Improved Security in IEEE 802.11 Wireless LANs

FAHAD SAMAD, WAQAR MAHMOOD, ARSHAD ALI, UMAR KALIM
Department of Information Technology (NIIT)
National University of Science & Technology (NUST)
H. # 166-A, Street # 9, Chaklala Scheme III, Rawalpindi
PAKISTAN
fahadsamad@niit.edu.pk

Abstract: - Attempts such as the standard 802.11i, in the domain of wireless communication, have been successful in securing the data-exchange procedures between hosts. However the secure exchange of management frames is yet to be addressed. In this paper we propose a solution to address the vulnerabilities that exist in the exchange of management frames (as presented by the IEEE 802.11 standards). We employ the simplistic approach of Diffe-Hellman's algorithm to devise an algorithm for authentication and integrity checks and consequently prevent threats such as active eavesdropping, session hijacking, MitM and DoS attacks. We further verify our results by presenting a performance analysis on our simulation results. This solution provides a reliable platform for a secure session of communication between two or more hosts using the 802.11 standard. No change in standard is required.

Keywords: - IEEE 802.11 Standard, Management Frames, Diffie-Hellman Method, Wireless LAN Security, Media Access Control (MAC) Layer, Service Set Identifier (SSID).

1. Introduction

The world is really going Wireless. Firstly it is quite an attractive technology for known reasons. Secondly, these are portable and easily deployable. But networks need something more. Turning the networks to wireless doesn't mean that one is free of all the worries. One of the most important tasks in wireless network is to make it as secure as possible. Here we discuss IEEE 802.11i (especially its MAC layer architecture), its security loopholes and propose an architecture to make it secure.

In 802.11 at MAC layer, there are three types of frames [1, 2]. Management frames, Control Frames and Data Frames. To make the exchange of frames secure a technique called Wi-Fi Protected Access (WPA) with 802.1x Authentication has been introduced [3, 4]. This mechanism presumes that the management frames have already been exchanged successfully in order to make the data frame exchange secure. A secure solution is still needed in order to make the management exchange free of adversary's attack [5]. The wireless communication is done with unlicensed band so an eavesdropper tuning to a particular frequency can extract all the information [6].

In this paper we have proposed an architecture which can be utilized to make the management frames of 802.11i secure and in turn abating the vulnerability and threat. Using this architecture attacks like Active eavesdropping, Session Hijacking, MitM attack and Denial of Service attacks [5] can be detected and eliminated. In the end a performance comparison of the current standard and our architecture is also presented.

2. Background

To explain our proposed architecture it is essential to first know about the structure of management frames in 802.11i and then identify how communication takes place using these frames by using the Robust Security Network Association (RSNA) establishment procedure [5].

The access point (AP) after becoming active, continuously broadcasts beacon frames to ask if there is any client or STA available and on the other hand showing its own availability. In this frame (Beacon) the AP sends its SSID (Service Set ID), which is a 32-byte alphanumeric value as a unique identifier for the Access point. This SSID is sent in plain text [1, 2, and 5].

The client station (STA) on the other hand starts communication by sending a probe request. This frame enquires about if there is any AP available in the vicinity with which the STA can associate itself. If there is any AP available, the STA responds the request with probe reply frame having almost the same information it sends in the beacon frames. Here too, the SSID goes in plain text [1, 2, and 5]. This SSID value is an identifier which is used by the AP to authenticate the STA. So the client (STA) that has this value (showing the AP's SSID) is a valid client for that AP. Now this is the point where the problem starts.

In Wireless LANs all communication takes place through air and the frequency band used is unlicensed [6]. Due to this reason, any adversary or illegal device can listen to the conversation between a valid AP and a valid STA just by tuning itself to that frequency.[6] So an adversary not only can steal useful information by doing this but it can also gain control of the conversation by impersonating itself as a valid party. This is one of the most important reasons why the data frames which have been made secure by the Wi-Fi Protected Access (WPA) by using Robust Security Network Association [5] become vulnerable.

In this paper we propose an improved security solution for 802.11i by securing management frames. We limit our work within a single Basic Service Set (BSS).

Now we discuss the management frames format in IEEE 802.11 standard.

3. Proposed Architecture

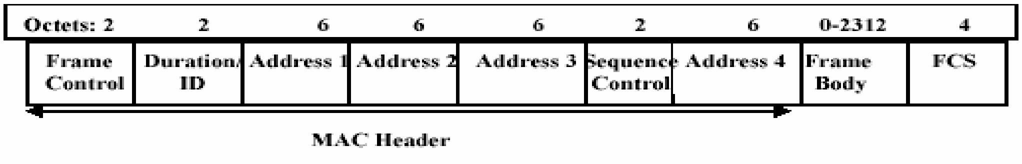
As discussed [1, 2, and 6] there are many types of management frames for example: beacon, probe request/response and authentication etc. For our architecture we first need to consider the General MAC-Frame Format which is shown in figure 1:

Out of the four address fields present in the general MAC frame, only three are used in the specific management frame format. These are Destination Address, Source Address and Basic Service Set Identifier (BSSID) respectively. The management frame is shown in figure 2.

The Management Frame doesn't use the last of the address fields in general frame (6-byte field). In order to secure the frame specially the SSID (in frame body), we utilize these bytes for the identification of valid source originating the frame. The technique we use is a "***modification***" of the well-known Diffie-Hellman Key- Agreement Method [7]. The term "modification" implies two things: Firstly it will serve the purpose of authentication whereas originally it is used as a method for generation of secret session keys. Secondly, hashing technique is used in combination of this algorithm which is not present in the original method. For Integrity, we use a second hash which is of the complete management frame type and place the hash value inside the frame body after the information related to the individual management frame types (i.e. beacon, probe request/response etc.).These all are explained with detail in the next sub-sections:

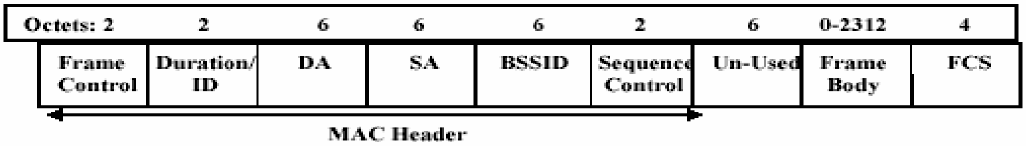
3.1 Modified Diffie-Hellman Method for Authentication

This method allows two hosts to generate a value Z by exchanging two randomly generated numbers each by one of the two parties. This value will be used for the authentication of the subsequent management frame types. The steps of the authentication key generation are as under:



Octets: 2	2	6	6	6	2	6	0-2312	4
Frame Control	Duration/ID	Address 1	Address 2	Address 3	Sequence Control	Address 4	Frame Body	FCS

Figure-1: IEEE 802.11 General-MAC Frame Format [1]



Octets: 2	2	6	6	6	2	6	0-2312	4
Frame Control	Duration/ID	DA	SA	BSSID	Sequence Control	Un-Used	Frame Body	FCS

Figure-2: 802.11 Management Frame Format [1]

i. First both the STA and AP must get the "Diffie-Hellman parameters”. A prime number, 'p' (larger than 2) and "base", 'g', an integer that is smaller than 'p'. The assumption here is that both will get g & p values “privately” by some external mechanism.

ii. The client (STA) will generate a random private number called ‘x’ which is less than "p - 1".

iii. The Access Point (AP) will generate a random private number called ‘y’ which is also less than "p - 1".

iv. Then each of the two parties evaluates one of the following values:

$$A = g^x \bmod p \quad and \quad B = g^y \bmod p$$

v. The first five bytes of both A & B are placed in the last 5 of the 6-byte field un-used in the management frame header. The remaining bytes of A and B will be placed in the available bytes within the frame body with total length of the remaining bytes before the actual hash bytes.

vi. After computing A & B each of the two parties exchanges these values in the first three management frames (beacon, probe request and probe response) [1, 2].

vii. Once these values are exchanged, AP calculates $A^y \bmod p$ and STA calculates $B^x \bmod p$ so that both of these get Z $= g^{xy} \bmod p$. This Z value will be used to authenticate both STA and AP which will be explained in section 3.3.

$$Z_{AP} = (A)^y \bmod p \qquad Z_{STA} = (B)^x \bmod p$$

$$\Rightarrow Z_{AP} = (g^x \bmod p)^y \bmod p$$

$$\Rightarrow Z_{STA} = (g^y \bmod p)^x \bmod p$$

By the laws of algebra: $Z_{AP} = Z_{STA} = Z$

3.2 Hashing for Integrity (External Hash)

In order to keep the integrity of the MAC frames, a hash of the complete management frame is computed and the value (hash) is placed in the frame body with its total length in bytes after the internal hash value (hash for authentication). Here the two parties will have choice to use either of the two message digest algorithms for calculating hashes. One is MD5 and the other is SHA-1. The 1 byte still un-used in the original management frame header will be used to identify that which message digest has been used for the particular frame. The table 1 below shows the code and the name of the algorithm:

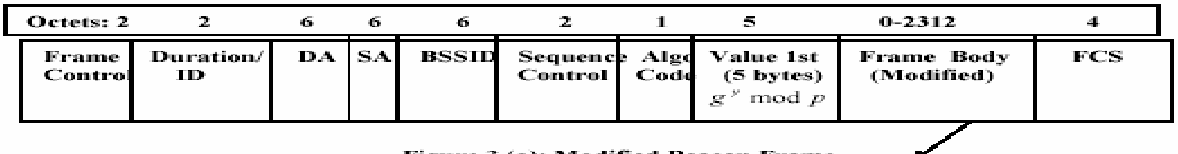
Algorithm Code	Name of the Hashing Algorithm
1	MD5
2	SHA-1

Table 1: Hash Algorithms for Integrity

3.3 Management Frames Message Exchanges

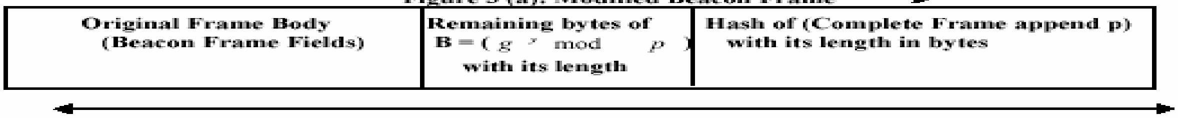
The message exchanges with modified management frames take place as under:

i. The Access Point (AP) repeatedly (after certain Beacon Interval) sends the beacon frames with the information shown in the figure 3(a) and 3(b). The appendaged p in the hash of the complete frame will make sure that the adversary cannot capture the frame and recalculate the hash after modifying it.



Octets: 2	2	6	6	6	2	1	5	0-2312	4
Frame Control	Duration/ ID	DA	SA	BSSID	Sequence Control	Algo Code	Value 1st (5 bytes) $g^y \bmod p$	Frame Body (Modified)	FCS

Figure 3 (a): Modified Beacon Frame



Original Frame Body (Beacon Frame Fields)	Remaining bytes of B = ($g^y \bmod p$) with its length	Hash of (Complete Frame append p) with its length in bytes

Figure 3 (b): Modified Frame Body of Beacon Frame

ii. The STA sends probe request with the algorithm code, first 5-bytes of $A = g^x \bmod p$, remaining bytes of A, with length within frame body after original probe request fields and the complete frame hash with its length appended as in the case of beacon. This is shown in figure 4 below:

iii. The AP in response sends probe reply with the same $B = g^y \bmod p$, broken in two parts as in previous frames, algorithm code and again the complete frame hash.

iv. At this stage, both AP and STA will calculate $Z = g^{xy} \bmod p$

v. Now after the three initial frames exchange, the STA will send authentication request frame (802.11 Authentication). This frame instead of containing values like A or B contains hash of the value $Z = g^{xy} \bmod p$. By using this hash the AP will authenticate that the client is a valid client as only the valid party would calculate correct value of Z. This hash placed within the frame body after the original fields within its length before the complete hash and the length of the frame. In addition it also contains the complete hash of the frame for integrity.

vi. The AP receiving the Authentication Request computes the hash of the complete frame and compares it with the hash it receives in the request; if both are same then integrity is satisfied. It then computes hash of Z and compares it with the internal hash it receives in the probe request. If both are same this mean the authentication is satisfied and the STA is a valid client.

vii. The AP then sends the Authentication Reply by placing the total hash of the Z appended by p. This change (i.e. p appended in hash) is made so that the adversary cannot copy the hash from the Authentication Request Frame. Then the complete hash of the frame is computed and placed same as before.

viii. The STA receiving the frame response after computing the integrity by comparing the frame hash the STA checks the authenticity of the frame again by computing the hash of (Z add p) and comparing it with the received value. If same, the frame is valid otherwise frame is dropped. The successful authentication will guarantee the validity of AP.

ix. The STA in case of successful authentication of the Authentication Response sends Association Request. Here too the same mechanism as in previous frame is used but the internal hash (for auth) now has the hash value of Z append g. This is again for the same reason that the value in previous frame should not be copied by the adversary.

x. Now after receiving the Association Request the AP first checks the integrity then the authentication in the same way as for the previous frame and if both checks are successful, it sends the association response now with hash of (Z plus p plus g) in the internal hash. It obviously sends the hash of complete frame as well.

xi. The STA receiving the Association Reply computes & compares both the hashes and a successful association is established after authentication.

xii. The protocol also has three other management frames: de-authentication, disassociation and re-association frames. Here in our architecture, re-association is not required since if the client sends disassociation, it will also be de-authenticated and the authentication will start from the beginning. This is done in order to make the AP free of the burden of maintaining number of states when two or more STAs are associated with it.

xiii. The de-authentication frame will be sent by the client with the internal hash (for authentication) of ($g^{xy} \bmod p - p$). The complete frame hash will also be taken for integrity. The modified frame is shown in figure 10.

xiv. The last remaining frame type is the disassociation frame. This frame will contain the internal hash of ($g^{xy} \bmod p - g$). When this frame is sent, the de-authentication frame will also be sent automatically. So to re-associate the client will have to initiate the whole process (probe, authentication and association) again. The frame is shown in figure 11.

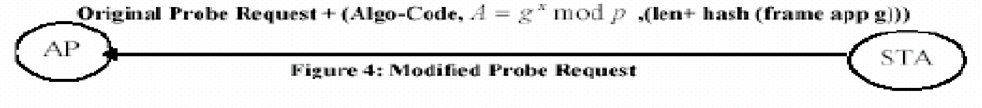

Original Probe Request + (Algo-Code, $A = g^x \bmod p$,(len+ hash (frame app g)))
AP
STA

Figure 4: Modified Probe Request

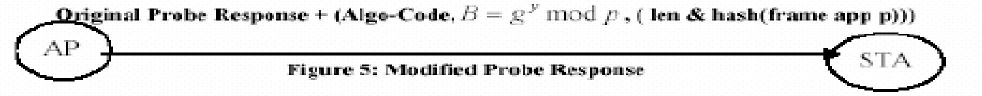

Original Probe Response + (Algo-Code, $B = g^y \bmod p$, (len & hash(frame app p)))
AP
STA

Figure 5: Modified Probe Response

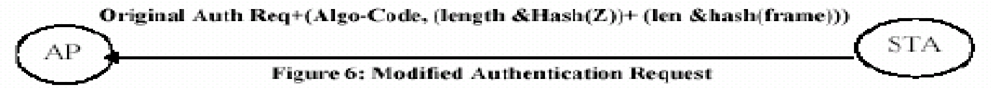

Original Auth Req+(Algo-Code, (length &Hash(Z))+ (len &hash(frame)))
AP
STA

Figure 6: Modified Authentication Request

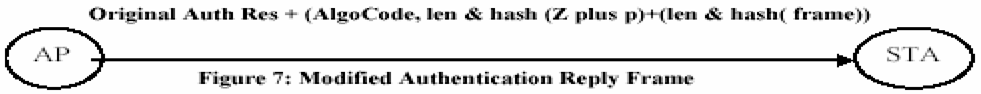

Original Auth Res + (AlgoCode, len & hash (Z plus p)+(len & hash(frame))
AP
STA

Figure 7: Modified Authentication Reply Frame

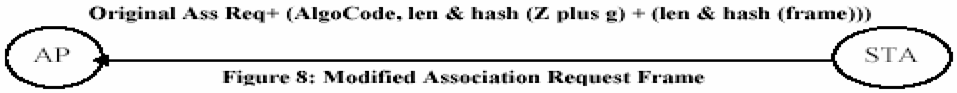

Original Ass Req+ (AlgoCode, len & hash (Z plus g) + (len & hash (frame)))
AP
STA

Figure 8: Modified Association Request Frame

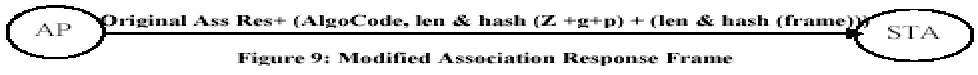

Original Ass Res+ (AlgoCode, len & hash (Z +g+p) + (len & hash (frame)))
AP
STA

Figure 9: Modified Association Response Frame

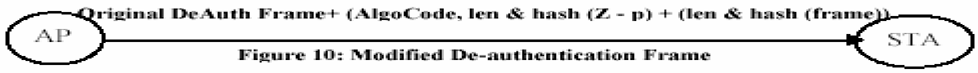

Original DeAuth Frame+ (AlgoCode, len & hash (Z - p) + (len & hash (frame))
AP
STA

Figure 10: Modified De-authentication Frame

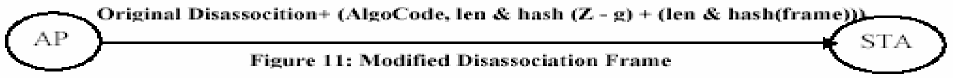

Original Disassocition+ (AlgoCode, len & hash (Z - g) + (len & hash(frame)))
AP
STA

Figure 11: Modified Disassociation Frame

After discussing the management frames architecture, now we show the simulation results and a brief discussion on these results.

4. Simulations Results

To verify our work, we have implemented the first two stages of RSNA establishment procedure [5] i.e.the exchange of all management frames. We used Java (sdk 1.4.2) to implement the simulation model. It was found that these attacks were successfully prevented by the valid parties and the poisonous frame is easily discarded by them.

5. Discussion

In this work we have presented a proposed solution for the vulnerable management frames. We have prevented several kinds of attacks by using our proposed solution by securing management frames. Our architecture also works in case of Malicious AP since the malicious AP has to authenticate itself by using some privately generated values (like g, p and Z) which it cannot generate, so the possibility of this kind of attack is negligible. In our architecture, we have checked frames for both authentication and integrity frame by frame in order to make the protocol as secure as possible and to prevent waste of exchanging malicious management frames. Since the disassociation and de-authentication frames are check for authenticity before processing so this DoS attack is not possible.

Our solution doesn't cater for the passive eavesdropping attack and it is not needed until we are sure that the adversary won't make any changes in the frames and even if it will, that should be detected at the receiving end. This is successfully achieved in our case so this attack is not harmful.

In this work work important thing is that by using hashing techniques which are one-way functions we have achieved our results without requiring the encryption/decryption mechanism. This mechanism would have been a very processing intensive solution if required. The other important thing is that even if the integrity violation is poisoned by the adversary by capturing the authentication and association frames (request and reply), modifying it and recalculating the frame hash, it cannot poison the internal hash authentication mechanism as hash of Z appended with different private values (p , g and p+g) are used for this purpose. So the adversary cannot calculate Z by any means. So the adversary cannot authenticate himself. The next section discusses the performance comparison with current standard.

6. Performance Analysis

In this section we show the comparison performance comparison between the current 802.11 protocols standard and our proposed protocol (management frames only). The following table shows the processing + execution time differences (on average) between the management frames of the current standard and our modified management frames. The processor used is Pentium IV 2.4 Ghz with 1 GB of RAM. Moreover, Java compiler with sdk 1.4.2 is involved in these results.

Type of Frame	Current Std.	Proposed Std.
Beacon Frame	< 1 msec	< 10 msecs
Probe Request	< 1 msec	< 10 msecs
Probe Response	< 5 msecs	< 5msecs
Auth. Request	< 1 msec	< 13 msecs
Auth. Response	< 5 msecs	< 10 msecs
Association Req	< 1 msec	< 1 msec
Association Res.	< 5 msecs	< 5 msecs
Deauthentication	< 1msec	< 1msec
Disassociation	< 1msec	< 1msec

Table: 2 Performance Comparisons

So by applying our architecture the time taken by each frame has been increased. This is understandable since we have applied authentication checks at each step and hash functions for integrity. If we see the total execution time then we get 21 millisecs approx. for current standard and 56 millisecs for our proposed solution. The result we are getting from out proposed work is far worthier than the time difference. Secondly security is gained at some cost and that is here in terms of execution time. So this is a trade-off.

7. Related Work:

Changua He [6] et al. studied the 802.11i wireless networking standard in various aspects. He covered the area of data confidentiality, mutual authentication, availability and integrity. They presented an improved architecture to make 802.11i DoS resistant. It discussed WEP and Wi-Fi Protected Access WPA mechanisms. But no solution proposed for vulnerable management frames.

Kjell J. Hole et. al. [8] in their work analyzed different security techniques and suggested ways to secure these kinds of networks. Jyh-Cheng Chen [9] discussed 802.1x authentication mechanism in the light of roaming users. William A. Arbaugh [10], in his work differentiated between the wired and wires security. Still problems were there for MAC layer security.

8. Conclusion & Future Work

By using a modified version of Diffie-Hellman method and securing management frames, the overall security of the 802.11 Wireless LAN has been improved. At each step authentication and integrity of the information is maintained and the expected results for many of the severe kinds of attacks have been achieved. For the 802.11 protocol standard the physical layer attacks like frequency jamming is still an issue. Also the availability is not the primary goal of this attack. These are still needed to be solved.

References:

[1] ANSI/IEEE Standard 802.11, 1999 Edition (R2003)

[2] IEEE Standard 802.11i™-2004

[3] http://www.wi-fi.org

[4]htttp://www.microsoft.com/whdc/device/network/802x/WPA.mspx

[5] Changhua He & Dr. John C Mitchell, ***"Security Analysis and Improvements for IEEE 802.11i"***, in Proceedings of Network and Distributed System Security Symposium Conference: DSS'05), February 2005, pages 90-110

[6] William Leh & Lee W. McKnight, ***"Wireless Internet Access: 3G vs. WiFi***?" by Massachusetts Institute of Technology - MIT Published, August 2002

[7] RFC 2631, ***Diffie-Hellman Key Agreement Method***

[8] Kjell J. Hole, Erlend Dyrnes and Per Thorsheim, ***"Securing Wi-Fi Networks".,*** IEEE Computer Society, 2005, pp. 28-34.

[9] Jyh-Cheng Chen, Ming-Chia Jiang, and Yi-Wen Liu, "***Wireless LAN Security and IEEE 802.11i",*** IEEE Wireless Communications Magazine, vol. 12, no. 1, February, 2005 , pp. 27-36.

[10] William A. Arbaugh, ***"Wireless Security is Different"*** IEEE Computer, vol. 36, August 2003, pp. 99 – 101.

[11] Kristin Lauter, "***The advantages of elliptic curve cryptography for Wireless Security"*** IEEE Wireless Communications Magazine, Vol.11, No.1, February, 2004, pp.62-67.