A DUAL LAYERED ENCRYPTION ALGORITHM FOR WIRELESS EQUIVALENT PRIVACY (WEP) ALGORITHM.

[1]Olufade, F. W. Onifade, [2]Adenike, O. Osofisan and [3]Chukwuzitere, U. OBODO

[1,2]Department of Computer Science, University of Ibadan, Ibadan, Oyo State, Nigeria.
[3]Department of Computer Science & Information Tech, Bowen University, Iwo, Osun State, Nigeria

ABSTRACT
Wired Equivalent Privacy (WEP) protocol was adopted to protect authorized users from unauthorized access and eavesdropping in the IEEE 802.11 wireless LANs (WLAN). However, it had been proven that the WEP protocol fails to provide data confidentiality and authentication [2, 14]. The WEP provides encrypted communication using an encryption key between the client station and Access point (AP). All client stations and APs on a network use the same key to encrypt and decrypt data. The key resides on both the client station and the AP [14]. The above rendered the WEP protocol naked to major forms of attack. Thus in this paper a keyed-message authentication code aimed at preventing an intruder from tampering with packets in transit is proposed, with a revised authentication scheme to avoid authentication spoofing and reduce replay attacks. We also employed private IV scheme alongside the use of day and session keys that counters several attacks.

The proposed WEP was tested with HTTPNet and TelnetNet with each client having their respective sessions. Finally, simulation methodology is presented, with the simulated results provided. Our studies show that the enhanced model provides better data confidentiality.

Keywords:
IEEE 802.11, Security, Wired Equivalent Privacy, Wireless LANs (WLAN), Integrity Check Value (ICV).

1.0 INTRODUCTION

The IEEE 802.11 standard [1] specifies Wired Equivalent Privacy (WEP), a wired LAN equivalent data confidentiality algorithm, to protect authorised users for security purposes. Unfortunately, the WEP protocol seriously fails to accomplish its security goals, and has been proved that prominent flaws exist [2]. Therefore, the growing popularity of the IEEE 802.11 products has been met with a growing concern for security reasons.

The WEP protocol employs the well known and believed secure RC4 stream cipher [9], a symmetrical cryptographic algorithm, with either a 40-bit or a 128- bit [1, 2]. In a symmetric key cryptographic algorithm, the same key is used in the encryption process as well as the decryption process. Due to this implementation of the RC4 cipher in the WEP protocol, many security flaws were discovered based on known drawbacks of the RC4 cipher thus, allowing eavesdropping and tampering with the wireless transmission [1, 9].

Network access control can be implemented using a Service Set Identifier (SSID) associated with an access point (AP) or a group of APs. To access the WLAN, client station must be configured with the correct SSID. Without knowledge of the AP's SSID, a mobile station cannot associate with it. This could be a simple way of securing an AP by not revealing the SSID to unauthorized stations. However, this minimal security is compromised if the AP is configured to broadcast its SSID, which might be a requirement where it is cumbersome or restrictive to configure the client station accessing the AP. When this broadcast feature is enabled, any station is allowed to scan the SSID and access the AP. In addition, since users typically have access to the configuration of client stations with the appropriate SSIDs, they are widely known and easily shared.

Wireless transmission have been know to be susceptible to interception more than the wired equivalent, thus, in an attempt to minimize this risk of security breach, the IEEE 802.11 standard specifies WEP for encryption and authentication [11]. The WEP provides encrypted communication using an encryption key between the client station and AP. All client stations and APs on a network use the same key to encrypt and decrypt data. The key resides on both the client station and the AP. Taking into cognizance this vulnerability, we propose two security enhancements to WEP algorithm to provide better data confidentiality and authentication.

2.0 OVERVIEW OF WEP SPECIFICATIONS & VULNERABILITIES

The WEP algorithm provides the 802.11 WLANs functionality of authentication and privacy services. The IEEE 802.11 claims the WEP algorithm to be reasonable strong to withstand brute-force attack to find the secret key. It is self-synchronizing, meaning that once the WEP option is turned on, it automatically encrypts each message frame traveling through the medium [11]. WEP is efficient by making it suitable to be implemented in either hardware or software on wireless devices, which typically have limited

Manuscript received May 5, 2008
Manuscript revised May 20, 2008

computational power when compared to its counterparts in a wired LAN. WEP is used in both authentication and data privacy. In authentication, encryption on the message (challenge text) is done by the station to prove that it has the right key to get into the network [10, 12]. In data privacy, encryption on message is done by the station to prevent eavesdropping by un-authorized stations. WEP employs symmetric key algorithm. A symmetric key algorithm is one where the same key is used in both encryption and decryption [14]. When the plain text (P) is encrypted with an encryption algorithm using the key K, cipher text (C) is obtained, i.e., $C=E_k(P)$, where $E_k(?)$ denotes the encryption algorithm/function. When the cipher text (C) is decrypted using the same key K, the original plaintext is obtained, i.e. $D_k(C) =P$: where $D_k(?)$ denotes the decryption algorithm/function. Obviously, the relationship $D_k[E_k(P)] = P$ follows. The key K is shared among the AP and all member stations of a Basic Service Set (BSS) [4, 6].

Unfortunately, the WEP has not well achieved confidentiality, access control, and data integrity. Although the WEP protocol attempts providing data privacy equivalent to that of a wired LAN, several vulnerabilities have been discovered in recent years [8, 14]. Attacks based on these vulnerabilities not only reveal the confidential data being transmitted, but also derive the secret key shared by the participating stations. Below we present some of the common attacks amongst the plethora of its vulnerabilities to establish the rational for this research.

2.1 MESSAGE TAMPERING

The WEP protocol adopts Cyclic Redundancy Check (CRC-32) to calculate a checksum integrity field (i.e. 32-bit Integrity Check Value- ICV) and is encrypted along the payload. This field is used after decryption to check the integrity of the message in transit. CRC-32 used in IEEE 802.11 is to ensure the integrity of the message by detecting random errors in messages, but not to ensure data security [3, 4]. Since the CRC checksum function is linear and stream ciphers such as RC4 are also linear, it is possible to tamper with the message in transit without detection through simple XOR methods [2]. Since the attacker knows the ciphertext C, the message is modified without the knowledge of the key stream and even without the knowledge of the message. The message modification process was presented in [2] as follows. Let P be the message to be modified and $C = RC4 (IV, K) \oplus (P, CRC (P))$ be the corresponding ciphertext. Let P' be the modified message and $\gamma = P \oplus P'$ be the modification made on P. let C' denote the modified ciphertext which is given to the AP, which will not find this message modification because CRC is a linear function, i.e., $CRC(P) \oplus CRC(\gamma) =CRC(P)$. We have

$C' = RC4 (IV, K) \oplus (P' CRC (P'))$ ------ (1) $= RC4 (IV, K) \oplus (P \oplus \gamma, CRC (P \oplus \gamma)) - (2)$

$= RC4 (IV, K) \oplus (P, CRC (P))) \oplus (\gamma, CRC(\gamma))$–– (3)

$= C \oplus (\gamma, CRC(\gamma))$ ---------- (4)

When the ciphertext is passed to the AP, the intruder hacks the cipher and modifies the message by XOR cipher (C) with $\gamma + C (\gamma)$ to the cipher. The modified text is sent to the AP, which decrypts the message and finds that the message is not modified. The main reason behind this successful modification of text without the WEP's knowledge is that during the encryption process, the secret key is not applied on the plaintext. CRC which is applied on the text is for data integrity and it cannot handle the message modification [1].

2.2 KEY MANAGEMENT

The 802.11 standard does not specify how the secret key is distributed to all the stations; it relies on an external system to do this. This practice seriously affects the security of the system that depends on a single key for its entire protocol to remain effective [12]. Thus, a constant secret key would increase chances of IV reuse and thereby key sequence reuse. Furthermore, compromise of a station could reveal the secret key, which would thwart the security of the entire network [2].

2.3 MESSAGE INJECTION

Based on known key sequences attack, it is possible to introduce an arbitrary number of messages into the WEP protected WLAN, thus circumventing access control since the same IV can be reused any number of times, and as long as the key sequence corresponding to a particular IV is correct, the AP cannot tell the difference between a message originating from an authenticated station or an intruder. An intruder needs only to encrypt random messages with the discovered key sequences, supply the IV along with it, and transmit the message to an accepting AP [2, 8]. When the intruder gets hold of the challenge text, the intruder can access the network traffic by simply injecting the message to the challenge text. If the intruder knows the challenge text and the cipher text, the intruder will get the key sequence according to ($RC4 (IV, K) = C \oplus P$). With the knowledge of the key sequence, the intruder can then use the key sequence to inject a message to the traffic and therefore cause increasing the traffic load, i.e., $C'= (P'CRC (P')) \oplus RC4 (IV, K)$.

2.4 AUTHENTICATION SPOOFING

A simple extension of plaintext attack leads to an authentication spoofing attack [1, 2]. During the authentication exchange, an intruder can eavesdrop and obtain a plaintext and a ciphertext pair. Using the pair, it becomes easy to obtain the key sequence. This exploit may be used to authenticate with an AP and open grounds for further attacks. An intruder may authenticate with an AP

without knowledge of the secret key assuming that the AP use the same pair of IV and the challenge text [2].

2.5 KEY SEQUENCE REUSE AND KNOWN PLAINTEXT ATTACK

The WEP provides data confidentiality using a stream cipher called RC4. A well known pitfall of stream ciphers is that encrypting two messages with the same key sequence can reveal information about both messages without any knowledge of the secret key [11, 14]. This could lead to a number of attacks (such as cryptanalysis of XOR plaintext strings, frequency analysis) unveiling the contents of each message individually [1]. To prevent key sequence reuse, the WEP recommends varying key sequences for payload so that the WEP uses a 24-bit IV [1], nearly guaranteeing that the same key sequence (caused from reuse of limited IVs and generally constant secret key) is being reused for multiple messages. Since IVs are public, key sequence reuse is easily detected through reuse of the IV (assuming the secret key may not have changed) thereby exposing the system to key sequence reuse attacks. Thus, a popular pitfall of stream ciphers servers is the compromise in the WEP recommendations. The secret key K always remain the same, but the change in the key sequence is due to the change in the IV every time. We observe that there exist chances for the IV to get reused since the length of the IV is 24 Bits. The key sequence generated by the WEP algorithm is the same if the IVs are the same. If the same key sequence is used for two plaintexts (P1 and P2), the cipher texts C1 and C2, respectively, are defined as follow.

$C1 = \{P1, ICV(P1)\} \oplus RC4(IV, K)$ ---- (5)

$C2 = \{P2, ICV(P2)\} \oplus RC4(IV, K)$ ---- (6)

In the above example, RC4 (IV, K) are reused. When the same IV is used for encrypting two different plaintexts, it is called a collision. Note that this collision concept is not that in channel access [2, 12].

$C1 \oplus C2 = P1 \oplus P2$ ---------- (7)

By the knowledge of C1, C2, and P1, P2 can be obtained as follows.

$P2 = (C1 \oplus C2) \oplus P1$ ---------- (8)

To find the key sequence reuse is easy and described as follows. The IVs are public and when they are sent with the ciphertexts, the intruder can obtain these IVs. Therefore, when the IVs are reused, the duplication of IVs can be easily spotted out. The main reason behind this attack is the length of the IV, which is 24 bits, and the maximum possible combinations of IVs can go up to 2^{24}. Experimental result depicts that the 1st collision occurs after transmitting 5000 packets which are few minutes after the data transmission. Considering the above, the attackers can get the duplicated IVs. However, the intruders can only obtain the messages using the same IV, under the condition that the triplet (P1, IV, C1) are known already [12].

Other forms of attack include: Man in the Middle Attack; Decryption Dictionary; Message Tampering e.t.c.

3.0 WEP SECURITY ENHANCEMENTS

The proposed enhancements attempt to rectify the vulnerabilities hitherto enumerated amongst others. We propose to enhance the WEP with Keyed Message Authentication Code and Enhanced Authentication (WEP-KMAC-EA) for data confidentiality, and to enhance the WEP with Private IV and Session/Day keys (WEP-PIV-SDK) for improved authentification process.

The proposed WEP-KMAC-EA adopts two enhancements of the WEP: Keyed Message Authentication Code and Enhanced Authentication.

3.1 KEYED MESSAGE AUTHENTICATION CODE

A WEP encrypted message can be subject to message tampering using attacks like man-in-the middle attacks and replay attacks. This is due to an un-keyed linear function (CRC32). CRC32 operation is linear, aimed at facilitating the data integrity however, cannot prevent the message from being tampered by an intruder. In other word, the generated message integrity check field depends only on the message and does not depend on the secret key. Borisov et al. [2] recommend the use of a KMAC to provide considerable strength. An intruder cannot tamper with the ICV of a message since he does not have the secret key used to generate it. Specifically, the WEP's ciphertext C is $(IV, P \oplus RC4(IV, K))$, whereas the KMAC uses $C = (K \oplus IV, P \oplus K \oplus RC4(IV, K))$.

3.2 ENHANCED AUTHENTICATION

The authentication method shown in RC4 authentication involves transmitting an unencrypted challenge text and an encrypted response of the same challenge text. This gives out a known plaintext-ciphertext pair to an intruder eavesdropping on the channel. Through known plaintext attacks, the intruder may spoof authentication and gain unauthorized access to the WEP. With this apparent vulnerability, we believed transmission of any plaintext-ciphertext pairs must be avoided. Thus, in the enhanced authentication mechanism: on request for authentication by a station, the AP can send a challenge nonce encrypted using the WEP with the shared key to the station: where a nonce is random number guaranteed not to be repeated (as much as possible) during the lifetime of the server generating it. The station decrypts it using the shared secret key, increments the nonce by 1, encrypts it with the WEP, and sends it back to the AP. The essence of the increment is to serve as acknowledgement to the AP that the station was in fact able to understand the challenge text through successful decryption. The authentication can be followed by transmissions of the session keys for subsequent transmissions.

3.3 WEP-PIV-SDK

The proposed WEP-PIV-SDK adopts two enhancements of the WEP: Private IV and Session keys.

3.3.1 PRIVATE IV

The reason why the IV is transmitted in the clear is because the 802.11 standard assumes that an intruder gains no useful information from its knowledge. It is clearly not true as shown in the various types of attacks discussed in earlier sections. The reason for using the IV is to produce key randomness, and the reason for transmitting the IV is to help the AP decrypt the information sent from the station. To strengthen the security, we propose to encrypt the IV by the WEP or the Day/Session key. This will disable an intruder's ability to easily map IVs to known key sequences. Specifically, the WEP's ciphertext C is (IV, P ⊕ RC4 (IV, K)), whereas the WEP with private IV uses C= (K_1⊕ IV, P ⊕ RC4 (IV, K)) where K_1 is the Day key or Session key.

3.3.2 SESSION/DAY KEYS (S/DKs)

The 802.11 does not specify the key management strategies, thus it relies on an external media to distribute the secret key to all stations, consequently, frequent changes of the secret key makes it management cumbersome. As a result, a constant secret key is used and it leads to reuse of key sequences culminating into serious vulnerabilities. Instead of using the secret key for generating the key sequence used to encrypt the payload, we used a day key which is the output of the randomized function on the secret key. Specifically, the WEP's ciphertext C is (IV, P ⊕ RC4 (IV, K)), whereas the WEP with day key uses

C = (K_D ⊕ IV, P ⊕ RC4 (IV, K_D)). ----- (9)

The AP generates the day key everyday. The AP can generate a day key per station and the traffic communicated between the station and the AP would be encrypted with a unique day key allocated for the station. Transmission of the day key can take place as soon as a station is authenticated via the original WEP encryption, therefore management of the day key will not pose an extra cost on WEP operation.

This provides a double-layered protection for the transmission of K_D in equation (9). Since the IV space is limited (24 bits in length), the above mechanism helps to change the key to achieve the requirement of supplying unique pairs of key and IV to the RC4 algorithm, and therefore, the problem of key sequence reuse can be largely avoided.

Using session keys alone defends a lot of attacks, but may lead to disconnected problem, in which when the station associated with the network gets temporally disconnected and wants to join the network after some time, it may be difficult to join the network since it does not have the key to get inside. The above point was taking care of in our model by issuing a separate key for the authentication process and a dedicated key for the information exchange between stations. The day key was used to generate the key sequence which in turn is used to encrypt the payload.

Once the station enters into the network using the shared key, it request for the authentication procedure. The AP sends the challenge text encrypted with the shared key. The station decrypts and increases the text by one, encrypts it back and sends it to the AP. When the text is sent, the IV is also encrypted and sent to the AP. This is makes the IV private so that the hackers can not gain any useful information. The AP gets the text, decrypts it and checks for the correctness of the text. Once the process is successful, a day key is generated for the station. The station uses the day key to process the information to other stations through the AP. This day key as the name implies is only valid for a day. Once the day key expires, a new day key is generated the station by the AP.

3.4 MODEL SIMULATION AND PARAMETERS DEFINITION

The HTTPNet/TelnetNet model uses the client-server approach for network communication. During communication, packets are to be secured from all forms of threat peculiar to the WLAN. The network to be simulated consists of four components: clients, cloud, server and packets. All components are characterized by one or more parameters. All parameters may be specified by user at the time of component creation with room for modification. For the simulation of the HTTPNet/TelnetNet model, the following parameters were used.

NUMBER OF CLIENTS

We consider a client-server network with various numbers of clients. In this experiment we employed a network of five, ten, fifteen, twenty-five, and fifty clients. The design of the operation of the simulator is that the network topology is robust, i.e. the user can specify their network topology as large as he wants within the constraints of the hardware.

PROPAGATION DELAY

The propagation delay is the time taken by the bit to propagate from one node to the next. It is a function of the distance between the routers and propagation speed and has nothing to do with packet length or transmission rate. The propagation speed depends on the physical medium of the link (e.g. multimode fiber, twisted-pair copper wire). That is the propagation delay is d/s, where d is the distance between the nodes and s is the propagation speed.

SERVICE TIME

The service time or rate of a server is the time taken to complete a service. In this case, the server responds to data requests from the user. Since clients' desire access to information in the server, packets containing user's credentials are sent to the server for processing. In this experiment, encryption and decryption of packets takes place at all nodes. Thus, the time taken to perform encryption and decryption of packets is referred to as the service rate of the server.

4.0 SIMULATION RESULT AND FINDINGS OF THE WORK

In the cause of the simulation, we varied parameters used to judge the performance of the proposed scheme. This enabled us to compare the rate of encryption and decryption of packets in the two models. These design parameters combined with the quality of the implementation determine the security of the sessions on the WLAN. The simulation was programmed using OMNet++.

Performance evaluation of the research work provides an atmosphere for testing the limitations in the existing WEP scheme with the provision of the proposed WEP (enhanced) scheme. Our evaluation was based on the following:

- Rate of encryption and decryption- evaluations include the time it takes for the packet or data moving in the network to be processed at the nodes (client, cloud and server), and
- Key reuse detection - key stream reuse is a major vulnerability of the WEP and if an intruder is able to detect it, the intruder can collect the respective "collision" packets for analysis.

4.1 RESULT DISCUSSIONS

In the implementation, clients are made to request authentication from the cloud using the WEP-KMAC-EA scheme. This experiment shows the rate of encryption and decryption that takes in the two models at the nodes (client, cloud, and server). This implementation improves WEP by changing the ICV from CRC-32 to HMAC-SHA-1 which is a keyed ICV. This implementation improves WEP by using day keys and session keys. Day keys are derived as a RC4 key stream using the secret key, concatenated with the day of the year, as the seed. Session keys are random bytes that are decided by the sending station and conveyed to the receiving station in encrypted form along with the data. Session keys are changed after a certain number of packets are processed.

We captured the time required to process varying amounts of data for various numbers of clients. Number of clients was varied, and the maximum number of clients used in this experiment is fifty. The time taken is in milliseconds. Each data was derived from an average of 10 trail test.

Two other important parameters used in the experiment, propagation delay and service time are also varied. The figures below shows the result for this experiment and it shows the rate of encryption and decryption for the different number of clients. We observe that the proposed schemes increase process time in some factors; however, data security was greatly enhanced.

Parameter definition I: cloud.propDelay = 0.1s; server.serviceTime = 0.1s

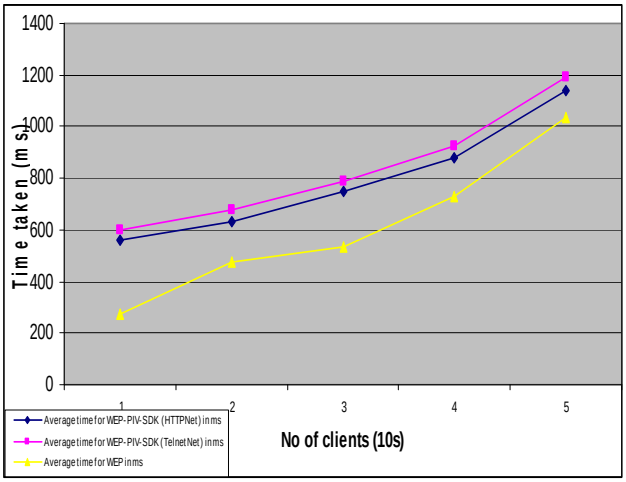

Figure 4.1: Rate of encryption and decryption for Parameter definition I

Figure 4.1 above was derived from the simulation results showing the average time taken in millisecond to complete encryption and decryption for various numbers of clients using the WEP-PIV-SDK technique. The propagation delay and the service time are kept at 0.1s. It can be observed form figure 4.1 that in the two models (HTTPNet and TelnetNet), the rate of encryption and decryption increases as the number of clients increases using the WEP-PIV-SDK scheme. Also, as the number of clients increases the rate of encryption and decryption also increases using the ordinary WEP scheme. Because of the implementation provided by WEP-PIV-SDK, the process time for encryption and decryption is increased when compared to the original WEP since the IV is transmitted encrypted in the proposed scheme and day keys and session keys are changed after a certain number of packets.

Parameter definition II; cloud.propDelay = 0.5s; server.serviceTime = 0.5s

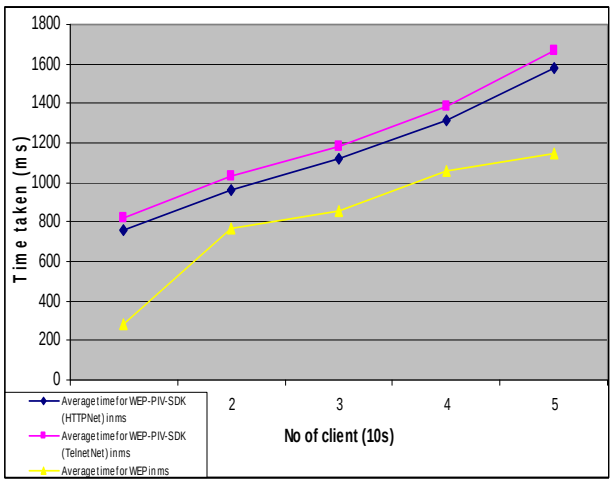

Figure 4.2: Rate of encryption and decryption for Parameter definition II

Figure 4.2 shows the rate of encryption and decryption for various number of clients when the propagation delay and service time are kept at 0.5s. The rate of encryption and decryption as observed from figure 4.2 increases in both models using the WEP-PIV-SDK when compared to the original WEP scheme. The increase in service time and propagation delay of the server and cloud respectively lead to more time needed to complete the data request which involved encryption and decryption. The implementation of the WEP-PIV-SDK scheme is still considered more efficient when compared to the original WEP even though both scheme increase process time when the service time and propagation delay are both increased.

Parameter definition III; cloud.propDelay = 1s; server.serviceTime = 1s

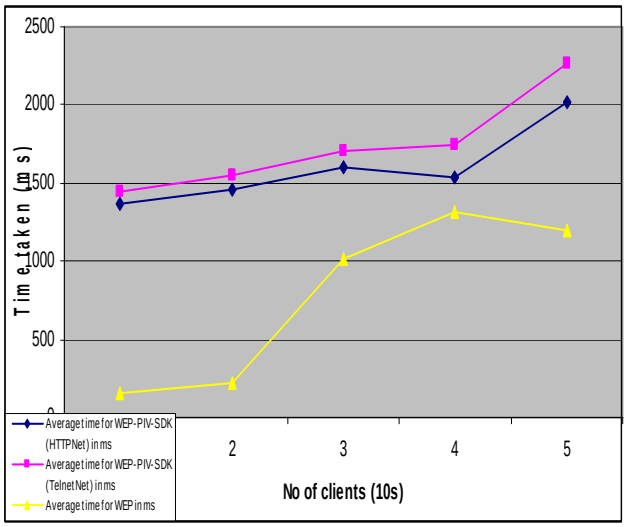

Figure 4.3: Rate of encryption and decryption for Parameter definition III

Figure 4.3 shows the rate of encryption and decryption when the service time and propagation delay are kept at 1s.

As observed there is still an increase in process time in both models using the WEP-PIV-SDK model when compared to original WEP. Again an increase in the service time and propagation delay has made the process time to increase. The WEP-PIV-SDK scheme thus makes the WLAN more secured than the original WEP since it is observed that the implementation of the proposed scheme is such that it is suitable for varieties of network models that implement the IEEE 802.11 WLAN.

In the next figure, (4.4), we used the aggregated simulated result to generate the performance of the schemes tested together. This is with a view to determine the level of reliability guaranteed to the transmitted packets across the cloud. Deliberate attempt was made to simulate the possible attack earlier stated and it was found out that, the new scheme present higher number of "clean packets" than the normal WEP. The indication is that, security of data is enhanced in the new scheme. Key reuse is major vulnerability of the WEP. The more the amount of the 'collision' bytes, the better is the chance of an intruder to compromise the security of the system. As expected, the WEP causes a number of collisions because of key stream reuse. Intuitively reasoning, this is because in the WEP, only the IV varies and the secret key is constant and therefore the key stream can be only in one of 2^{24} states for a given IV.

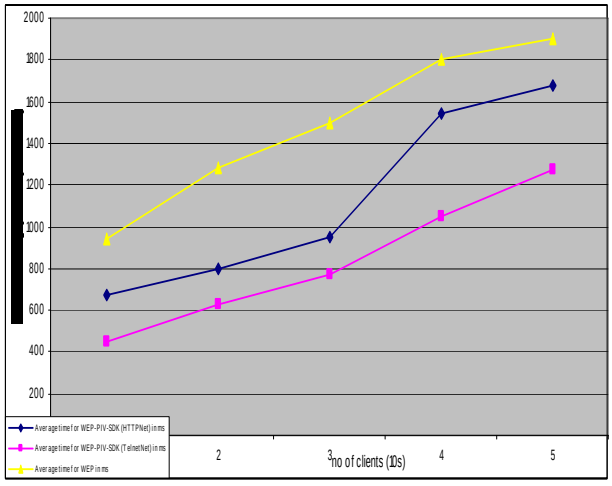

Figure 4.4: Packets transmission reliability versus the number of clients

However, with the use of day keys and session keys, the key stream is generated with a much better varying seed and it can be in one of 2^{64} states. This is why no key stream reuse is detected in the above experiments. Thus it can be concluded that WEP-PIV-SDK does a better job at generating random key streams than the standard WEP, and hence causes fewer collisions.

5.0 CONCLUSION

In this paper, we evaluated the security issues in the IEEE 802.11 WLANs, and proposed two enhancements for the

WEP. We conducted simulations/experiment on comparisons of these schemes with the original WEP scheme. The proposed WEP enhancements were justified using two network models: TelnetNet and HTTPNet. The propose enhancements provide better data confidentiality with some degree of computing cost as the tradeoff. The improved schemes overcome the weakness resulting from Key sequence reuse. They make use of not only the varying IV states, but also varying key states in order to supply a higher seed space resulting in lesser key stream reuse. With this new implementation, it is not easy to mount decryption dictionary attacks, since the total number of key streams to be discovered increases largely relative to the WEP, and the key streams used change from day to day for the same IV.

Key management is partially solved since the system is not easily compromised despite the secret key remaining unchanged for a long time. Message tampering is completely avoided from the use of keyed message authentication mechanism. Security against message injection is heightened since the discovery of a key stream is useful to the intruder only until the next session key change. If session key is refreshed frequently enough, depending on the network traffic, the vulnerability can be kept under check.

5.1 RECOMMENDATIONS & FUTURE RESEARCH

We realized in the experiments some drawbacks in the proposed algorithm. It is apparent that the keyed message authentication is a little computationally costly. More research needs to be done to determine a satisfactory trade off to find an easily computable integrity check value that cannot be easily tampered with. Alternate schemes may be explored that would improve the randomization factor of key stream. Authentication remains an area to be improved since the proposed authentication mechanism is vulnerable to replay and man-in-the-middle attacks.

REFERENCES

[1] IEEE 802.11 WG, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification, Standard, IEEE, Aug, 1999.

[2] N. Borisnov, I. Goldberg and D. Wagner, “Intercepting Mobile Communications: the Insecurity of 802.11”, The Seventh Annual International Conference on Mobile Computing and Networking (MOBICOM 2001).

[3] W. Arbaugh, N. Shankar, and Y. C. W. Justin Wan, “Your 802.11 Wireless Network has No Clothes”, Department of Computer Science, UMCP (March 2001).

[4] S. Fluhrer, L. Mantin, and A. Shamir, “Weakness in key scheduling algorithm of RC4” the Eight Annual Workshop on Selected Areas in Cryptography (August 2001).

[5] A. Stubblefield, J. Ioannidis, and A. Rubin, (2001) “Using the Fluhrer, Mantin and Shamir Attack to Break WEP”, AT&T Labs Technical Report (August 2001).

[6] D. Verton, “Your Wireless LAN Can Be Hacked-Flaws in 802.11 can leave data vulnerable” PCWorld.com.

[7] R. D. Vines, (2002) Wireless Security Essentials: Defending Mobile Systems from Data Piracy, (2002).

[8] C. Bandela, (2002) “Improving WEP Security in IEEE 802.11 Wireless Networks”, Georgia State University, Master Thesis, 2002.

[9] R. L. Rivest. The RC4 Encryption Algorithm. RSA Data Security, Inc. (March 1992, Proprietary).

[10] J. T, Yu (2004): “Performance Evaluation on a Linux bridge” Telecommunication System Management Conf. 2004, houseville, Kentucky.http:/facweb.cti.depaul.edu/jyu/publications/yu- linux7cu2004.pdf.

[11] Atheros Communications Inc. “802.11 Wireless LAN Performance”http://www.atheros.com/pt/atheros_range_whitepaper.pdf, 2003.

[12] B.A. Forouzan (2003): “ Local Area Networks” Mc Graw-Hill publishing, pp 23- 45, 2003.

[13] M. Leon, R. Aldeco & S. Meriro (2005): “ Performance Analysis of the confidentiality security service in IEE 802.11 using WEP” AES-CCM and ECC. In proc. of 2nd Int’l Conf. on Elect. Elect Engr. (ICEEE), XI conference on electrical engr. (CIE 2005).

[14] OMNet ++ Simulator: http://www.omnetpp.org (accessed on March, 2007).

Olufade .F. W. Onifade is a lecturer in the department of Computer Science, University of Ibadan, Oyo State, Nigeria. A recipient of the French government grant for the doctoral co-supervised thesis (University of Ibadan & Nancy 2 University, France). His research interests are in High Speed Networks, ATM Networks, Mobility Management in Mobile Ad hoc Networks, Video Streaming and Applications of Fuzzy Logic in system design and cognitive processes. He has published articles in International Journals of repute. Presently, his research is on Risk determination and management in Economic Intelligent processes towards strategic decision making.

Adenike .O. Osofisan is a reader and the current Head of department of Computer Science, University of Ibadan, Oyo State, Nigeria. She had her Master's of Science degree from Georgia-Tech, USA. She holds a doctorate degree from University of Ife, now Obafemi Awolowo University, Ile-Ife, Osun State, Nigeria. She is currently the president of the Computer Professionals Registration Council of Nigeria (CPN). Her articles are widely published in International journals and conferences proceedings. Her current research is in Data warehousing and Data mining applied to educational information.

Chukwuzitere, U. OBODO is a graduate of Bowen University, Iwo, Osun State from the department of Computer Science & Information Tech. His current research is the optimization of security provisions in wireless protocols.