudo-random function f

Goal: AE and ASUE share a session key $k = f_{bk}(ADDID \| N_{AE} \| N_{ASUE})$

1. AE $\rightarrow$ ASUE: s, MAC_{AE}, N_{AE}

2. ASUE $\rightarrow$ AE: s, MAC_{ASUE}, N_{ASUE}

Figure 9 Pre-shared key based protocol REKEY in AM

Applying the authenticator in Figure 10 to each of the flows in REKEY protocol and joining (piggy-baking) the common flows, then we can get protocol WAI'2 in UM. Follows from Theorems 1 and 2, WAI'2 is a SK-secure without PFS protocol under UM.

Premise: AE and ASUE share bk and HMAC

1. AE $\rightarrow$ ASUE: m

2. ASUE $\rightarrow$ AE: r

3. AE $\rightarrow$ ASUE: $m, HMAC_{bk}\left(MAC_{ASUE}, r, m\right)$

Figure10 MT-authenticator based on MAC

4.4. Discussion

We can see from the above analysis that the protocol WAI' can offer the security proof under CK security model, then it has the following security attributes.

Mutual authentication

Both parties carry on mutual authentication through public key certificates. The result of the certificate authentication and signature by ASE guarantees the legitimacy and authenticity of the certificate, thus WAI protocol realizes the security goal of mutual authentication.

Mutual key agreement and control

Protocol is based on Diffie-Hellman key exchange. The freshness of session key can guarantee the random number is appropriately selected. Both sole recognized this key by both sides to key figure of material sign to guarantee to enjoy alone. Security parameters α and β are selected randomly by the AE and ASUE, respectively. Thus, AE and ASUE are beyond the control of key generation.

Mutual key confirm

At the end of the protocol, AE and ASUE produce the hash value $HMAC_{BK}(MAC_{AE}, N_{AE}, s, N_{ASUE})$ and $HMAC_{BK}(MAC_{ASUE}, N_{ASUE}, s, N_{AE})$ respectively, moreover the two sides can ensure that they have a specific key.

Perfect forward secrecy

Session key is established by Diffie-Hellman key exchange, thus, WAI' protocol has the attractive property of PFS. The establishment of the unicast session key is based on the protocol REKEY, this protocol itself dose not have PFS characteristic. However, it is an inalienable whole that the certificate distinguishes process and sows the key and consults the process only, even if the long-term keys of AE and ASUE are let out, base key BK set up before will not let out this time and click, so PFS is a attribute of WAI'.

5. Conclusion

WAPI is the first WLAN protocol standard that China develops by herself, its implementation will play a very important role to the development in the fields of protocol standard formulation and wireless communication security etc. By doing in-depth analysis to WAPI standard and the implementation of the guidelines to carry on, this paper points out the security flaws existed in WAI and proposes an improvement protocol WAI'. The proposed scheme has all security attributes

required by WLAN which can realize information privacy and identity authentication, guarantee data integrity and the security goal of inserting control, and provide perfect forward secrecy as well as the resistance to known key attacks. Thereby, this new protocol WAI' ensures the information security in wireless communication, which is of great theoretical and practical meaning.

References

[1] GB 15629.11-2003-XG1-2006. Information technology-telecommunications and information exchange between systems-local and metropolitan area network-specific requirements-parts 11: wireless lan medium access contro (mac) and physical layer (phy) specifications amendment 1, 2006.

[2] F.Zhang and J.Ma. Security analysis on Chinese wireless lan standard and its solution. In 34th International Conference on Parallel Processing Workshops, pages 436-443. IEEE Computer Society, 2005.

[3] LI Xiehua, LI Jianhua, YANG Shutang etc. Formal Analysis and Verification for Authentication Process of WAPI. Computer Engineering, 2006, 32（22）: 10－13

[4] Colin Boyd, Anish Mathuria.Protocols for Authentication and Key Establishment. Berlin: Springer-Verlag, 2003

[5] Canetti R, Krawczyk H. Analysis of Key-Exchange Protocols and Their Use for Building Secure Channels. In:Pfitzmann ed.Proceedings of Eurocrpt'01. Lecture Notes in Computer Science 2045. Berlin: Springer-Verlag, 2001, 453-474

[6] Bellare M, Canetti R, Krawczyk H, et al. A Modular Approach to the Design and Analysis of Authentication and Key Exchange Protocols. In: Proc. of the 30th Annual Symp. on the Theory of Computing. New York: ACM Press, 1998. 419-428