An Enhanced Lightweight Authentication Protocol for Access Control in Wireless LANs

Kui Ren, Hyunrok Lee, Kyusuk Han, Jaemin Park and Kwangjo Kim
IRIS, Information and Communications University
Daejon, 305-714 Korea
{kren, tank, hankyusuk, jaeminpark, kkj}@icu.ac.kr

***Abstract*— In this paper, an enhanced lightweight identity authentication protocol for access control in IEEE 802.11 networks are presented. The proposed protocol is nicely integrated with the current MAC frame structure and takes the most advantage of the redundancy bits inside the MAC frame header to convey the authentication information, as well as the synchronization information in case of synchronization loss happening. A much more efficient and fault-tolerant synchronization algorithm is given at the same time, which significantly improved the performance of the proposed protocol as compared to the previous ones. The proposed protocol is highly effective as evaluated via a thorough mathematical analysis. A quantitive attack detection framework is also established based on the evaluation result. Finally, the proposed protocol is well suited in a wireless constrained environment for its low communication and computation overheads, requiring only several additional bits (less than 8) for transmission and random bit generating operation.**

I. Introduction

While wireless local area networks (WLANs) have come into great use in recent years, the security over wireless networks is becoming a significant issue. It is evident that anyone with a radio receiver can eavesdrop on a WLAN, and therefore widely acknowledged that a WLAN needs a mechanism to counter this threat. It is less understood but equally true that anyone with a transmitter can write messages to a WLAN, rendering access controls meaningless. Because forgeries are easy to create, a WLAN needs mechanisms to counter this threat, too [10].

Wired Equivalent Privacy (WEP) is the security scheme defined in the IEEE 802.11 [2]. Several research studies report the vulnerabilities of WEP and concluded that WEP is insecure [3], [4], [10]. Several alternative schemes have been proposed to make wireless networks more secure [11]. Currently, the most commonly used scheme for commercial applications is to combine virtual private network (VPN) and WEP to ensure wireless security. As VPN has already been existing as a security architecture, it was an obvious choice for adding security to the wireless environment. IP tunnelling is a significant technique used in VPN. If the IP tunnelling scheme namely IPSec/VPN [5], [6] is used, the fixed host is responsible for all the authentication process, which means all the authentication information will be exchanged between the mobile host and the fixed host. Although it is practical, the responsibility is low obviously. If both the WEP and IP tunnel are used, the responsibility and security problems are solved. But this causes a redundancy problem. Each frame sent out from the mobile stations is encrypted twice for authentication, *i.e.* one for IP tunnel, and one for WEP. This is unnecessary and wastes rare computing resource at the mobile hosts. Thus, this paper provides a lightweight authentication scheme to replace WEP. With both the proposed scheme and IP tunnel, the system can solve the redundancy problem and keep high responsibility and high security.

Several lightweight authentication protocols are proposed by using only single authentication bit and thus achieve a higher efficiency and lower communication overhead [1], [7], [8]. Canetti *et al.*, first described an authentication scheme using a Message Authentication Code with a single bit output that is 0.5-per message unforgeable, and the scheme uses a MAC with a single bit output from current constructions of MACs [8]. Johnson *et al.*, applied this concept and proposed an one bit authentication protocol attempted to solve the problem of redundancy in wireless security and provide identity authentication at the MAC layer [1]. A severe synchronization problem exists in their work due to the frame loss problem in the error-prone wireless communication environment, and hence it is unlikely to solve the problem [7]. Also in their work, the authentication bit is generated through a random bit generator instead of MAC. Latterly, Wang *et al.*, developed an improved synchronization algorithm for the above work [7]. However, the proposed algorithm is still not efficient enough. The number of synchronization runs required to recover from non-synchronization is linear to the distance between the two communication parties' positions in their respective authentication bit stream. As frame loss happens frequently in wireless communications [9], non-synchronization between the communication parties occurs frequently too. A large number of synchronization rounds means high communication overheads, which waste a lot of limited wireless channel resource; it also results in additional communication delay, which could be critical to many realtime applications.

To make the thing worse, a large number of synchronization rounds seriously weakens the performance of the proposed authentication scheme. The probability to distinguish an attack from a normal recovery procedure performed by the legal host decreases accordingly as the required number of synchronization rounds increases. Another serious drawback of the previous schemes is that the authentication bit stream generator may lose synchronization itself, and an arbitrary may occur between the sender and the receiver. But both of the above schemes provide no mechanism to solve this problem.

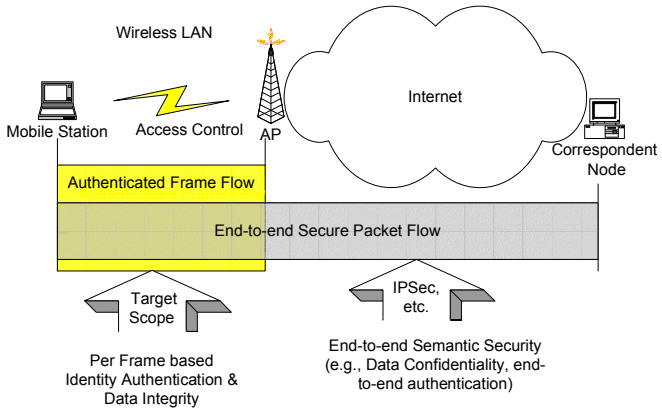

Fig. 1. Security service coverage of the proposed protocol

Moreover, the performance analysis in [7] is quite rough and thus inaccurate.

In view of the drawbacks identified above, this paper proposes an enhanced lightweight authentication protocol for access control at the MAC layer in wireless LAN. By carefully examining the redundancy existed in the MAC header, we adopt an enhanced 3-bit authentication mechanism, which provides a higher probability for attack detection compared to that of the one-bit schemes. Further, we develop a much more efficient and fault-tolerant synchronization algorithm, which is designed to correct the system non-synchronization due to the frame losses, as well as the state loss of the authentication bit stream generator itself. At the same time, The proposed protocol is fully compatible with the existing frame structure of IEEE 802.11. This means the proposed scheme doesn't modify the frame structure and is compatible with legacy devices which do not use the authentication scheme. The major purpose of the proposed protocol is to detect an attack in an error-prone wireless environment. When the system detects an attack, some protection or anti-attack approaches for each type attack can be triggered. The proposed protocol identifies the attack by using a statistical way and provides access control. Data confidentiality, integrity and key management are out of the scope of this paper and can be done at the higher layers (*e.g.*, IP layer).

The rest of the paper is organized as follows. Section II describes the framework of our authentication protocol. Section III gives the details of the synchronization algorithm and its analysis. In Section IV, we discuss the implementation issue. In Section V, we describe the statistical method used to identify the attacks and its performance evaluation in the first part, and then give the security analysis in the second part. Finally, Section VI concludes the paper and mentions future work.

II. The Proposed Lightweight Authentication Protocol

The proposed protocol provides a per-frame based 3-bit authentication mechanism at the MAC layer for wireless LANs. We indicate the protocol service scope in Fig.1. The

ACK frame received	Data frame received		
	Authenticated	Lost/Error	Invalid
ACK-success	✓	/	/
ACK-failure	/	/	✓
ACK-success lost	✓	/	/
ACK-failure lost	/	/	✓

TABLE I

All the possible communication error types

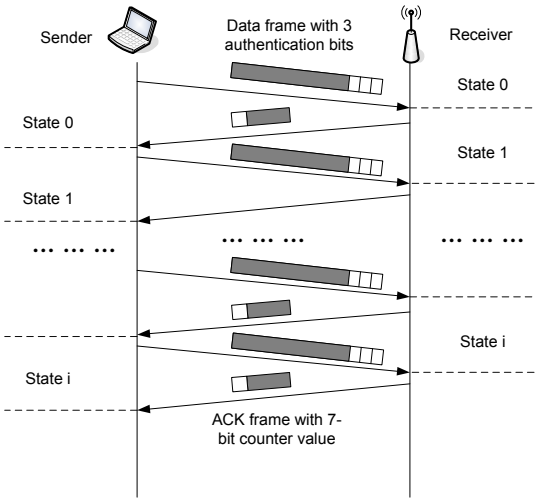

Fig. 2. Overview of the proposed protocol

identity authentication is achieved by continuously checking a series of frames transmitted by the given host.

Like other security solutions in Wireless LAN, the proposed protocol is also established on pre-shared secret key between the mobile station and AP. Specifically, in our protocol, a mobile station S_i is pre-configured a unique set of secret credentials with AP:

$\{K_{AP-S_i}, SV_{AP\rightarrow S_i}, SV_{S_i\rightarrow AP}, C_{AP\rightarrow S_i}, C_{S_i\rightarrow AP}\}$,

where K_{AP-S_i} is a shared secret key between AP and S_i, which is used to generate new seed values when the previous authentication bit stream is exhausted. $SV_{AP\rightarrow S_i}$ and $SV_{S_i\rightarrow AP}$ are two 64-bit secret seed values of a mutual agreed pseudo random bit stream generator, each of which is used unidirectionally. Therefore, the period of authentication bit stream can be up to 2^{64}. The length of seed value also could be adjusted according to the underlying data transmission rate so that a desirable seed value update interval could be achieved to avoid high key management overhead. $C_{AP\rightarrow S_i}$ and $C_{S_i\rightarrow AP}$ are two random 64-bit counters, each of which is also used unidirectionally. The counters are used for synchronization algorithm to provide high efficient synchronization recovery service. Note that the sequence number contained in IEEE 802.11 MAC header can't be used for our purpose. The reason is that the counter as well as the bits in the authentication bit stream should still advance ahead even if the frame is a retransmitted one (In this case, there is no change in sequence number.).

Conceptually, the proposed protocol works as follows: At the beginning, the sender and the receiver establish a random bit stream generator by using the shared seed value. The established random bit stream generator continuously outputs

//Receiver receives the data frame with 3-Bit authentication unit AU_{sender}
$Counter_{receiver}++ \ \ (mod 2^{64})$; //a 64-bit integer used to store the position information
    update $AU_{receiver}$ *with the next 3 bits generated by the generator*;
if $AU_{sender} == AU_{receiver}$ *then*
    reply sender "ACK-success";
else
    $C_{ACK} = Counter_{receiver} \ \ (mod 128)$; //a 7-bit integer used to store position information in ACK frame header
    insert C_{ACK} *into "ACK-failure"*;
    reply sender "ACK-failure";

//Sender receives the ACK frame
if ACK == "ACK-success" then
    $Counter_{sender}++ \ \ (mod 2^{64})$;
    update AU_{sender};
else if ACK == "ACK-failure" then
    extract C_{ACK} *from "ACK-failure"*;
    $C_{sender} = Counter_{sender} \ \ (mod 128)$; //a 7-bit integer used to store position information
    if $C_{ACK} > C_{sender}$ *then*
        $Counter_{sender} = Counter_{sender} + (C_{ACK} - C_{sender}) \ \ (mod 2^{64})$;
        update AU_{sender} *with the next* $(3 \cdot (C_{ACK} - C_{sender}) + i) - th \ (i = 1, 2, 3)$ *bits generated by the generator*;
    else
        $Counter_{sender} = Counter_{sender} + (C_{ACK} - C_{sender} + 128) \ \ (mod 2^{64})$;
        update AU_{sender} *with the next* $(3 \cdot (C_{ACK} - C_{sender} + 128) + i) - th \ (i = 1, 2, 3)$ *bits generated by the generator*;

TABLE II
PSEUDO CODE OF OUR SYNCHRONIZATION ALGORITHM

3 bits as a unit each time and stores it locally. Obviously, the sender and the receiver will generate the same authentication bit streams and they are synchronized initially. We call that both the sender and the receiver are at the same state (*i.e.*, the same counter value and the same 3-bit authentication unit).

When the sender is going to send a frame, it attaches the 3-bit authentication unit to the frame. Upon receiving a frame, the receiver first checks the 3-bit authentication unit value in the MAC header. If the value matches that of the receiver's, which is independently generated by the receiver, the frame is authenticated and is passed to the upper layer for further processing. The receiver, at this time, also increases its counter value by one (modular 2^{64}) and generates a new 3-bit authentication unit to replace the old one. Then the receiver replies an ACK-success to the sender. On the contrary, if the two value does't match, the frame will be rejected and an ACK-failure will be sent after the receiver updates its state in the same way as above. The ACK-failure frame contains the necessary information needed by the sender to be recovered from the non-synchronization and will trigger the synchronization scheme at the sender. More specifically, the ACK-failure frame contains the current counter value (modular 128). We will describe it in details later. When the sender receives the ACK-success frame, it performs the same state update operation as the receiver does. If the received frame is ACK-failure, it will first adjust its own counter value, compute the corresponding 3-bit authentication unit, and then retransmit the previous frame attaching the newly obtained 3-bit authentication unit.

An authentication failure at the receiver side implies two possible reasons: i) there is no synchronization between the sender and the receiver authentication bit pointers or ii) the sender is an illegitimate host. Due to error-prone nature of wireless communications, the first reason happens frequently and causes non-synchronization between the sender and the receiver from time to time. In order to distinguish an attack from non-synchronization, it is important to clarify the effects of different error types posed on the system synchronization. We summarize all the possible error types in Table 1 and discuss their respective influences below.

In IEEE 802.11, when a data frame is lost, the sender waits for ACK timeout and retransmits the frame. At this point, the sender and the receiver's counter will not be increased. Thus, the system still will be synchronous.

When an invalid data frame doesn't pass the authentication, the receiver's counter still increases by one and the 3-bit authentication unit is updated accordingly. But the ACK-failure frame would trigger a synchronization operation at the sender side. Therefore, once the ACK-failure is correctly received by the sender, non-synchronization will be corrected in our protocol. Note that even if the invalid data frame is due to synchronization lost of the sender's random authentication bit stream generator itself, still the sender can recover from the non-synchronization by using the counter value provided in the ACK-failure frame. In the previous schemes, if either side lost synchronization in their random authentication bit stream generator, there's no way for them to relocating the right position in the authentication stream. On the other side, if the ACK-failure frame is lost, the sender will finally timeout and retransmit the data frame without update its state. Therefore, the retransmitted data frame will fail to pass the authentication the receiver again, and trigger another ACK-failure frame. Hence, ACK-failure frame loss will result in non-synchronization.

When a data frame is correctly received and authenticated, the receiver will update its state and reply an ACK-success to the sender. If wireless error happens in the transmission of ACK-success frame, the sender will not get the ACK-success and no state update operation is performed. Upon timeout, the sender will retransmit the previous data frame and therefore, non-synchronization occurs. Observe that even if the system is non-synchronized, the data frame may still pass the authentication with probability 0.125, that is, the sender's current 3-bit authentication unit happens to be the same as

the receiver's although their position in the authentication bit stream is not the same.

We conclude this section by showing an overview of our proposed protocol in Fig. 2.

III. Synchronization Algorithm of the Proposed Protocol

The synchronization scheme is a crucial component for the efficiency of the whole system. We describe the pseudo code of our synchronization algorithm in Table 2. We also have the following conclusion:

Lemma 1: When the system is not synchronized, the receiver's counter is always greater than the sender's.

Proof: The sender's counter is allowed to increase only when sender receives ACK-success, which implies the receiver's pointer has already increased before this transmission. Note that the receiver increases its counter no matter whether the received data frame passed the authentication or not. Thus, in all the cases, the receiver's counter always advances earlier than the sender's. Then, when the ACK fails to arrive at the sender, this causes non-synchronization and makes the sender's pointer one lower than the receiver's.

Lemma 2: When non-synchronization is detected by the receiver, it takes exact one round for the system to regain synchronization if only the ACK-failure frame is correctly received by the sender.

Proof: As specified in **Lemma 1**, the receiver's counter is always greater than the sender's. Thus, when the sender gets the counter value from the ACK-failure frame, it can easily examine the difference between the two counters and adjust the 3-bit authentication unit to the correct position. The system is thus regain synchronized. Also note that due to the limited redundant space in the header of the MAC control frame, only a 7-bit counter value could be sent to the sender via ACK frame. Therefore, the above conclusion is safely drawn based on the assumption that the difference of the two counter values is less than $2^7 = 128$, which implies that there are less than 128 continuous ACK-failure frame loss.

We give an concrete example in Fig. 3. The system is non-synchronized at first. Because the authentication unit happens to be the same, the receiver fail to detect the non-synchronization at the first round. We also find that once the receiver detects the non-synchronization, the sender will adjust its counter value and corresponding 3-bit authentication unit immediately as long as the ACK-failure frame is correctly received by the sender.

In case that the authentication bit generator loses its state synchronization itself, the host can regenerate the whole bit stream and relocate the right position in the authentication bit stream according to the value of the stored local counter.

IV. Implementation of the Proposed Protocol

In this section, we describe the implementation details of the proposed protocol. As we mentioned before, our protocol is fully compatible with current IEEE 802.11 frame structure [2].

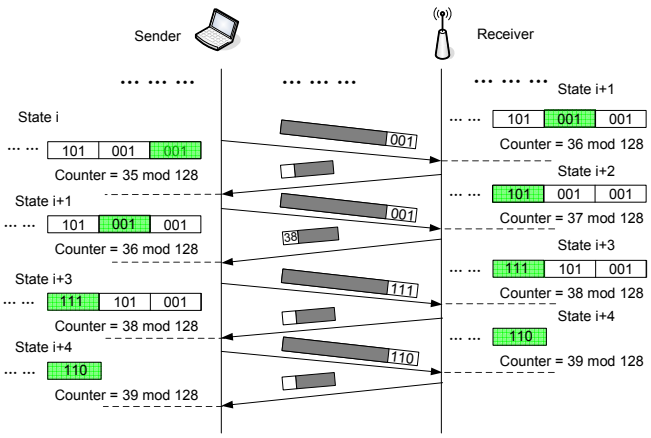

Fig. 3. An example of synchronization operation

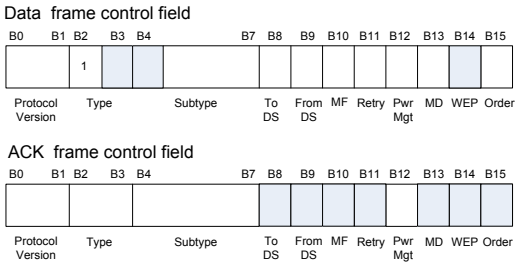

Fig. 4. Adaption of frame format to the proposed protocol

We first discuss how to insert the 3-bit authentication unit to the MAC header without interfering with the current settings by utilizing the redundant bits in the *Frame Control* field as shown in Fig.4. More specifically, we can use the following 3 bits: B3, B4 and B14. We know that *Type* has 2 bits and includes 4 possible types: 00 for management frame, 01 for control frame, 10 for data frame and 11 is reserved. We also know that the *Subtype* field consists of 4 bits, and for data frame type, only 3 of them are used to defined the corresponding subtypes and the remaining values 1000-1111 are reserved. It is easy to observe that a data frame type can be easily judged by bit B2 only, and its corresponding subtypes can be judged by the last 3 bits. Therefore, the bits B3 and B4 both can be used to insert the authentication information. Also the WEP bit can be used for insert another authentication bit because the proposed scheme is aimed to replace WEP.

When the ACK-failure frame is sent, we need to insert 7-bit counter value to the frame. As shown in Fig.4, seven corresponding bits are chosen based on the structure of the frame control field of the control frame. It is known that the above 7 bits in the control frame are simply set to be 0. We make use of these 7 bits to convey the position information to the sender.

V. Analysis of the Proposed Protocol

In this section, we first use a statistical method to analyze the performance of the proposed protocol and then discuss its security property in the following subsection.

A. Performance analysis of the proposed protocol

We use a statistical method to analyze the attack detection ability of the proposed protocol. The proposed protocol detects a potential attack by continuously checking the received frame sequence.

From the protocol description we can easily find that to forge a valid single frame, an attacker need to provide a right 3-bit value, the probability of which is $2^{-3} = 0.125$. It is easy to show that in a perfect channel without any loss, an attacker has a probability of 2^{-3w} to successfully cheat the receiver, given the authentication window size of w. However, in the error-prone wireless communication environment, the reason of a data frame, which is fail to pass the authentication, may due to the non-synchronization between the sender and the receiver caused by frame losses. Hence, we need a statistical method to be using by the receiver to efficiently detect the potential attacks with high probability. We have the following theorem.

Theorem 1: In the proposed protocol, given the conditions that i)the *priori* probability of a sender to be an attacker is 0.5, *i.e.*, $Pr(A) = Pr(L)(A = attacker, L = legal\ host)$; ii)the receiver's authentication window size is set to be w; iii)the number of data frames that fail to pass the authentication at the receiver side within the last w frames, is s; iv) the bit error ratio (BER) of the given wireless channel is p, the probability of the sender to be an attacker is given by

$$Pr(A|w,s) = \frac{0.125^{w-s}0.875^{s}}{0.125^{w-s}0.875^{s} + \overline{p}^{112(w-s)}(1-\overline{p}^{112})^{s}},$$

where $\overline{p} = 1 - p$.

Before proceeding to prove **Theorem 1**, we first prove **Lemma 3** and **Lemma 4**.

Lemma 3: In the proposed protocol, whenever one ACK frame is lost, exactly one invalid data frame will fail to pass the authentication at the receiver side. If more ACK frames are lost no matter whether they are continuous or not, the exactly the same number of invalid data frames will fail to pass the authentication at the receiver side.

Proof: In the proposed protocol, whenever a ACK-success frame is lost, exactly one data frame subsequently sent by the sender will contain an invalid 3-bit authentication unit and thus fail to pass the authentication. The triggered ACK-failure frame sent by the receiver will help the sender correct itself and regain synchronized with the sender immediately, therefore, no more invalid frame will be sent by the sender. Once the ACK-failure frame is lost, another invalid data frame will be sent to the receiver because of the system non-synchronization. If the ACK-failure frames continue to be lost, exactly the same number of the invalid data frames will triggered at the sender side. Ultimately, once one ACK-failure frame arrives at the sender successfully, the system regains synchronized immediately and no more invalid data frame will be generated. The above property holds because the proposed synchronization algorithm takes exactly one round for both the sender and the receiver to return to the synchronization state.

Lemma 4: If the given wireless channel has a BER of p, the ACK frame loss rate r of the given channel is given by

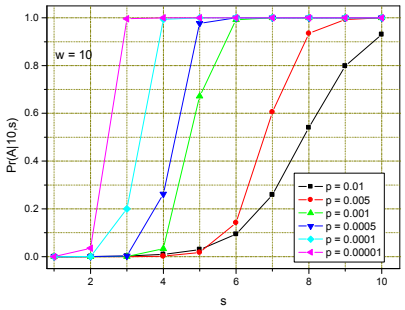

Fig. 5. Probability to detect an attacker with $s = 1 \sim 10, w = 10$

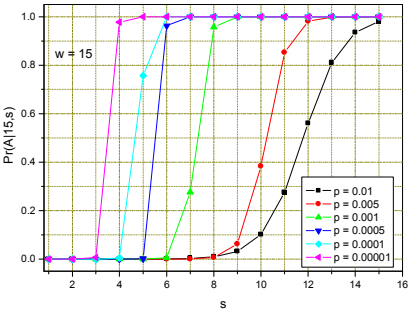

Fig. 6. Probability to detect an attacker with $s = 1 \sim 15, w = 15$

$$r = 1 - (1-p)^{112}.$$

Proof: Given the probability of any bit being in error is p, then the probability of an n-bit frame being received entirely correctly is $(1-p)^n$. We know that the length of the control frame in IEEE 802.11 is 14 bytes, *i.e.*, 112 bits. Hence, we obtain the above result.

Proof of Theorem 1:

According to Bayer's formula, we have

$$Pr(A|w,s) = \frac{Pr(w,s|A)Pr(A)}{Pr(w,s|A)Pr(A) + Pr(w,s|L)Pr(L)}$$

$$= \frac{Pr(w,s|A)}{Pr(w,s|A) + Pr(w,s|L)}.$$

We first consider $Pr(w,s|A)$: Because the illegitimate sender can't compute the secret authentication bit stream, the probability for him to generate a valid frame is 0.125 as discussed above. Hence, we have

$$Pr(w,s|A) = C_{w-s}^{w}0.125^{w-s}(0.875)^{s}.$$

Next, we consider $Pr(w,s|L)$: In **lemma 3**, we have proved that the number of invalid data frames is equal to that of the lost ACK frames. Hence, we have $Pr(w,s|L) = Pr(w,m|L)$, where m represents the number of the lost ACK frames. Let the ACK frame loss rate be r. It is easy to calculate

$$Pr(w,m|L) = C_{s}^{w}r^{s}(1-r)^{w-s}.$$

By combing above three equations, we can easily have

$$Pr(A|w,s) = \frac{0.125^{w-s}0.875^{s}}{0.125^{w-s}0.875^{s} + r^{s}(1-r)^{w-s}}.$$

Finally, by applying **Lemma 4**, we obtain that

$$Pr(A|w,s) = \frac{0.125^{w-s}0.875^{s}}{0.125^{w-s}0.875^{s} + \overline{p}^{112(w-s)}(1-\overline{p}^{112})^{s}},$$

where $\overline{p} = 1 - p$.

Hence, we complete the proof of **Theorem 1**.

Fig.5 and Fig.6 show the performance of the proposed scheme. We can find that when the wireless channel BER p is relatively high, it is hard to distinguish an attack from non-synchronization between the sender and the receiver. For example, when $p = 0.01$ (Note that generally, the BER of a wireless channel is no larger than 0.0001 [9].), the successful rate to correctly transmit a ACK frame is near to zero, that is, the ACK frame loss rate is near to 1. Note that the number of ACK frame loss equals to that of invalid data frames. Therefore, it's very hard to detect an attacker, because the sender itself is sending invalid frames with a probability near to 1. However, as p decreases, it is becoming more and more easier to detect an attack. Also we can find that when the p is lower than 0.001, an attack can be identified with a probability larger than 0.95, when the number of invalid frames s is large than half of the authentication window size w. Further, when p is relatively low, it is very easy to detect an attack in the proposed protocol. For instance, we can identify an attack with confidence larger than 0.95, when $s = 4$ and $w = 15$.

More generally, when the attached authentication unit is n bits and the *priori* probability of a sender to be a illegitimate host is $x(0 \leq x \leq 1)$, similarly we have **Theorem 2**.

Theorem 2: In the proposed protocol, given the conditions that i) $Pr(A) = x$ and $Pr(L) = 1 - x$; ii)the receiver's authentication window size is set to be w; iii)the number of data frames that fail to pass the authentication at the receiver side within the last w frames, is $s(0 \leq s \leq w)$; iv) the bit error ratio (BER) of the given wireless channel is $p(0 \leq p \leq 1)$; v)the authentication unit contains n bits, the probability of the sender to be an attacker is given by

$Pr(A|w,s) =$

$$\frac{x2^{n(s-w)}(1-2^{-n})^{s}}{x2^{n(s-w)}(1-2^{-n})^{s} + (1-x)\overline{p}^{112(w-s)}(1-\overline{p}^{112})^{s}},$$

where $\overline{p} = 1 - p$.

Hence, a highly effective attack detection framework can be established based on **Theorem 2**. We can efficiently evaluate a potential attack as a function of n, x, w, s and p. Among them, x and p can be set as environment parameters and usually are fixed for a given wireless communication system, while s and w can be adjusted to improve performance. In the proposed protocol, n is set as 3, because of the limitation of the redundancy bits in the MAC header. A simple analysis can be easily shown that as the value of n increases, it becomes easier to detect a potential attack.

B. Security analysis of the proposed protocol

The proposed protocol provides a high efficient identity authentication scheme in terms of both communication and computation for IEEE 802.11. It's major purpose is to detect an attack, and it offers a statistical way to identify the origin of the data frame for the purpose of access control. In [1], the authors identified four types of potential attacks against this type of protocols: Denial-of-service attack, overwrite attack, Man-in-the-middle attack, and authentication bits guessing attack. By basically a same discussion as in [1], we can have the same conclusion that the proposed protocol is immune to above four types of attacks.

One point addressed here is that one may argue that as a type of Man-in-the-middle attack, an attacker may forge a valid data frame simply by modifying the payload field, while keeping the other part untouched, and in this case the attacker may aim to steal the bandwidth from the legal sender or insert the content into the frame as desired by the attacker itself. However, the kind of attack would not succeed. If the attacker is trying to steal the bandwidth, the legal host will not advance in the identity authentication stream and send new TCP data packets due to the nature of the absence of TCP response packets. If the attacker's goal is the later one, it can be easily prohibited by the message integrity protection method adopted at the high layer (*e.g.*, IPsec at network layer). We have pointed out at the beginning of this paper that it is beyond the scope of this paper.

VI. Conclusion

In this paper, an enhanced lightweight identity authentication protocol for access control in IEEE 802.11 networks are presented. The proposed protocol is nicely integrated with the current MAC frame structure and takes the most advantage of the redundancy bits inside the MAC frame header to convey the authentication information, as well as the synchronization information in case of synchronization loss happening. A much more efficient and fault-tolerant synchronization algorithm is given at the same time, which significantly improved the performance of the proposed protocol as compared to the previous ones. The proposed protocol is highly effective as evaluated via a thorough mathematical analysis. A quantitive attack detection framework is also established based on the evaluation result. Finally, the proposed protocol is well suited in a wireless constrained environment for its low communication and computation overheads, requiring only several additional bits (less than 8) for transmission and random bit generating operation. As for the future work, we would like to implement the proposed protocol into the real wireless network system to further evaluate it.

References

[1] H. Johnson, A. Nilsson, J. Fu, S. Wu, A. Chen and H. Huang, "SOLA: A One-bit Identity Authentication Protocol for Access Control in IEEE 802.11", In Proceedings of IEEE GLOBECOM, Taipei, Taiwan, September 2002

[2] "LAN MAN Standards of the IEEE Computer Society. Wireless LAN medium access control (MAC) and physical layer (PHY) specification. IEEE Standard 802.11, 1997 Edition," 1997.

[3] J. Walker, “Unsafe at any key size: an analysis of the WEP encapsulation, ” Tech. Rep. 03628E, IEEE 802.11 committee, March 2000.
[4] N. Borisov, I. Goldberg, and D. Wagner, “Intercepting Mobile Communications: The Insecurity of 802.11.”
[5] S. Kent and R. Atkinson, “Security Architecture for the Internet Protocol”, RFC 2401, Nov. 1998.
[6] R. Thayer, N. Doraswamy and R. Glenn, “IP Security Document Road Map”, RFC 2411, Nov. 1998.
[7] H. Wang, A. Velayutham, and Y. Guan, ”A Lightweight Authentication Protocol for Access Control in IEEE 802.11,” in Proceedings of IEEE Globecom 2003, San Francisco, CA, December 1-5, 2003.
[8] R. Canetti, J. Garay, G. Itkis, D. Micciancio, M. Naor, and B. Pinkas “Multicast Security: A Taxonomy and Efficient Constructions.” In Proc. INFOCOM ’99, Vol. 2, pp. 708-716, New York, NY, March 1999.
[9] A. Tanenbaum, “Computer Networks”, Fourth Edition, Prentice Hall, 2003.
[10] N. Cam-Winget, R. Housley, D. Wagner, and J. Walker, “Security flaws in 802.11 data link protocols, COMMUNICATIONS OF THE ACM, May 2003/Vol. 46, No. 5, pp 35-39, 2003.
[11] Ferguson, N. Michael: An improved MIC for 802.11 WEP. IEEE 802.11 doc 02-020r0, Jan. 17, 2002; grouper.ieee.org/groups/802/11.