Pioneering Technique for Averting Congestion Occurrences by Unsystematic Padding

B. R. S. S. Raju
Assistant Professor
Department of CSE
Aditya Engineering College
Surampalem E.G.Dt.,

Dr. G. Naga Satish
Associate Professor
Department of CSE
Aditya Engineering College
Surampalem E.G.Dt.,

Abstract: **In wireless network the problem of intrusion is typically referred as congestion. This intentional nosiness with wireless transmissions can be used as a platform for mounting Denial-of-Service attacks in any type networks. Mostly congestion is addressed as external thread model. In this paper the problem of congestion is addressed in wireless network. In these type of attacks, the antagonist is active only for a short period of time selecting targeting messages with of high prominence. To diminish these type of attacks, we proposed a Robin signature schemes that prevent real-time packet classification by combining cryptographic primitive with physical-layer attributes.**

Keywords:- Congestion, attacks, antagonist

I. INTRODUCTION

Wireless Networks consists of large number of nodes interconnected to each other, are becoming a viable solution to many applications like domestic, commercial, and military applications. Wireless networks collects and sends the data from the areas even where ordinary networks are unreachable for various environmental and strategic reasons. The most promising concepts of wireless networking are auto-configurable and self-organizing. And it provides an adaptable and flexible wireless connectivity to the mobile users. The same notion can be used for different classes of wireless technologies such as wireless local area network (WLAN), wireless personal area network(WPAN), and wireless metropolitan network (WMAN). Due to the computation and power limitations wireless networks are more vulnerable to security threats.

Due to the computation and power limitations wireless networks are more vulnerable to security threats. Security does not come free, adding heavy security measures in terms heavy security measures in terms of computation power, limitation in memory poses and energy significant challenges in designing a light weight security solution against attacks on wireless networks.

Wireless networks rely on the uninterrupted availability of the wireless medium to interconnect participating nodes. However, the open nature of this medium leaves it vulnerable to multiple security threats. Anyone can eavesdrop on wireless transmissions, inject spurious messages, or jam legitimate ones. While eavesdropping and message injection can be prevented using cryptographic methods, congestion attacks are much harder to counter. They have been shown to actualize severe Denial-of-Service attacks against wireless networks. In the simplest form of jamming, the adversary interferes with the reception of messages by transmitting a continuous congestion signal, or several short congestion pulses. Typically, congestion attacks have been considered under an external threat model, in which the clamor is not part of the network. To prevent the congestion attacks we are proposing the following technique.

II. RELATED WORK

Although several studies have targeted congestion attacks but definition of congestion was unclear. An assumption is made that clamor transmits signal in wireless channel, so that channel is completely blocked and intended receiver may not be able to receive message. Therefore, clamor is an entity who is purposefully trying to interfere with transmission and reception of message across the wireless channel. Recently, several congestion strategies have been introduced.

III. EXISTING SYSTEM

Considering scenario where clamor jams the channel by blocking one or more nodes and block or corrupts the packets. This continuous congestion can be used as denial of-service attacks. The clamor controls the probability of congestion and transmission range to cause maximal damage to the network in terms of corrupted transmission links. The clamor action ceases when it is monitored detecting node and notification message is passed out of congestion region. To detect congestion attacks some statistics are used such as signal strength, carrier sensing time, packet delivery ratio.

IV. PROPOSED SYSTEM

The proposed system is used to prevent selective congestion attack in wireless network. This can be overcome by using Rabin signature algorithm for preventing congestion attacks. Before performing the Rabin signature we are generating random padding using Diffie Hellman key exchange algorithm. The generation of random padding has follows.

V. UNSYSTEMATIC PADDING GENERATION

1. Sender and verifier agree to use a prime number p and base g.
2. Sender chooses a secret integer $\boldsymbol{a}$, then sends verifier $A = g^a \bmod p$.
3. Verifier chooses a secret integer $\boldsymbol{b}$, then sends sender $B = g^b \bmod p$.
4. Sender computes $\boldsymbol{s} = B^a \bmod p$.
5. Verifier computes $\boldsymbol{s} = A^b \bmod p$.
6. Now the sender and the receiver share same random padding.

After generation of random padding the sender generate signature by using Rabin signature algorithm. The signature generation is as follows.

VI. KEY GENERATION

1. The signer S chooses primes p,q each of size approximately k/2 bits, and computes the product n=p*q.
2. S then chooses a random number b in {1, 2……n}.
3. The public key is (n, b).
4. The private key is (p, q).

VII. ENCRYPTION AND DECRYPTION OF MESSAGE

The sender will encrypt the message using cryptography technique. In this paper the DES algorithm is using for message encryption and decryption purpose. The procedure of Des algorithm as follows:

DES (and most of the other major symmetric ciphers) is based on a cipher known as the **Feistel block cipher**. Each round contains bit-shuffling, non-linear substitutions (S-boxes) and exclusive OR operations. Most symmetric encryption schemes today are based on this structure (known as a **feistel network**). DES expects two inputs - the plaintext to be encrypted and the secret key. The manner in which the plaintext is accepted, and the key arrangement used for encryption and decryption, both determine the type of cipher it is. DES is therefore a symmetric, 64 bit **block cipher** as it uses the same key for both encryption and decryption and only operates on 64 bit blocks of data at a time5 (be they plaintext or cipher text). The key size used is 56 bits, however a 64 bit (or eight-byte) key is actually input. The least significant bit of each byte is either used for parity (odd for DES) or set arbitrarily and does not increase the security in any way. All blocks are numbered from left to right which makes the eight bit of each byte the parity bit. Once a plain-text message is received to be encrypted, it is arranged into 64 bit blocks required for input. If the number of bits in the message is not evenly divisible by 64, then the last block will be padded. Multiple permutations and substitutions are incorporated throughout in order to increase the difficulty of performing a cryptanalysis on the cipher. It is generally accepted that the initial and final permutations offer little or no contribution to the security of DES and in fact some software implementations omit them (although strictly speaking these are not DES as they do not adhere to the standard). The sequence of events that occur during an encryption operation. DES performs an initial permutation on the entire 64 bit block of data. It is then split into 2, 32 bit sub-blocks, L_i and R_i which are then passed into what is known as a **round** of which there are 16 (the subscript i in L_i and R_i indicates the current round). Each of the rounds is identical and the effect of increasing their number is twofold - the algorithms security is increased and its temporal efficiency decreased. Clearly these are two conflicting outcomes and a compromise must be made. For DES the number chosen was 16, probably to guarantee

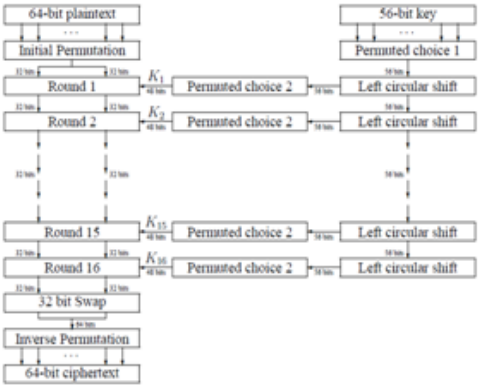

Fig.1:Process of Algorithm

The elimination of correlation between the cipher text and either the plain-text. At the end of the 16th round, the 32 bit L_i and R_i output quantities are swapped to create what is known as the **pre-output**. This [R16, L16] concatenation is permuted using a function which is the exact inverse of the initial permutation. The output of this final permutation is the 64 bit cipher text. So in total the processing of the plaintext proceeds in three phases as can be seen from the left hand side of figure 2.2:

1. Initial permutation (**IP**) rearranging the bits to form the "permuted input".
2. Followed by 16 iterations of the same function (substitution and permutation). The output of the last iteration consists of 64 bits which is a function of the Plain-text & key. The left and right halves are swapped to produce the preoutput.
3. Finally, the preoutput is passed through a permutation (**IP**−1) which is simply the inverse of the initial permutation (**IP**). The output of**IP**−1 is the 64-bit cipher text.

The inputs to each round consist of the L_i, R_i pair and a 48 bit **Subkey** which is a shifted and contracted version of the original 56 bit key.

• Initially the key is passed through a permutation function.
• For each of the 16 iterations, a Subkey ($\mathbf{K}_i$) is produced by a combination of a left circular shift and a permutation (**PC**2) which is the same for each iteration. The resulting sub key is different for each iteration because of repeated shifts.

A. Details of Individual Rounds

The main operations on the data are encompassed into what is referred to as the **cipher function** and is labeled ***F***. This function accepts two different length inputs of 32 bits and 48 bits and outputs a single 32 bit number. Both the data and key are operated on in parallel. The operations are quite different. The 56 bit key is split into two 28 bit halves Ci and Di (C and D being chosen so as not to be confused with L and R). The value of the key used in any round is simply a left cyclic shift and a permuted contraction of that used in the previous round. Mathematically, this can be written as:

$$C_i = L_{csi}\ (Ci-1),\ Di = L_{csi}\ (Di-1).$$

$$K_i = PC_2\ (C_i, D_i).$$

Where Lcsi is the left cyclic shift for round i, C_i and D_i are the outputs after the shifts, PC2 (.) is a function which permutes and compresses a 56 bit number into a 48 bit number and Ki is the actual key used in round i. The common formulas used to describe the relationships between the input to one round and its output is:

$$L_i = R_{i-1}$$

$$R_i = L_{i-1}\ \boldsymbol{F}\ (R_{i-1}, K_i)$$

Where L and R have their usual meaning and ***F*** (.) is the cipher function. This function ***F*** is the main part of every round and consists of four separate stages:

1. The E-box expansion permutation - here the 32-bit input data from Ri−1 is expanded and permuted to give the 48 bits necessary for combination with the

48 bit key. The E-box expansion permutation delivers a larger output by splitting its input into 8, 4-bit blocks and copying every first and fourth bit in each block into the output in a defined manner. The security offered by this.

2. The bit by bit addition modulo 2 (or exclusive OR) of the E-box output and 48 bit sub key K_i.

3. The S-box substitution - this is a highly important substitution which accepts a 48-bit input and outputs a 32-bit number. The S-boxes are the only non-linear operation in DES and are therefore the most important part of its security. They were very carefully designed although the conditions they were designed under have been under intense scrutiny since DES was released. The input to the S-boxes is 48 bits long arranged into 8, 6 bit blocks (b1, b2. . . b6). There are 8 S-boxes (S1, S2. S8) each of which accepts one of the 6 bit blocks. The output of each S-box is a four bit number. Each of the S-boxes can be thought of as a 4 × 16 matrix. Each cell of the matrix is identified by a Coordinate pair (i, j), where 0 <i < 3 and 0 < j <15. The value of i is taken as the decimal representation of the first and last bits of the input to each S-box, i.e. Dec(b1b6) = i and the value of j is take from the decimal representation of the inner four bits that remain, i.e. Dec(b2b3b4b5) = j. Each cell within the S-box matrices contains a 4-bit number which is output once that particular cell is selected by the input.

4. The P-box permutation - This simply permutes the output of the S-box without changing the size of the data. It is simply a permutation and nothing else. It has a one to one mapping of its input to its output giving a 32 bit output from a 32 bit input.

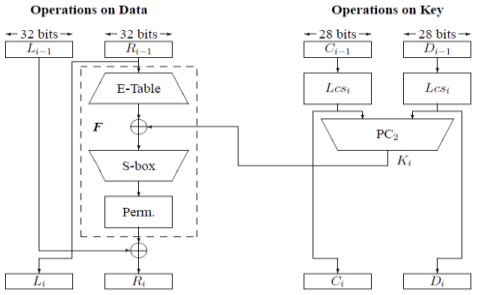

Fig 2: Operation on Data and key

B. Modes of Operation

The DES algorithm is a basic building block for providing data security. To apply DES in a variety of applications, five modes of operation have been defined which cover virtually all variation of use of the algorithm.

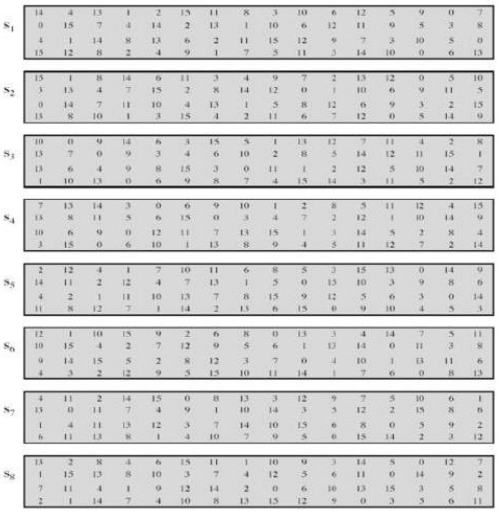

Fig 3: Modes of Operation

C. DES Decryption:

The decryption process with DES is essentially the same as the encryption process and is as follows:

- Use the cipher-text as the input to the DES algorithm but use the keys K_i in reverse order. That is, use K_{16} on the first iteration, K_{15} on the second until K_1 which is used on the 16th and last iteration.

VIII. SIGNING

1. To sign a Message m the signer S picks random padding U and calculates H (m, U).
2. S then solves x(x+b) =H (m, U) mod n.
3. If there is no solution s picks a new pad U and tries again. If H is truly random the expected number of tries is 4.
4. The Signature on m is the pair of (U, x).

A. Verification:

1. Given a message *m* and a signature *(U, x)* send to the verifier.
2. The verifier *V* calculates *x(x+b)* and *H(mU)* and verifies that they are equal then retrieve the packet . If not equal discard packets sended by sender to the verifier. This way we can prevent the selective jamming attack.

CONCLUSION

Congestion can also be arise because of various different reasons like it can be intentionally created by attackers which lead to denial of service attack or it can be unintentionally created on network due to congestion. In this paper we proposed a Robin signature schemes that prevent real-time packet classification by combining cryptographic primitive with physical-layer attributes. It improves the performance and reliability of wireless networks. We got various types of parameters we conclude that our system gives us better results.

REFERENCES

[1] T. X. Brown, J. E. James and A. Sethi. Jamming and sensing of encrypted wireless ad hoc networks. In Proceedings of MobiHoc, pages 120–130, 2006.
[2] M. Wilhelm, I. Martinovic, J. Schmitt, and V. Lenders. Reactive jamming in wireless networks: How realistic is the threat? In Proceedings of WiSec, 2011.
[3] W. Xu, W. Trappe, Y. Zhang, and T.Wood. The feasibility of launching and detecting jamming attacks in wireless networks. In Proceedings of MobiHoc, pages 46–57, 2005.
[4] IEEE. IEEE 802.11 Standard. http://standards.ieee.org/getieee802/download/802.11-2007.pdf, 2007.
[5] Akyildiz, I. F., Wang, W., & Wang, W. (2005, January). Wireless mesh networks: a survey. Computer Networks Journal, 47(4), 445-487.
[6] D. Stinson. Cryptography: theory and practice. CRC press, 2006.
[7] Eriksson, J. and Koivunen, V.: Identiability, separability, and uniqueness of linear ica models. IEEE Signal Processing Letters, 11(7), July 2004.
[8] P. Tague, M. Li, and R. Poovendran. Mitigation of control channel jamming under node capture attacks. IEEE Transactions on Mobile Computing, 8(9):1221–1234, 2009.
[9] Ismail, Z., Hassan, R. “Effects of Packet Size on AODV Routing Protocol Implementation in Homogeneous and Heterogeneous MANET” Third International Conference on Computational Intelligence, Modelling and Simulation (CIMSiM), 2011, pp. 351 – 356
[10] Thorat, S.A., Kulkarni, P.J. “Design issues in trust based routing for MANET” International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2014,pp. 1 – 7. [11] Durai, K.N., Baskaran, K. “Energy efficient random cast DSR protocol with mediation device in MANET” International Conference on Advanced Computing and Communication Systems (ICACCS), 2013, pp. 1 – 5.