Journal of Advances in Mathematical & Computational Sciences
An International Pan-African Multidisciplinary Journal of the SMART Research Group
International Centre for IT & Development (ICITD) USA
© Creative Research Publishers
Available online at https://www.isteams.net/ mathematics-computationaljournal.info
DOI: dx.doi.org/10.22624/AIMS/MATHS/V9N2P3
CrossREF Member Listing - https://www.crossref.org/06members/50go-live.html

An Enhanced Security System for Wireless Network Using Encryption Technique

Abdul-Salaam, Aminat Obakhume
Department of Computer Science
Federal College of Education (Special), Oyo
Oyo State, Nigeria
Phone Number: +2348068174689
Email: obakhume@gmail.com

ABSTRACT

The 802.11 standard defines the Wired Equivalent Privacy (WEP) and encapsulation of data frames. It is intended to provide data privacy to the level of a wired network. WEP suffered the threat of attacks from hackers owing to certain security shortcomings in the WEP protocol. Lately, many new protocols like WiFi Protected Access (WPA), WPA2, Robust Secure Network (RSN) and 802.11i have come into being, yet their implementation is fairly limited. Despite its shortcomings, one cannot underline the importance of WEP as it remains the most widely used system and this study chose to address certain security issues and proposed some modifications to make it more secure. In this paper, a three-level security data communication over an interface that includes secure wireless networking components like access points and user stations, spoof-proof encryption, authentication protocols and watchdog were discussed. The research work proposed a security system that employed a digitally signed authentication mechanism to achieve authentication, AES-CBC with PKCS padding and RSA to provide confidentiality and SHA-I hasting to provide integrity and eliminate the problem of the shared key was designed using Microsoft C# programming language. The enhanced system has the following advantages over WEP: the enhanced system uses SHA-I, which is stronger against brute force attacks, the RSA digital signature mechanism used which provides authentication and enhanced security and it does not have the problem of a shared key.

Keywords: Enhanced System, Encryption, Decryption, Wireless Network

Abdul-Salaam, A.O. (2021): An Enhanced Security System for Wireless Network Using Encryption Technique. Journal of Advances in Mathematical & Computational Science. Vol. 9, No. 2. Pp 33-50. Available online at www.isteams.net/mathematics-computationaljournal.
DOI: dx.doi.org/10.22624/AIMS/MATHS/V9N2P3

1. BACKGROUND TO THE STUDY

Using high-frequency radio waves instead of wires to communicate between nodes, Wireless Local Area Networks (WLAN) are convenient and flexible, providing the user with mobility and ease of use. In the last few years, we have seen incredible growth in the use and popularity of such networks. Today, the costs of equipment have dropped dramatically and "going wireless" is becoming mainstream. Wireless cards for laptops and wireless routers (access points) are in use everywhere ranging from large scale infrastructures to home networks (Machta, 2003).

At the same time, security concerns have been on the increase. A large amount of information travels across the air in the form of radio waves, and there is a need for the information to be kept secret and confidential, preserving its integrity. Wireless security can be broken into two parts: Authentication and encryption. Authentication mechanisms can be used to identify a wireless client to an access point and vice-versa, while encryption mechanisms ensure that it is not possible to intercept and decode data (Purandare, 2005).

Wired Equivalent Privacy (WEP) is the original native security mechanism for WLANs since the release in 1997 of the 802.11 specifications for WLAN by the Institute of Electrical and Electronics Engineers (IEEE). Wired Equivalent Privacy (WEP) is a deprecated algorithm to secure IEEE 802.11 wireless networks. WEP was intended to provide confidentiality comparable to that of a traditional wired network (Machta, 2003). However, WEP has been found to have several flaws, including cryptographic weaknesses.

A series of independent studies from various academic and commercial institutions had shown that an intruder equipped with the proper tools and a moderate amount of technical knowledge could gain unauthorized access to a WLAN even with WEP enabled (WiFi, 2003). WEP failed to achieve its goals in almost all the areas including authentication, access control, replay prevention, message modification detection, message privacy and key protection (Edney and Arabaugh, 2004). Serious security flaws like the presence of relatively short Initialization vectors (IVs) (Walker, 2000), keys that remain static, and a subtle vulnerability in the RC4 algorithm's usage in the WEP have made it relatively weak.

Lately, many new protocols like WiFi Protected Access (WPA), WPA2, Robust Secure Network (RSN), WEP-40, WEP-104 and 802.11i have come into being, yet their implementation is fairly limited. Despite its shortcomings, one cannot undermine the importance of WEP. WEP is still widely in use (Bandela, 2002). WEP is often the first security choice presented to users by router configuration tools even though it provides a level of security that deters only casual use, leaving the network vulnerable to any serious attempt at compromise (Bittau, et. al, 2008). This paper proposes a new encryption system that is free from the above drawbacks. The proposed system will ensure data authentication, confidentiality and integrity.

1.1 Statement of the Problem

Wireless transmissions have been known to be susceptible to interception more than the wired equivalent. This is because high-frequency radio waves instead of wires are used to communicate between nodes. As a result of this, risks are inherent in any wireless technology. The IEEE 802.11 standard specifies WEP for encryption and authentication. Unfortunately, the WEP protocol seriously fails to accomplish its security goals and has proved that prominent flaws exist. Serious security flaws like the presence of relatively short Initialization Vectors (IVs), keys that remain static, and a subtle vulnerability in the RC4 algorithm's usage in the WEP have made it relatively weak.

Gupta and Mohapatra (2008) observed that the following problems have been found out in WEP:

- WEP suffers from Passive Attacks & Active Attacks
- WEP suffers from Dictionary-building Attack
- No proper key Management Protocol in WEP
- Uses manually entered Shared key in the place of a randomly generated key

This study intends to design a security system that employs a digitally signed authentication mechanism to achieve authentication, Advanced Encryption Standard with Cipher-Block Chaining (AES-CBC) mode with Public-Key Cryptography Standards (PKCS) padding and RSA to provide confidentiality and SHA-I hashing to provide integrity and eliminate the problem of the shared key.

1.2 Purpose of the Study

This study intends to design an enhanced security system that employs the use of a new shared key that is capable of minimizing the information that an attacker can retrieve from the transmitted frames and minimizing the time available to him to launch an attack.

1.2 Aim and Objectives

Encryption technique covers a broad range from simple encryption that guards against accidental disclosure, to sophisticated methods that protect against all but the highly-trained criminal who has an in-depth knowledge of cryptanalysis and considerable deciphering equipment can break into an encrypted message and decrypt it without the knowledge of the sender and the receiver. It is not possible to have a hundred percent security; however, the efficiency of any security system can be measured in terms of the cost involved in breaking such security. This research work aims to design and implement an enhanced WEP security system that employs a digitally signed authentication mechanism to achieve authentication, AES-CBC mode with Public-Key Cryptography Standards (PKCS) padding and RSA to provide confidentiality and SHA-I hashing to provide integrity and eliminate the problem of a shared key. The objectives of the project include:

a. to design a wireless security system based on an enhanced WEP algorithm to overcome some known vulnerabilities and thus provide better data confidentiality and authentication.
b. to design an enhanced system that ensures the integrity

2. ANALYSIS OF THE EXISTING SYSTEM

WEP was ratified by IEEE in September 1999. Since its inception, WEP has been used by organizations or individuals as a wireless security protocol (Arash et al, 2009, Rehma et al, 2010 and Reddy et al, 2010). In 2005, a group of FBI personnel gave a demonstration on how they can use easily accessible tools to crack WEP encrypted system in less than 3 minutes (SmallNetBuilder, 2005). This demonstration is one of the most popular references to confirm the weakness of WEP. Subsequently, IEEE declared that WEP was obsolete and was replaced with WPA (Muhammad et al, 2012).

2.1 Enhancement of WEP

WEP uses the RC4 encryption algorithm which is known as a stream cipher. RC4 falls short of the standards set by cryptographers for a secure cipher in several ways and thus is not recommended for use in new applications. There are a few flaws in the way RC4 has been used in WEP and this has made WEP vulnerable to attackers (Borisov et al, 2001 & Fluhrer, et al 2001).

Similarly, researchers such as Hyten, et al (2006), Garcia (2006) & Hong and Lenhachneche (2003) observed that the RC4 algorithm in WEP has a subtle weakness that can be exploited to crack keys in WEP. Following this, Walker (2000) proposed an enhanced WEP where the Advanced Encryption Standard (AES) algorithm is used in place of the RC4 algorithm. Gupta and Mohapatra (2008) and Singh et al (2011) proposed an EWEP system that employed the use of SHA-1 hashing algorithm to generate message digest (i.e. hash code) which is then signed using the RSA algorithm. The Hash code is a function of all bits of the message, so it provides an error detection capability.

Any change of bit in the message results in a change in the hash code. So, by implementing the proposed protocol one can achieve authentication and integrity. The system employs the use of the AES encryption algorithm, in place of the RC4 algorithm to encrypt and decrypt a cipher text. The confidentiality of the system is achieved by using the AES algorithm. They used the following diagrams to depict the EWEP system:

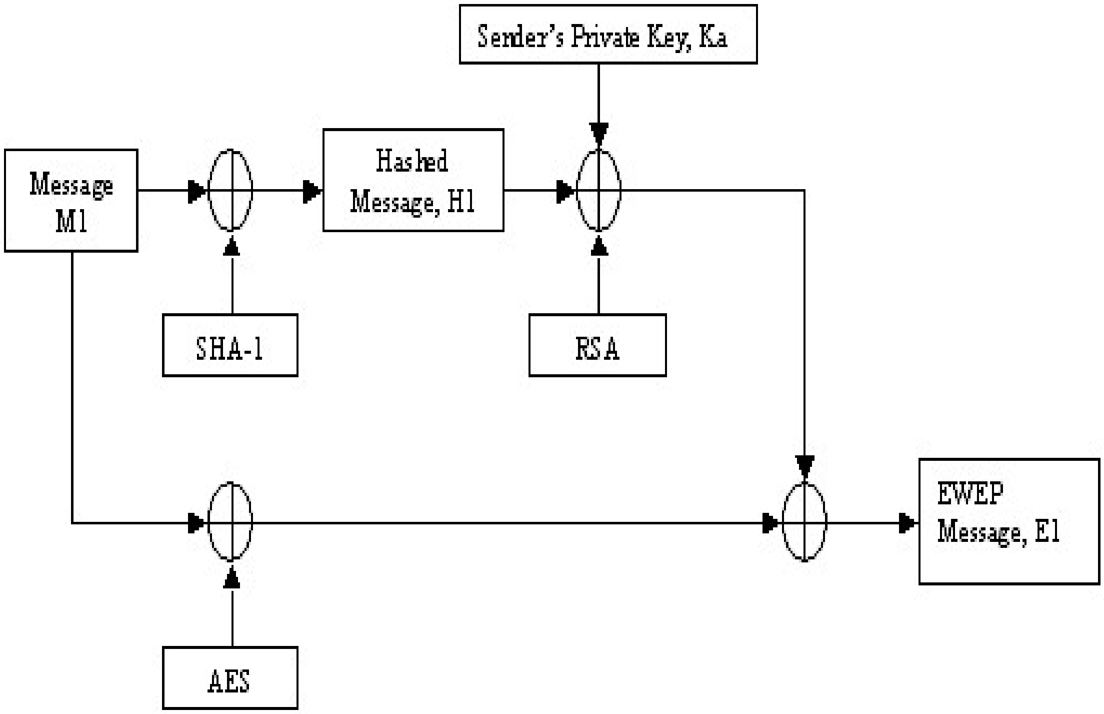


Fig. 1: Framework for Encryption Process for the Enhanced WEP
Source: (Gupta and Mohapatra, 2008) and (Singh et al 2011)

2.2 The Decryption Process

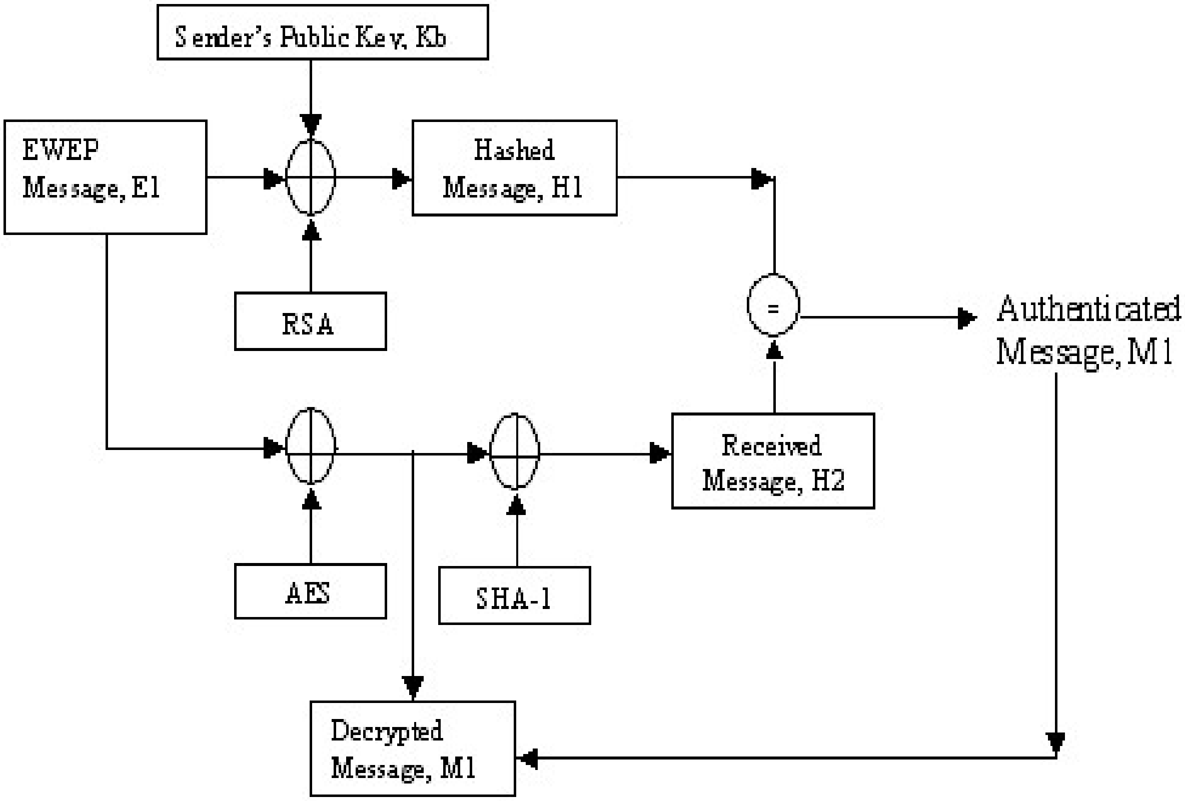


Fig. 3.2 Framework for Decryption Process for the Enhanced WEP
Source: Gupta and Mohapatra, 2008; Singh et al 2011

However, this system has its drawback, the system employs the use of the same key for encryption and signing. This makes the system prone to attack. If an attacker can convince a key holder to sign an unformatted encrypted message using the same key, then he/she gets the original message and this makes the system vulnerable.

This project intends to design an enhanced security system that employs the use of a new shared key that is capable of minimizing the information that an attacker can retrieve from the transmitted frames and minimizing the time available to him to launch an attack. The new system employs the use of a session key, that is generated using the day's date to encrypt the message.

2.3 Enhanced Encryption Process

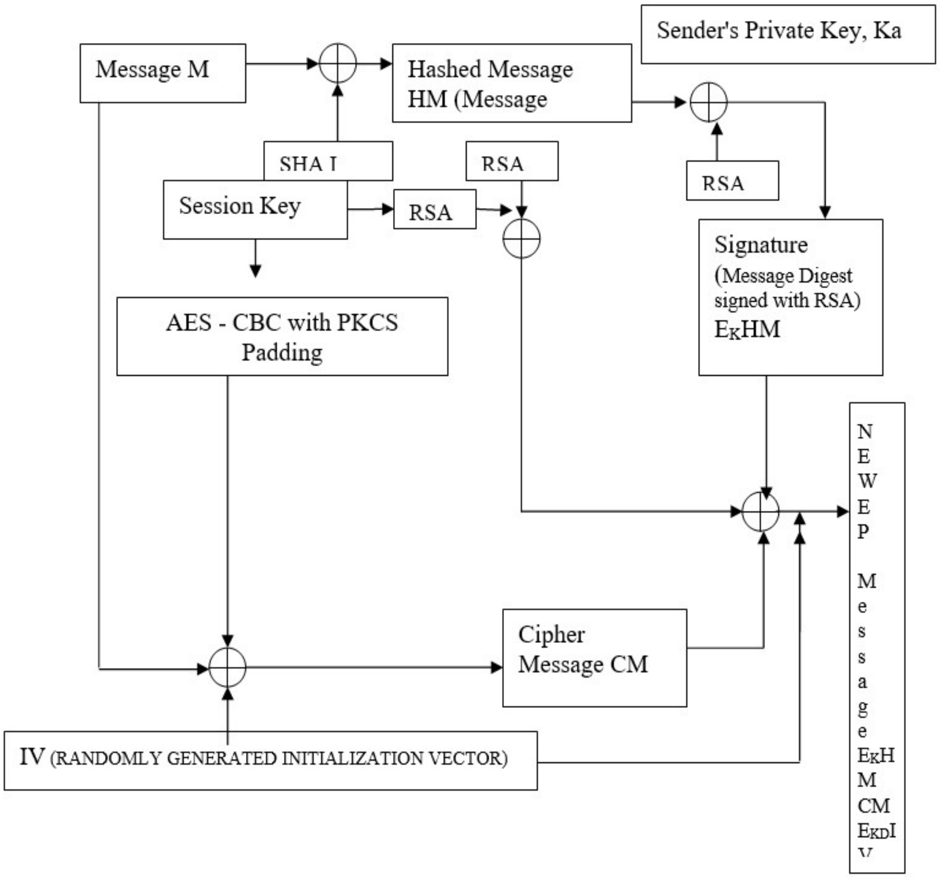

Message M
Hashed Message HM (Message
Sender's Private Key, Ka
SHA I
RSA
RSA
Session Key
RSA
Signature (Message Digest signed with RSA) E_KHM
AES - CBC with PKCS Padding
NEWEP Message EKHM CM EKDIV
Cipher Message CM
IV (RANDOMLY GENERATED INITIALIZATION VECTOR)

Fig. 3 Conceptual Framework for Encryption in the Proposed NEWEP System

1. **Message M:** The message M is the plaintext, that the sender intends to send.
2. **Hashed Message HM:** This is the Message Digest. It is generated by hashing the plaintext using SHA-1 hashing code i.e M $\oplus$ SHA-1.
3. **Signature E1:** This is generated by signing the Hashed Message using RSA Encryption algorithm with the Sender's Private Key, Ka .i.e. E_KHM = HM $\oplus$ RSA $\oplus$ Ka
4. **Cipher Message E2:** This is produced by encrypting the Plaintext M with a Block Encryption Algorithm AES-CBC with PKCS Padding, using a randomly generated IV and a Session Key i.e.CM = M $\oplus$ AES-CBC $\oplus$ IV $\oplus$ K_D
5. **NEWEP Message E1E2E_{KD}:** This is the cipher text, that is produced by the system. It is formed by concatenating the Signature E_KHM with the Cipher Message CM and the Encrypted Session Key with the IV i.e. E_KHM CME_{KD}IV

The New Enhanced Wired Equivalent Protocol (NEWEP)

The New Enhanced Wired Equivalent Protocol (NEWEP) embarked upon in this study is an improvement on the EWEP proposed by Gupta and Mohapatra (2008) and Singh et al (2011). The study employs the use of SHA-I hashing algorithm to generate a message digest (i.e. has code) which is then signed using the RAS algorithm using the sender's Public Key Kb as proposed by Gupta and Muhapatra (2008) and Singh et al (2011).

The improvement is seen in the replacement of the AES algorithm with a block cipher encryption algorithm, the Advanced Encryption Standard - Cipher Block Chaining Mode algorithm (AES-CBC) which features the combing of the plaintext blocks with the previous ciphertext blocks. Because the text to be encrypted is not an exact multiple, then the need to pad the message before encrypting by adding a padding string now arise. This study employs the use of the PKCS padding algorithm. It also uses a randomly generated Initialization Vector (IV) and a Session Key.

A copy of the plain text to be sent is hashed using the SHA-I hashing code to generate the Message Digest (HI). The hashed code (HM) is then signed using the Sender's Public key Kb, to generate the Signature (E_KHM). Another copy of the plain text is encrypted using the AES-CBC with PKCS padding using the session key and the randomly generated IV to produce the cipher text (CM). The session key is encrypted using the RC4 algorithm (SI). The signature E_KHM is concatenated with the cipher text CM and the session key to generate E_KHM CME$_{KD}$IV which is then sent as the NEWEP message to the recipient.

2.4 Enhanced Decryption Process

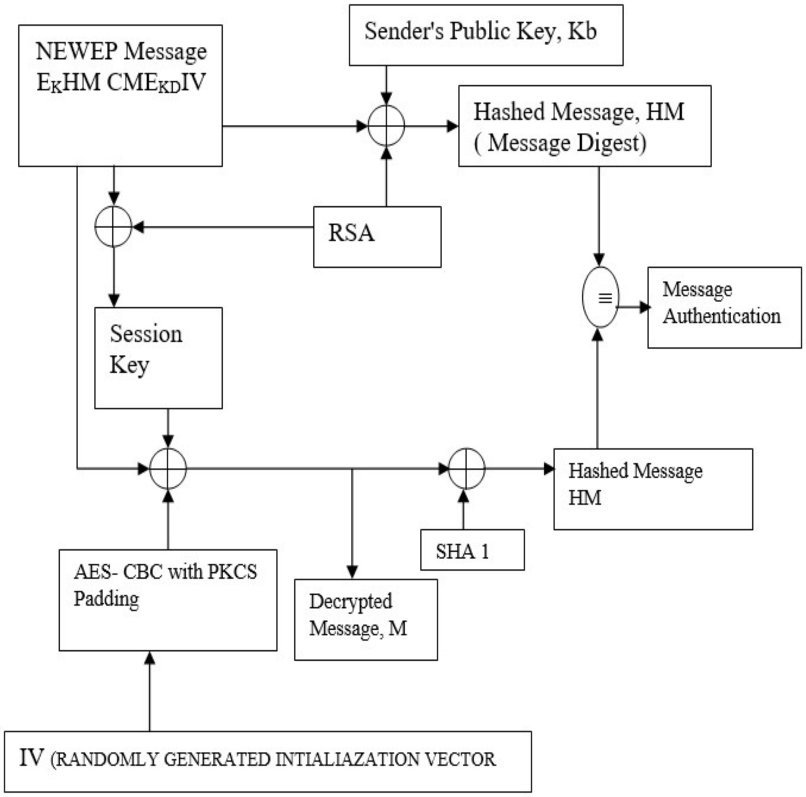
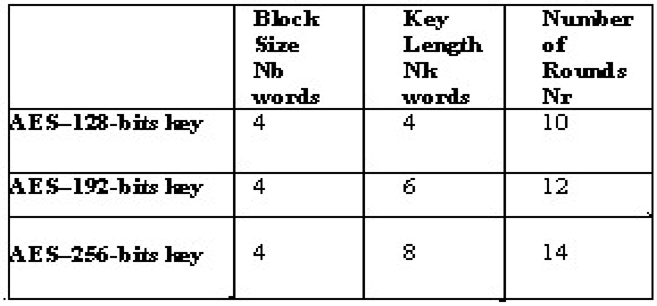


Fig. 4 Conceptual Framework for Decryption in the Proposed NEWEP System

2.5 Advanced Encryption Standard (AES)

The Advanced Encryption Standard (AES) specifies a FIPS-approved cryptographic algorithm that can be used to protect electronic data. The AES algorithm is a symmetric block cipher that can encrypt (encipher) and decrypt (decipher) information. Encryption converts data to an unintelligible form called ciphertext; decrypting the ciphertext converts the data back into its original form, called plaintext. The AES algorithm is capable of using cryptographic keys of 128, 192, and 256 bits to encrypt and decrypt data in blocks of 128 bits (Singh et al 2011). This standard specifies the **Rijndael** algorithm a symmetric block cipher that can process **data blocks** of **128 bits**, using cipher **keys** with lengths of **128**, **192**, and **256 bits**. Rijndael was designed to handle additional block sizes and key lengths, however, they are not adopted in this standard. Throughout the remainder of this standard, the algorithm specified herein will be referred to as "the AES algorithm." The algorithm may be used with the three different key lengths indicated above, and therefore these different "flavors" may be referred to as "AES-128", "AES- 192", and "AES-256".

Table 1: Number of Rounds



	Block Size Nb words	Key Length Nk words	Number of Rounds Nr
AES–128-bits key	4	4	10
AES–192-bits key	4	6	12
AES–256-bits key	4	8	14

AES operates on a 4x4 array of bytes (referred to as "state"). The algorithm consists of performing four different simple operations.

These operations are:

- Sub Bytes
- Shift Rows
- Mix Columns
- Add Round Key

1. **Sub Bytes** perform byte substitution which is derived from a multiplicative inverse of a finite field.
2. **Shift Rows** shifts elements from a given row by an offset equal to the row number.
3. **Mix Columns** step transforms each column using an invertible linear transformation.
4. **Add Round Key** step takes a 4x4 block from an expanded key (derived from the key), and XORs it with the "state".

AES is composed of four high-level steps. These are:

- Key Expansion
- Initial Round
- Rounds
- Final Round

The Key Expansion step is performed using a key schedule. The Initial Round consists only of an Add Round Key operation. The Rounds step consists of Sub Bytes, Shift Rows, Mix Columns, and an Add Round Key operation. The number of rounds in the Rounds step varies from 10 to 14 depending on the key size. Finally, the Final Round performs Sub Bytes, Shift Rows, and Add Round Key operations.

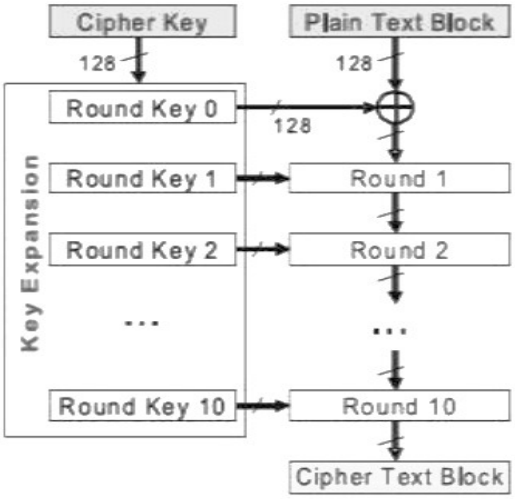


Fig. 5 Basic Concepts of AES Algorithm

The AES encryption and decryption processes for a 128-bit plain text block are shown in Fig. 6 and 7. The AES algorithm specifies three encryption modes: 128- bit, 192-bit, and 256-bit. Each cipher mode has a corresponding number of rounds Nr based on the key length of Nk words. The state block size, termed Nb, is constant for all encryption modes. This 128-bit block is termed the state. Each state is comprised of 4 words. A word is subsequently defined as 4 bytes. Table 1 shows the possible key/state block/round combinations (Thulasimani, 2010).

2.6 Encryption Process

The Encryption and decryption process consists of several different transformations applied consecutively over the data block bits, in a fixed number of iterations, called rounds. The number of rounds depends on the length of the key used for the encryption process. For the key length of 128 bits, the number of iteration required are10. (Nr = 10). As shown in Fig. 6, each of the first Nr-1 rounds consists of 4 transformations:
Sub Bytes(), Shift Rows(), Mix
Columns() & Add Round Key().

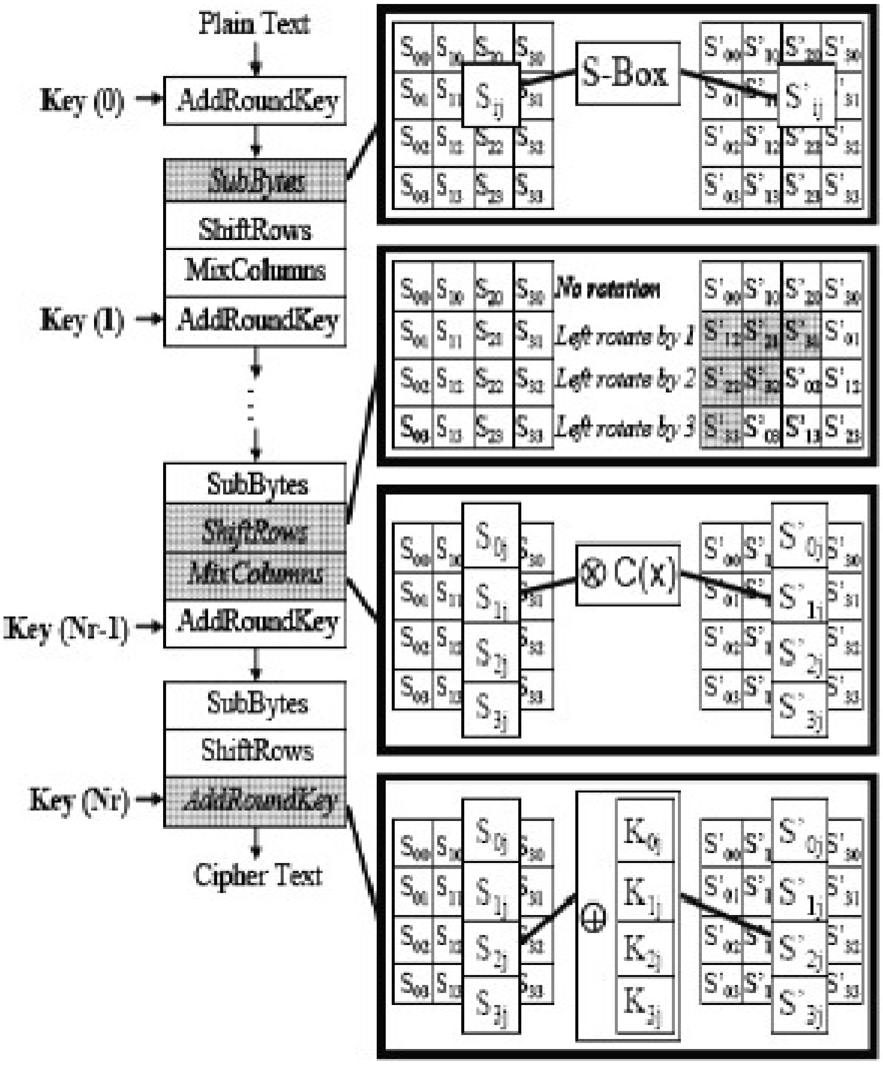


Fig 6: AES Encryption Process

1) **Sub Bytes Transformation**

It is a non-linear substitution of bytes that operates independently on each byte of the State using a substitution table (S box). This S-box which is invertible is constructed by first taking the multiplicative inverse in the finite field GF (28) with irreducible polynomial m(x) = x8 + x4+ x3 + x + 1. The element {00} is mapped to itself. Then affine transformation is applied (over GF (2)).

2) **Shift Rows Transformation**

Cyclically shifts the rows of the State over different offsets. The operation is almost the same in the decryption process except for the fact that the shifting offsets have different values.

3) **Mix Columns Transformation**

This transformation operates on the State column-bycolumn, treating each column as a four-term polynomial. The columns are considered as polynomials over GF (28) and multiplied by modulo x4 + 1 with a fixed polynomial a(x) = {03} x3+ {01} x2+ {01} x+ {02}.

4) **Add Round Key Transformation**

In this transformation, a Round Key is added to the State by a simple bitwise XOR operation. Each Round Key consists of Nb words from the key expansion. Those Nb words are each added into the columns of the State. Key Addition is the same for the decryption process.

5) **Key Expansion**

Each round key is a 4-word (128-bit) array generated as a product of the previous round key, a constant that changes each round, and a series of S-Box lookups for each 32-bit word of the key. The Key schedule Expansion generates a total of Nb (Nr + 1) words.

2.7 Decryption Process

For decryption, the same process occurs simply in reverse order – taking the 128-bit block of cipher text and converting it to plaintext by the application of the inverse of the four operations. Add Round Key is the same for both encryption and decryption. However, the three other functions have inverses used in the decryption process: Inverse Sub Bytes, Inverse Shift Rows, and Inverse Mix Columns. This process is the direct inverse of the Encryption process. All the transformations applied in the Encryption process are inversely applied to this process. Hence the last round values of both the data and key are first-round inputs for the Decryption process and follow in decreasing order.

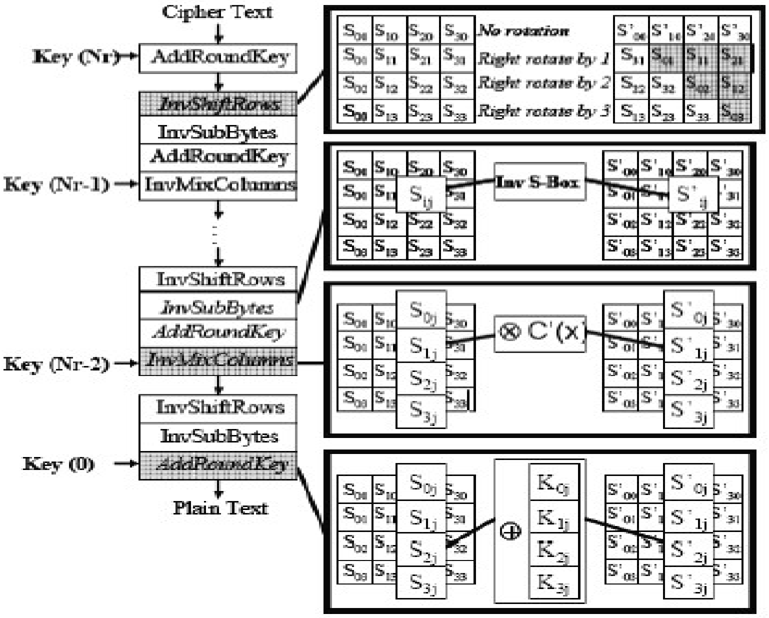


Fig 7: AES Decryption Process

2.8 Advanced Encryption Standard-Cipher Block Chaining Mode (AES-CBC)

The Cipher Block Chaining (CBC) mode is a confidentiality mode whose encryption process features the combining ("chaining") of the plaintext blocks with the previous ciphertext blocks. The input to the encryption processes of the CBC mode includes, in addition to the plaintext, a data block called the initialization vector (IV), denoted *IV*. The IV is used in an initial step in the encryption of a message and the corresponding decryption of the message. The IV need not be secret, but it must be unpredictable; The CBC mode is defined as follows:

CBC Encryption: Cipher text = $CIPH_K(P_1 \oplus IV_1, P_2 \oplus IV_2......, P_{128} \oplus IV_{128});$

CBC Decryption: Plain text = $CIPH^{-1}(C_1) \oplus IV......., (C_{128}) \oplus IV_{128})$

In CBC encryption, the first input block is formed by exclusive-ORing the first block of the plaintext with the IV. The forward cipher function is applied to the first input block, and the resulting output block is the first block of the ciphertext. This output block is also exclusive-ORed with the second plaintext data block to produce the second input block, and the forward cipher function is applied to produce the second output block. This output block, which is the second ciphertext block, is exclusive-ORed with the next plaintext block to form the next input block. Each successive plaintext block is exclusive-ORed with the previous output/ciphertext block to produce the new input block. The forward cipher function is applied to each input block to produce the ciphertext block.

In CBC decryption, the inverse cipher function is applied to the first ciphertext block, and the resulting output block is exclusive-ORed with the initialization vector to recover the first plaintext block. The inverse cipher function is also applied to the second ciphertext block, and the resulting output block is exclusive-ORed with the first ciphertext block to recover the second plaintext block. In general, to recover any plaintext block (except the first), the inverse cipher function is applied to the corresponding ciphertext block, and the resulting block is exclusive-ORed with the previous ciphertext block.

In CBC encryption, the input block to each forward cipher operation (except the first) depends on the result of the previous forward cipher operation, so the forward cipher operations cannot be performed in parallel. In CBC decryption, however, the input blocks for the inverse cipher function, i.e., the ciphertext blocks, are immediately available, so that multiple inverse cipher operations can be performed in parallel.

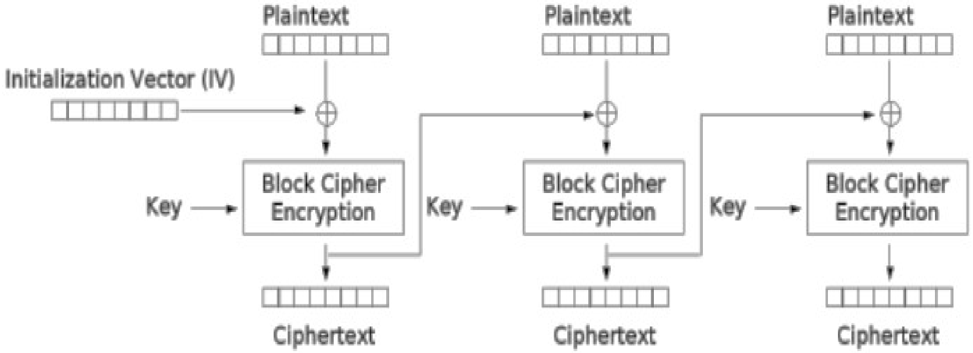


Fig 8: Cipher Block Chaining (CBC) Mode Encryption

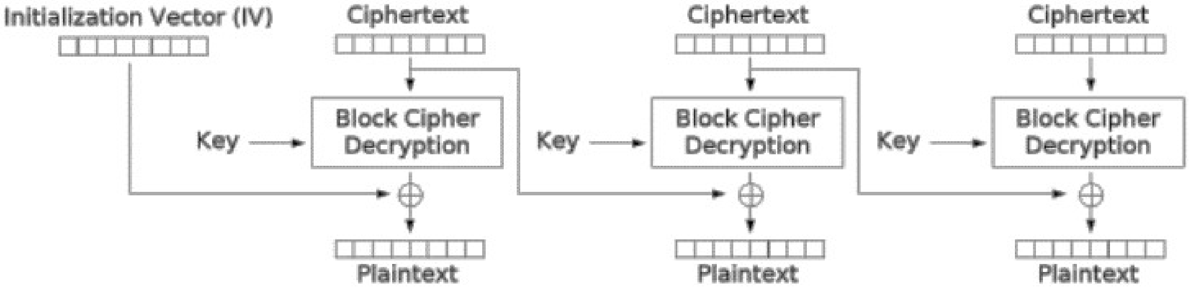


Fig 9: Cipher Block Chaining (CBC) Mode Decryption

2.9 Initialization Vector (IV)

An Initialization Vector (IV) is a block of bits that is used in an AES-CBC mode to randomize the encryption and hence to produce distinct ciphertexts even if the same plaintext is encrypted multiple times, without the need for a slower re-keying process. The IV is used in an initial step in the encryption of a message and the corresponding decryption of the message. In the AES-CBC mode, the IV must be unpredictable at the encryption line, in order to build a secured system.

The proposed NEWEP employs the use of CryptGenRandom to generate a randomized 16 bytes number which will be used as IV in the system. CryptGenRandom is a cryptographically secure pseudorandom number generator function that is included in Microsoft's Cryptographic Application Interface

2.10 Advantages of the proposed Protocol

The encryption mechanism used in WEP is a symmetric cipher called RC4 (Hannikainen et al, 2002) The Random Byte Generator used in this study is the CryptGenRandom that produces a long sequence of Random Bytes called Initialization Vector (IV). Then the IV is attached to the message unencrypted for the receiver to know which one to use. So the eavesdropper is unable to observe the IV.

The proposed Protocol has the following advantages over WEP:

1. The proposed protocol uses SHA-1, which is used to generate the Message Digest, SHA-I is stronger against brute force attacks (Mishra et al, 2003). This helps to ensure the confidentiality of the system
2. The RC4 digital signature mechanism is used which provides authentication and enhanced security. At the receiver's end, the Signature is Decrypted using the RC4 algorithm to produce the Message Digest HI. The Ciphertext CI is decrypted using AES-CBC with PKCS padding algorithm using the session key and an IV which were sent with the message. The Message produced is then Hashed using the SHA-I hashing algorithm to produce another Message Digest H2. Both HI and H2 are then compared, if they are the same then authentication is achieved. With this authentication is achieved, it also helps ensure the integrity and confidentiality of the system.
3. It does not have the problem of a shared key. The Sender's Public Key is used to sign while a randomly generated session key is used to encrypt the message, so with this, the problem to the shared key is eliminated and this help to ensure the integrity of the system.

3. SYSTEM IMPLEMENTATION

3.1 Software Requirement

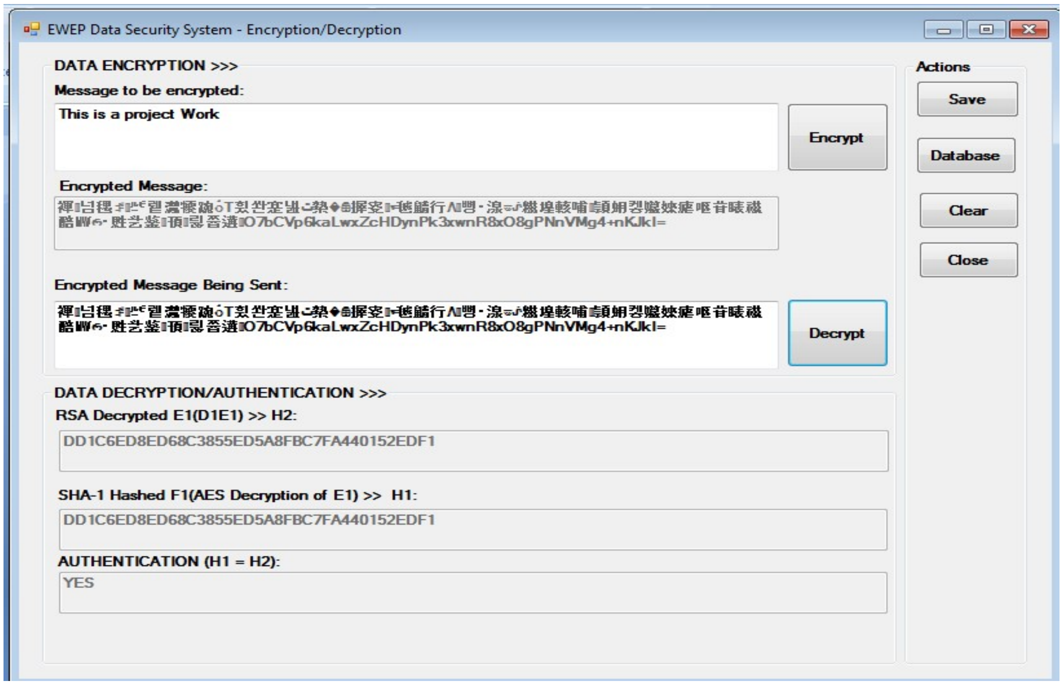
A computer requires but hardware and software to function effectively. The software requirements include:

1. 32- bit or 64-bit operating system
2. Visual C#
3. Microsoft Access

3.2 User Documentation

The application performs three basic operations, which are encryption, decryption and authentication. The program is loaded as follows:

1. Boot the system by switching on the CPU and the Monitor
2. After Booting, click on Program and then click on EWEP from the resulting sub-menu.
3. Wait and allow the application to load. After loading, the EWEP Data Security System - Encryption/Decryption screen or window is displayed. This menu comes with various commands. Some of which are:
   a. **Encrypt Button:** It is used to encrypt or encode the plain text into a cipher text.
   b. **Decrypt Button:** It is used to decrypt or decode the cipher text back into plain text by the recipient.
   c. **Save Button:** It is used to save the encrypted text before sending.
   d. **Database:** It is used to display the contents in the database.
   e. **Clear:** It is used to clear the contents entered in the message to be encrypted field.
   f. **Close:** It is used to close the opened window


EWEP Data Security System - Encryption/Decryption
DATA ENCRYPTION >>>
Encrypt
Decrypt
Actions
Save
Database
Clear
Close
DATA DECRYPTION/AUTHENTICATION >>>

Message to be encrypted:
This is a project Work

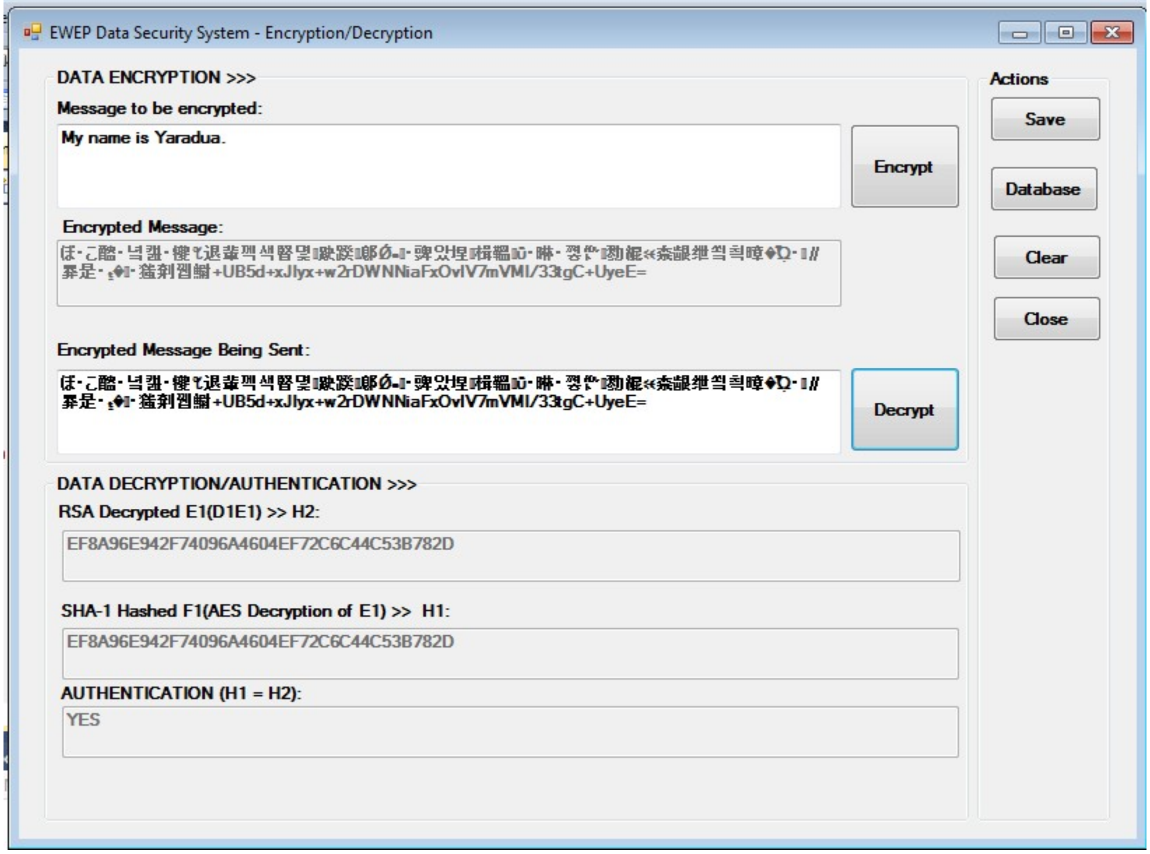
Encrypted Message:
[illegible]O7bCVp6kaLwxZcHDynPk3xwnR8xO8gPNnVMg4+nKJkl=

Encrypted Message Being Sent:
[illegible]O7bCVp6kaLwxZcHDynPk3xwnR8xO8gPNnVMg4+nKJkl=

RSA Decrypted E1(D1E1) >> H2:
DD1C6ED8ED68C3855ED5A8FBC7FA440152EDF1

SHA-1 Hashed F1(AES Decryption of E1) >> H1:
DD1C6ED8ED68C3855ED5A8FBC7FA440152EDF1

AUTHENTICATION (H1 = H2):
YES



DATA ENCRYPTION >>>

Message to be encrypted:

My name is Yaradua.

Encrypted Message:

ぼ·こ酷·녀캠·樓ℓ退蜚젝색똄몇▮歎騤▮郞Ø▪▮·魂았垾▮椙韜▮ü·晽·졍㑊▮勁䄇«叅䶳绁쐼칰暲◆Ꝺ·▮//
羿是·ȩ◆▮·篕剢졩鯏+UB5d+xJlyx+w2rDWNNiaFxOvlV7mVMI/33tgC+UyeE=

Encrypted Message Being Sent:

ぼ·こ酷·녀캠·樓ℓ退蜚젝색똄몇▮歎騤▮郞Ø▪▮·魂았垾▮椙韜▮ü·晽·졍㑊▮勁䄇«叅䶳绁쐼칰暲◆Ꝺ·▮//
羿是·ȩ◆▮·篕剢졩鯏+UB5d+xJlyx+w2rDWNNiaFxOvlV7mVMI/33tgC+UyeE=

DATA DECRYPTION/AUTHENTICATION >>>

RSA Decrypted E1(D1E1) >> H2:

EF8A96E942F74096A4604EF72C6C44C53B782D

SHA-1 Hashed F1(AES Decryption of E1) >> H1:

EF8A96E942F74096A4604EF72C6C44C53B782D

AUTHENTICATION (H1 = H2):

YES

Fig 11: EWEP Data Security System - Decryption

3.3 Simulation Methodology

In this research work, the simulation NEWEP was developed by the researcher, taking into consideration the proposed enhancement on WEP. While simulating, two entities were used, a sending station module and a receiving station module. The sending/receiving stations modules are assumed to engage in stream-based communication. The data is sent/received without enclosing any header information for simplicity. The medium of transmission is the local hard disk. The sending station encrypts packets of data and displays the ciphertext version of the plain text, while the receiving station decrypts packets of data and display the plain text. Encryption takes place per packet.

3.4 Performance Evaluation

The Performance Evaluation provides an atmosphere for testing the limitation in the existing WEP scheme with the provision of the proposed NEWEP scheme. The evaluation is based on the rate of encryption and decryption evaluations, which includes the times it takes to encrypt or decrypt packets of data. TheThe time required to process varying amounts of data is computed for the three pairs of station modules and tabulated. File sizes are in KB and the time taken is in milliseconds. Each data was derived from an average of 10 trial tests. Figures 11 and 12 show the results for evaluating the NEWEP system and it shows the total times for encryption and decryption for each pair of station modules.

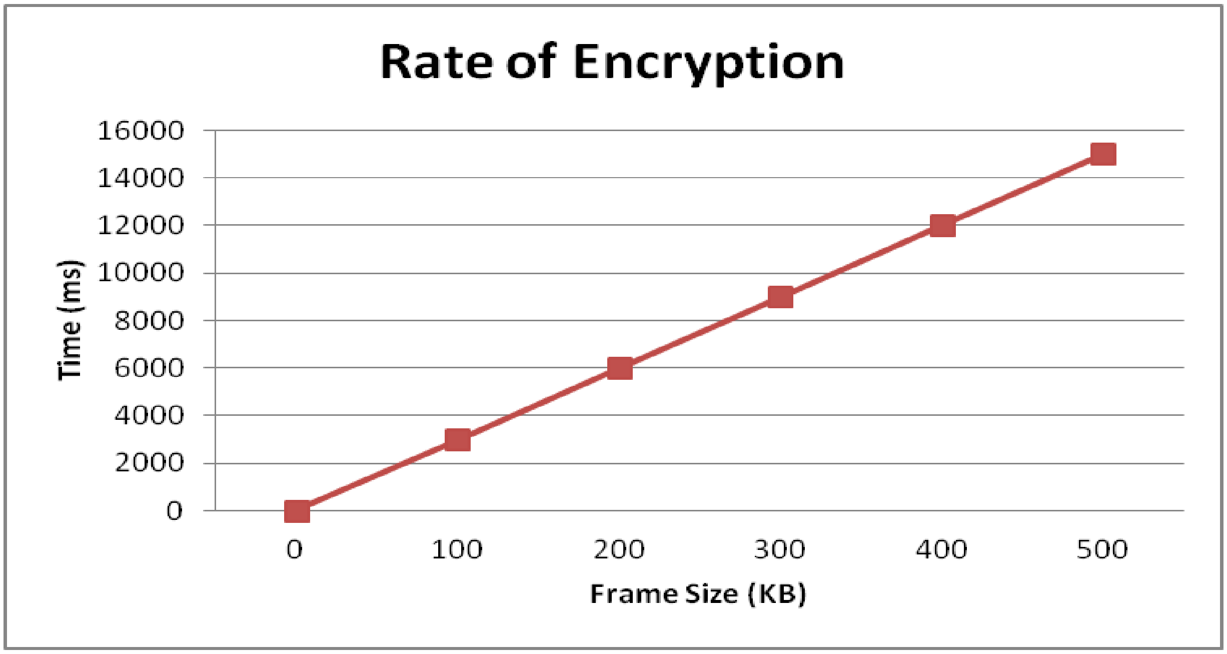


Fig. 11 Performance Evaluation: Rate of Encryption

Figure 11 above was derived from the simulation results showing the average time taken in milliseconds to complete encryption for various frame sizes using the NEWEP technique. The rate of encryption time evaluates the time it takes for the system to encrypt inputted text and display the ciphertext. The result shows that the rate of encryption increases when the frame size increases. The result from a research carried out by Xiao, et al (2006) shows that the rate of encryption also increases when the frame size increases in WEP. In the implementation of NEWEP, the time for encryption is increased when compared to the original WEP as observed in results from Xiao, et al (2006) as shown below:

Table 2: Performance Evaluation Table

Frame Size in KB	NEWEP Time of Encryption in ms	Xiao et al (2006) Time of Encryption in ms
0	0	0
100	3869	4010
200	6000	6471
300	9748	10000
400	11980	12123
500	15200	15491

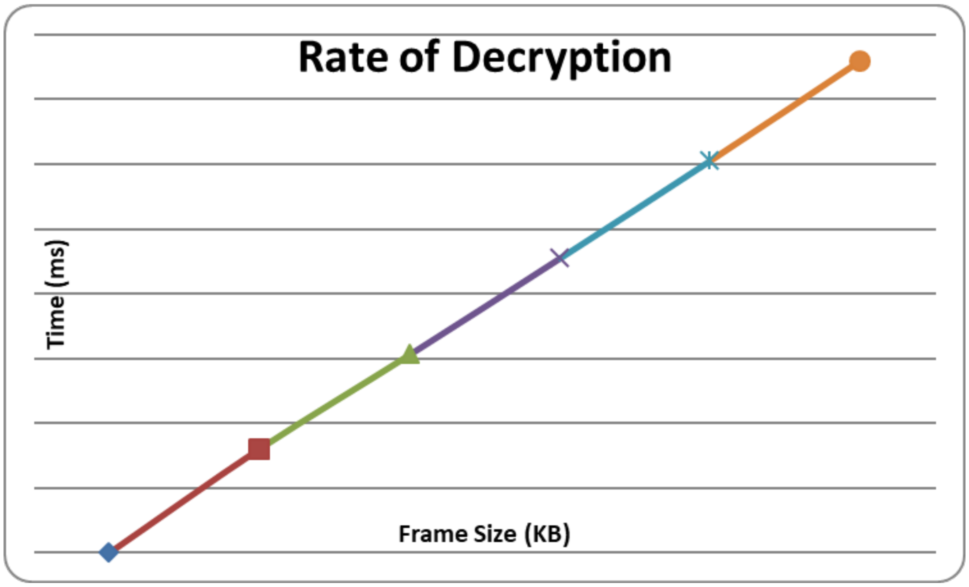


Fig. 12 Performance Evaluation: Rate of Decryption

Figure 12 above was derived from the simulation results showing the average time taken in milliseconds to complete decryption for various frame sizes using the NEWEP technique. The rate of encryption time evaluates the time it takes for the system to decrypt inputted text and display the ciphertext. The result shows that the rate of decryption increases when the frame size increases.

4. CONCLUSION

The existing WEP protocol is vulnerable to different kinds of cryptanalytic attacks. These stem from inappropriate usage of cryptography and not because of the key size. The possible drawback one can identify with this method is the computational overhead associated with generating and transmitting the session keys at the access point. This study proposed a NEW Enhanced WEP (NEWEP) which has so many advantages over WEP. The proposed protocol addresses a digitally signed authentication mechanism to achieve authentication, uses AES-CBC and RSA to provide confidentiality and SHA-I hashing to provide integrity.

5. RECOMMENDATIONS

During the simulation stage, some drawbacks in the proposed algorithm were observed. The keyed message authentication is a little computability 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. Alternative schemes may be explored that would improve the randomization factor of keystream. Authentication remains an area to be improved since the proposed authentication mechanism is vulnerable to replay and man-in-the-middle attacks.

REFERENCES

1. Arbaugh, W. A ., Shankar, N. & Wan, Y. J. (2001). Your 802.11 Wireless Network has no Clothes. In *IEEE International Conference on Wireless LANs and Home Networks*.
2. Arash, H. L. & Mir, M. S. D. (2009). A survey on wireless security protocols (WEP, WPA and WPA2/802.11i), *2nd IEEE International Conference on Computer Science and Information Technology* (ICCSIT), 48- 52.
3. Bandela, C. (2002). Improving WEP security in IEEE 802.11 wireless networks. Georgia State University, Master Thesis
4. Bittau, H. & Lackey, E. (2008). An IV Collision Avoidance Algorithm-Strengthening the WEP. [Paper Presentation]. The *2008 Int. Conference on Wireless Networks ICWN-08, Las Vegas, Nevada, USA.*
5. Borisov, N., Goldberg, I. & Wagner, D. (2001, July). Intercepting Mobile Communications: The insecurity of 802.11. In *MOBICOM 2001*, Rome, Italy.
6. Edney, J. & Arabaugh, W. A. (2004*). Real 802.11 security wi-fi protected access and 802.11i, 2004,* Pearsons Education Inc.
7. Fluhrer, M. I. & Shamir, A. (2001). Weaknesses in the Key-Scheduling Algorithm of RC4. In *Eighth Annual Workshop on Selected Areas in Cryptography*, Toronto, Canada.
8. Garcia, R. H.M. (2006, February 1-4). An analysis of wireless security. [Paper Presentation] The 16th South Central Conference.
9. Gupta & Mohapatra (2008) Wireless LAN Security with Enhanced Wired Equivalent Privacy (EWEP).
10. Hong, J & Lenhachhheche, K. (2003). *WEP: Protocol weaknesses and vulnerabilities. ECE 878: Computer and Network Security*. Research Project at Oregon State University.
11. IEEE 802.11 WG (1999) *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification.*
12. Machta, D. (2003) Securing WLAN from WEP to WPA.
13. Purandare, D. & Guha, R. (2005). An IV collision avoidance algorithm-strengthening the WEP. A paper presented at the *2005 International Conference on Wireless Networks ICWN-05, Las Vegas, Nevada, USA.*
14. Reddy, S. V., Sai-Ramani, K., Rijutha, K., Ali, S. M. & Reddy, C. P. (2010). Wireless hacking – A wifi hack by cracking WEP. *2nd Int Conference on Education Technology and Computer (ICETC),* VI-189-VI-193.
15. Rehman, S. U., Ullah, S & Ali, S. (2010). On enhancing the WEP security against brute force and compromised keys. *International Conference on Computer Information Systems and Industrial Management Applications (CISIM*), 250 - 254.
16. Singh, R., Rai, V. & Kumar, A. (2011). Enhancement for wired equivalent privacy. *International Journal of Information Technology and Knowledge Management, 4*(1), 91 - 95.
17. SmallNetBuilder (2005). *The Feds Can Own your WLAN Too*. Retrieved from http://www.smallnetbuilder.com/wireless/wireless-features/2425-thefedcanownyourwlantoo. retrieved on 15-4-2021.
18. Thulasimani, L. (2010). A single chip design and implementation of AES 128/192/256 encryption algorithms. *International Journal of Engineering, Science and Technology 2(*5), 1052 - 1059.
19. Walker, J. R. (2000). Unsafe at any Key Size; An Analysis of the WEP Encapsulation. IEEE Document 802.11-00/362, Oct 2020.
20. Xiao, Y., Bandela, C., Du, X., Pan, Y. & Dass, E. K. (2006). Security Mechanisms, Attack and Security Enhancements for the IEEE 802.11WLANs*. International Journal of Wireless and Mobile Computing, 1*(3&4), 276 - 288.