CBUPRE to Secure MANETs from DoS Attacks

Radha Krishna Reddy Pallavali[1], Samia Bouzefrane[2(✉)], and Selma Boumerdassi[3]

[1] Computer Science and Engineering, Lisbon, Portugal
Pallavali@gmail.com
[2] CEDRIC Labs, CNAM, Paris, France
Samia.Bouzefrane@cnam.fr
[3] INRIA, Paris, France
Selma.Boumerdassi@inria.fr

Abstract. Nowadays, mobile devices are an increasingly useful part of our daily life. Networks through which mobile devices communicate are named Mobile Ad-hoc Networks (MANETs). Without using a fixed infrastructure, mobile nodes dynamically build a wireless network for communication. In order to ensure reliable communications, a main security issue in MANETs is protection from Denial of Service (DoS) Attacks. From past decade, only a few proposals have been made to secure MANETs from DoS Attacks. Generally, the normal encryption schemes protect data confidentiality from unauthorized users, but these techniques are limited to encrypted data sharing between mobile nodes. To overcome this issue, in this paper we propose to use a new proxy re-encryption technique called Compression-Based Unidirectional Proxy Re-Encryption (CBUPRE) for secure data exchange and efficient bandwidth use even during DoS attacks. This technique uses a new node called Dynamic TCP Proxy Node to check the authorization of nodes, to encrypt at a proxy level, and to secure the network against unintended nodes. Based on this, secure data exchange can be achieved with encrypted data. We present a correctness proof of how, through the CUPRE technique, the data can be transferred to the destination node in the group within the low bandwidth network securely even during DoS attacks.

1 Introduction

As stated in [1], MANETs are assemblies of mobile nodes with the capability of configuring any momentary network without the assistance of any established infrastructure or centralized supervision. MANETs are not permanent in nature and are self-organizing because they use multi-hop communication mechanisms. In these mechanisms, each mobile node itself acts as a router to help other nodes. The features of MANETs, such as dynamic topology, mobility, and resource constraints make MANETs suitable to operate under disaster conditions, in the battlefield, news feeds, video-audio transmission, multiparty video games, etc. As described in [2], the nodes in MANETs create a small group network and stay connected to perform particular tasks. In this network, any node may join and leave at any moment in time, resulting in

S. Boumerdassi et al. (Eds.): MSPN 2015, LNCS 9395, pp. 163–175, 2015.
DOI: 10.1007/978-3-319-25744-0_14

dynamic changes in the network. Due to being dynamic in nature, these networks are vulnerable to denial of service (DoS) attacks, eavesdropping of communication channels and e-commerce transactions modifications [3]. In MANET, two nodes can communicate directly with each other as long as they are in radio communication range. In sensitive environments (e.g. military), data exchange must be secured between the communicating parties through intermediate nodes.

In general, DoS attacks are the most critical attacks in computing, relatively easy to implement and extremely difficult to detect and to prevent [4]. The DoS attack has several forms: HTML DoS attack, XML DoS attack or distributed DoS attack (DDoS attack). It consumes an entire network's resources or machine services, making resources unavailable to its respective users [5]. This kind of attack can affect any layer of an open systems interconnection model. In MANETs, the DoS attacks can be performed on MAC layer by several methods [6]. Because MANETs are resource-constraint based networks, the secure group communication must be efficient and inexpensive.

Radio signals can be jammed by wireless signal jamming, disruption of existing communication paths, and saturation of a normal node with a large number of fake communication requests by an illegitimate user. It is even easier for malicious insiders to successfully achieve DoS attacks that consume the constrained energy of the mobile nodes by flooding the network with garbage packets [7]. To achieve a MANETs secure group communication, a group key needs to be established among multiple nodes to ensure a secure exchange of secret information within the group [4]. Due to the distributed nature of the users and the systems, there is a growth of data exchange on existing network connections [8]. Due to slow networks, this results in poor performance. To overcome this problem, organizations moved to data compression techniques [9–12]. To convert original data into alternative formats with lower size, data compression techniques are used.

The major compression technique of fixed length code with n-grams uses 128 most frequently occurring n-grams. This algorithm generates 266 symbols based on frequent key sets used. Where 1 to 87 is for blank, upper and lower spaces, digits and special characters, 88 to 255 for n-grams compression = (Master + Combining). Here Master with 8 bits for A, E, I, O, N, T, U, blank and Combining with 21 bits for blank, plus everything except J, K, Q, X, Y, Z. It takes the original plain text as input and gives the compressed text as an output. A fixed length window 'n' is used to scan each word of the data; all possible n-grams are extracted, stored and analyzed. In n-grams we assumed that each word started with a letter and ended with space. With the below formula we can calculate how many n-grams are possible in a given data dictionary:

$$\mathrm{N}\ (\mathrm{n}) = \sum_{w-n}^{m} (w - n + c) f(w)$$

Where *N(n)* be the number of n-grams in the text, m be the maximum word length, *f(w)* be the frequency of words with length w, $c = 2$ with spaces.

Finally, the unidirectional proxy re-encryption (UPRE) technique is used to authenticate user nodes in the group and to exchange data and secret keys as well [13]. The UPRE scheme that allows a secret key holder node to create re-encryption key, and the proxy can use this key to translate a message that already encrypted by the sender node. This can be done without providing the authority on encrypted data to the proxy node. In detail, the

proxy node can neither recover the sender node's secret key nor decrypt the sender's ciphertext. However, UPRE schemes are under digital rights management (DRM) [14], distributed file storage systems [23], law enforcement [16], etc., In this case, there is no question of compromise at the proxy level re-encryption. Note, we can observe this type of compromise in Apple iTunes DRM [23]. The following are the few properties of UPRE, makes data transformation is more secure [17]:

Unidirectional: the encryption process is allowed from Alice to Bob only, but not from Bob to Alice.

Interaction prevention: for re-encryption key generation, the secret key of Alice and public key of Bob are only required, means that Bob's secret key is not required.

Authentic access: the encrypted ciphertext can decrypt by Alice because she is the authentic user.

Proxy invisibility: Actually the proxy is transparent, but we allowed the senders encrypted message, that can only be opened by the intended recipient (receiver).

Collusion-resistance: Alice and proxy both are working together, but they doesn't know anything about Bob's secret key.

In this paper we propose to use a combination algorithm for MANETs to secure data sharing and, for efficient network bandwidth usage, to combine a compression technique with UPRE. Even during an attack to a network, data already transmitted to that network must reach its destination node securely. We expect this to be achieved through data compression techniques and effective use of available bandwidth even during network attacks.

2 Related Work

Due to their inherent mobility patterns and dynamic topology, MANETs are susceptible to DoS attacks [18]. DoS attacks on MANETs have received significantly less attention than DoS attacks on wired networks. Many solutions have been proposed to protect networks from DoS attacks, some of which have also revealed solutions for ad-hoc networks problems.

In [7], the authors describe the design of a capability-based security mechanism (CapMan) to defend DoS attacks on MANETs, in particular for multi-path communications, where participating nodes are distributed along multi-path communications. The nodes maintain a global view based on summary of capability message exchange, and then dynamically adjust its local constraints to prevent DoS attack on a specific node. The exchange of summary messages between nodes can alter and consume all the network bandwidth.

In [3], the authors used IEEE 802.11 distributed coordination function (DCF) protocols to detect the DoS attacker in MANETs with different PHY layer techniques, by employing several MAC layer protocols. In Bianchi's model, the baseline is used for theoretical throughputs of different technologies. Even though the attacker doesn't have valid information about a transmitted packet, he/she manipulates the IEEE 802.11DCF standards to achieve successful packet transmission on his/her machines, consuming legitimate users' nodes bandwidth. A two-dimensional Markov Chain is used to determine the network capacity. The following are a few issues that that paper didn't

explain: DoS attacks related with other layers, no Quality of Service guaranteed through IEEE 802.11 DCF, no priority-based traffic and if all the nodes want to communicate at a single point of time, many collisions occur, leading to congestion collapse.

In [19], the authors implemented six different types of DoS attacks (flood, sinkhole, black hole, selective forward, selfish node attacks and re-request flood) in an ad-hoc on demand distance vector (AODV) and presented an ad-hoc on-demand multi-path distance vector-intrusion detection system (AOMDV-IDS) protocol, to detect and avoid malicious nodes in MANETs. It's a three-phase method: initial trust establishment, detection and elimination of intruder, and alternate path establishment. AODV is combined with the simple model to discover trustworthy paths and only single time route discovery is initiated. Packet forwarding ratio is used to evaluate node's trustworthiness and a continuous product of node trusts to estimate a path trust. There are a number of major problems with that paper, however: it does not include in the model a security for key exchanges to find initial trust among the nodes, there is no explanation on how the verification of intruder and elimination will take place, intermediate nodes can lead to inconsistent routes if the source sequence number is very low, and there is unnecessary bandwidth consumption due to periodic beaconing.

Especially problematic type of DoS attack is called the gray-hole attack. This attack consumes all the system's resources, and isolates legitimate users from the network by dropping the received data packet during the route discovery phase, [20] deals with this type of attack. To detect the attack, the authors propose a novel statistical approach. Based on two Bayesian classification models (Bernoulli and Multinomial with AODV), a routing protocol is developed. Bernoulli's mathematical model is used to identify the behavior of a node using vectors, and based on three exclusive types of packets: route request, route reply, and data packets. From these, the probability of dishonesty of a node over a defined period of time is calculated. The detection scheme is described as follows: the node listens to its neighborhood periodically and collects information about the forwarded packets for its neighbors. The information is then filtered and mathematically modeled as vectors of behaviors. Using the Bayesian filter, the probability of dishonesty for a given vector is calculated using a decentralized process, in order to fit with the mobility and with the dynamic topology of MANETs.

In [21], the authors improved the avoiding mistaken transmission table (AMTT) to prevent flooding attack, which targets DoS attacks on network layer of MANETs through the following three phases: route discovery, route maintenance and data transfer. This improvement is based on the neighbor suppression technique, which identifies malicious nodes during route construction phase. If a malicious node found, then it is isolated for some time and given an adequate penalty. This system monitors the behavior of the network to avoid DoS attacks at network layer. Each node checks the blacklist field in IAMTT before transferring the data packet. The problems are: once a DoS attacker node is found, this approach considers that node as a normal node temporarily. However, if it happens a second time, then the node is put into a blacklist. Sometimes a normal node may request a route again and again, which may be considered as RREQ flooding attack and the node, even though legitimate, can be considered as misbehaving and blacklisted from the network forever.

These existing solutions are based on a variety of security protocols aiming to provide secure communication (from DoS attacks) between nodes without reducing the

size of the data exchanged. These recent schemes are not; however, secure enough, because all the techniques are not related with cryptographic techniques. Also, each of these solutions deals with only one specific type of DoS attack. To overcome this limitation, we propose a solution, described in the next section, which applies the combination of data compression technique with UPRE while securing the communication in MANET's environment and maintaining an efficient bandwidth usage as well. The major contribution is that, the fixed length codes compression technique is combined with unidirectional proxy re-encryption technique. The encryption technique provides security for the data that the sender node wants to send via the network to the destination node, whereas the compression technique reduces the data size before it sending to the network for transmission. The main theme of compression technique is that, when the data was transmitted through the network during the DoS attack submitted to the destination node, by using the low bandwidth. In case of UPRE, the data transmitted securely even a key was compromised with the attacker, because the data was encrypted twice.

3 CBUPRE Methodology

CBUPRE: The proposed CBUPRE technique is to secure the communication between nodes in the group and efficient bandwidth usage even during DoS attacks in MANETs. With this technique, we improved upon the UPRE technique based on fixed length codes compression. In sender nodes side, encryption technique follows the compression technique to send the data securely in the MANETs group. In the same way, at the destination node side, the above scheme is performed in reverse so that the original plain text is achieved.

CBUPRE takes place in the following two phases:

1. Fixed Length Codes (FLC) compression.
2. Unidirectional Proxy Re-Encryption technique.

3.1 Phase 1-FLC

The main functionality of the FLC, for instance, two nodes are connected in remote LANs by a pair of routers, bridges and modems. At each end-point of the network connection, compression is done by network devices or by computers. Before putting data into the packet, the data is compressed first, because the compression of on packets generates the same number of packets. So there is no use to compress the packets. If data encryption is required, then data should be compressed first and then encrypted, before placing it in the packet [22].

3.2 Phase 2-UPRE

The second phase of our CBUPRE is UPRE [23]; the proxy is fully trusted because data sharing is done through the node identity and data compression. The Identity

Proxy node generates proxy level key in the group. The proxy signers in the proxy group are responsible for proxy key generation behalf of the original user identity.

Encryption E = (Setup, User level Key Generation, User level Encryption, Proxy level Key Generation, Proxy level Encryption).

As mentioned above, the encryption of compressed data takes place twice, once in a sender node level with senders' private key, and then at the proxy level with a re-encryption key. Then the achieved cipher-text is transferred to destination node, before receiving at destination side the proxy-level cipher-text decrypted with public key and sent to the destination node. Then with the sender nodes' public key decrypts user level cipher-text and received original compressed text.

3.3 Assumptions

Based on the below assumptions and related work, we can say that our proposal does not fall into single point failure problem, secure communication achieved, and efficient bandwidth usage, and key management is also achieved.

- This group communication is under distributed control.
- All the nodes in the group acts as end-systems as well as routers to forward the packets [21].
- Dynamic TCP proxies [24].
- Clustering based group key management [25].
- The newly joining or exiting nodes must follow the protocol standard.

3.4 Methodology of CBUPRE Technique

Our scheme is compression-based unidirectional proxy re-encryption and it is an improved version of UPRE [13].

3.4.1 Sender Side Encryption

The message space M, W is the word space in the message space = $w_1, w_2, \ldots, w_n$, C is the compressed text, C' = cipher text (already compressed), usk_i, upk_i are user level secret and public key pair, node identities Id_i, Id_j and etc.

1. **Setup:** uses system's parameters to setup our approach. We assume that k implicitly included in the following algorithms. The system parameters include a description of a compressed data and a description of a cipher text.
   a. **FLC:** The conditional probability $\Pr(w_i|w_i^{n-1})$ of the word space for an arbitrary n-gram $W = (w_1, w_2, \ldots, w_n)$ as follows:
      i. $\alpha(w_1^n)$ if w_1^n exists
      ii. $\beta(w_1^{n-1})Pr(w_n|w_2^{n-1})$ if w_1^{n-1} exists
      iii. $Pr(w_n|w_2^{n-1})$ Otherwise. Where $\alpha(w_1^n)$, $\beta(w_1^n)$ are smoothed probabilities
   b. **UPRE:** The two primes p and q such that $q|p\text{-}1$ and the bit-length of q is the security parameter k, g be the generator of group G, which is the subgroup of Z_q^*

with order q. Choose four hash functions $H_1 : \{0,1\}^{l_0} \times \{0,1\}^{l_1} \rightarrow Z_q^*$, $H_2 : G \rightarrow \{0,1\}^{l_0+l_1}$, $H_3 : \{0,1\}^* \rightarrow Z_q^*$ and $H_4 : G \rightarrow Z_q^*$. Here l_0 and l_1 are security parameters determined by k, and the message space M is $\{0,1\}^{l_0}$. The system parameters are $param = (q, G, g, H_1, H_2, H_3, H_4, l_0, l_1)$.

2. **Compression:** It takes the original plain text as input and gives the compressed text as an output.
   a. For example the trigram of a word space W is follows: $\Pr(w) \approx \prod_k \Pr(w_k|w_{k-2}.w_{k-1})$, then the probability of $\Pr(w_k|w_{k-2}.w_{k-1}) \approx \frac{frequency(w_{k-2}w_{k-1}w_k k)}{frequency(w_{k-2}w_{k-1})}$
3. **User level key pair generation:** user's identity is the input and the public, private key as a pair is the output.

$$UKeyGen() : Pickusk_i = (x_{i,1} \xleftarrow{\$} Z_q^*, x_{i,2} \xleftarrow{\$} Z_q^*) and\ set\ upk_i = (upk_{i,1}, upk_{i,2}) = (g^{x_{i,1}}, g^{x_{i,2}}).$$

4. **User level Encryption:** Compressed text and sender node's private key are inputs and the output is first/user level cipher-text.
   $UEncrypt(upk_i = (upk_{i,1}, upk_{i,2}), C)$: To encrypt a compressed text $C \in W \in M$:
   a. Pick $\xleftarrow{\$} Z_q^*$ *and compute* $D = (upk_{i,1}^{H_4(upk_{i,2})} upk_{i,2})^u$.
   b. Pick $\xleftarrow{\$}\{0,1\}^{l_1}$, compute $r = H_1(C, w)$.
   c. Compute $E = (upk_{i,1}^{H_4(upk_{i,2})} upk_{i,2})^r$ and $F = H_2(g^r) \oplus (C||w)$.
   d. Compute $= v + r \cdot H_3(D, E, F) mod q$.
   e. Output the ciphertext $C' = (D, E, F, s)$.
5. **Proxy level key pair generation:** based on the identity of participant nodes, the key pair is generated for proxy level encryption process.
   $PKeyGen(usk_i, upk_j)$: On input user *i's* secret key $usk_i = (x_{i,1}, x_{i,2})$ and user *j's* public key $upk_j = (upk_{j,1}, upk_{j,2})$, this algorithm generates the proxy level encryption (re-encryption) key $rk_{i \rightarrow j}$ as below:
   a. Pick $h \xleftarrow{\$} \{0,1\}^{l_0}$ *and* $\$ \{0,1\}^{l_1}$, *compute* $v = H_1\ (h, \pi)$.
   b. Compute $V = upk_{j,2}^u$ and W = $H_2(g^v) \oplus (h||\pi)$.
   c. Define $rk_{i \rightarrow j}^{<1>} = \frac{h}{x_{i,1} H_4(upk_{i,2}) + x_{i,2}}$. Return $rk_{i \rightarrow j} = (rk_{i \rightarrow j}^{<1>}, V, W)$.
6. **Proxy level Encryption:** First/user level cipher-text re-encrypted at proxy level by user's private key, so proxy level cipher-text (re-encrypted) resulted, and transferred to destination node.
   $PEncrypt(rk_{i \rightarrow j}, C_i^{'}, upk_i, upk_j)$: On input a proxy level encryption (user i to user j) key $rk_{i \rightarrow j} = (rk_{i \rightarrow j}^{<1>}, V, W)$, an original ciphertext $C_i^{'} = (D, E, F, s)$ under public key $upk_i = (upk_{i,1}, upk_{i,2})$, this algorithm re-encrypts $C_i^{'}$ into another C_j^* under destination node public key $upk_j = (upk_{j,1}, upk_{j,2})$ as follows:
   a. If $(upk_{i,1}^{H_4(pk_{i,2})} upk_{i,2})^s = D \cdot E^{H_3(D,E,F)}$ does not hold, return $\perp$.
   b. Otherwise, compute $' = E^{rk_{i \rightarrow j}^{<1>}}$, and output (E', F, V, W).

Let $= H1(C, w), v = H1(h, \pi)$, the re-encrypted ciphertext is of the following forms: $C_j^* = (E', F, V, W) = (g^{r.h}, H_2(g^r) \oplus (C||w), upk_{j,2}^v, H_2(g^v) \oplus (h||\pi))$.

3.4.2 Receiver Side Decryption

Decryption can be done in two levels, proxy level, user level and finally decompression to get the original text message from sender node:

7. **Proxy level Decryption:** proxy level encrypted data C^* and destination node's public keys are inputs and user level cipher-text is the output.
   C^* is a re-encrypted ciphertext in the form $C' = (E', F, V, W)$:
   a. Compute $(h||\pi) = W \oplus H_2(V^{1/uski,2})$ and $(C||w) = F \oplus H_2(E'^{1/h})$.
   b. Return C' *if* $V = upk_{i,2}^{H_1(h,\pi)}$ and $E' = g^{H_1(C,w)\cdot h}$ hold; else $\perp$.
8. **User level Decryption:** User level cipher-text *C'* decrypted by using user's public key, so it results compressed text decrypted by using user's public key, so it results compressed text.
   C is an original ciphertext (compressed) in the form $C = (D, E, F, s)$:
   a. If $(upk_{i,1}^{H_4(pk_{i,2})} upk_{i,2})^s = D \cdot E^{H_3(D,E,F)}$ does not hold, return $\perp$.
   b. Otherwise, compute $(C||w) = F \oplus H_2(\frac{1}{E^{xi,1H4(upki,2)+xi,2}})$.
   c. Return C if $E = (upk_{i,1}^{H_4(upk_{i,2})} upk_{i,2})^{H_1(C,w)}$ holds; else return $\perp$.
9. **FLC Decompression:** through the compressed text and the key set, the destination node extracts original plain text finally.

N-grams decompression of word space $W \in M$ message space = $(\frac{frequency(wk)}{N})$, where N-number of words. For example to unigrams: $r(w) \approx \prod_k Pr(w_k)$.

4 Experimental Discussions

The main theme of this experiment is to determine the effectiveness of our compression based unidirectional proxy re-encryption technique approach for secure group communication in MANETs and efficient bandwidth utilization even at the time of DoS attack occurred. In this proposed framework, there are nine steps as follows: compressed data with fixed length codes compression technique by using keyset of the source node, setup, key generation, User level encryption, proxy key generation, proxy level encryption, Proxy Decryption and User decryption on destination node, and then finally decompression.

Step 1: The setup phase, which sets-up the system, is ready to deal with compression technique and unidirectional proxy re-encryption technique. The inputs are the word space within the message space, and security parameters. Based on this, the output is system parameters, and includes a description of a finite compressed message space and a description of a cipher text.

Step 2: In the second step, the original data which we would like to transfer to the destination node in MANET communication is compressed by using the keyset from

source node with fixed length codes n-grams compression technique before packetting. This is because of the compression technique on packets gives the same number of packets, which is not useful.

Step 3: This user-level key generation step generates key pair based on the node identity on either nodes.

Step 4: In the fourth step, we will get user level cipher-text from compressed text and private key of sender node identity.

Step 5: This step is proxy-level key generation; it services the proxy to encrypt at proxy level whenever the destination node requires the data from source. This is the one way key generation from source to destination based on their identities.

Step 6: This is the proxy-level encryption step (proxy re-encryption) of user level cipher text by proxy node. The proxy will encrypt the first level cipher-text and transfer to destination node.

Step 7: This is the proxy-level decryption step in destination node side. In this step when destination node is ready to download data, the proxy re-encrypted message is decrypts at proxy level and gives to destination node.

Step 8: This is the user level decryption step, in this process the user gets compressed data from user level encrypted data by using his public key.

Step 9: This is the final step in destination node side to get original data from compressed text, based on decompression scheme with keyset.

Based on the working nature of our algorithm, we compared it with existing DoS detection and prevention methodologies. The Table 1, which shows the major advantages, such as bandwidth usage during the DoS attack, the layers in which the attack could be detected, and the cryptographic technology. The next section correctness proof, that supports our proposed scheme based on our CBUPRE.

Table 1. Comparisons between existing approaches with CBUPRE

	Detection	Prevention	No. of nodes	Bandwidth	Layer	Cryptography
CapMan		√	Dynamic		Transport	
IEEE802.11 DCF	√		Fixed		Data link	
AOMDV	√		–		–	
Statistical Method	√		Fixed		–	
IAMTT		√	–		Network	
CBUPRE	√		Dynamic	√	Presentation & Network	√

5 Proof of Correctness

This section proves our proposed scheme, which ensures a secure MANET group communication. Also, we assume that every incoming node is an authenticated node. This makes an incoming node is possible to study secure data transmission scheme in MANET group communication.

Lemma 1. All members of MANET's groups must know that they are using n-grams fixed length codes compression technique to reduce the burden on network bandwidth.

Proof. The n-grams FLC technique uses same number of bits for each symbol. For example, K-bit code compression technique supports 2 k different symbols. The below illustration shows the usage of FLC compression technique with n-grams. Where n = 1, 2, 3….. We take the following text for our example illustration.

"Was receiving a percentage from the farmer till such time as the advance should be cleared off oak found + that the value of stock, plant, and implements which were really his own would be about sufficient to pay his debts, leaving himself a free man with the clothes he stood up in, and nothing more.

<C vi > < P 88>
THE FAIR "THE JOURNEY" THE FIRE TWO months passed away. We are brought on to a day in February, on which was held the yearly statue or hiring fair in the country-town of Casterbridge. At one end of the street stood from two to three hundred blithe and hearty laborers waiting upon change "all men of the stamp to whom labor suggests nothing worse than a wrestle with gravitation, and pleasure nothing better than a renunciation of the same among these, carters and wagoner's were distinguished by having a piece of whip-cord twisted round their hats;"

Based on FLC n-grams compression for the above text, we have to calculate the different frequencies. First the unigram frequencies, then most frequently occurred bigrams, and then trigrams based on the bigram frequencies and etc. from the above text the n-grams where n = 1, 2, 3 calculated and displayed in Table 2.

Table 2. Trigrams fixed length compression for the above text

Count n = 1	Count n = 2	Count n = 3
125551	20452 e	11229 th
72431 e	17470 he	9585 the
50027 t	17173 t	8989 he
…..	…..	…..
…..	…..	…..

Based on this n-grams FLC technique the size of the data is reduced. So we can say that the data compression is guaranteed and the size of data is reduced by using n-grams fixed length compression technique.

Lemma 2. In the MANET group, every node is identified by their node identity. And all group members must know it.

Proof. We would like to prove this one by contradiction, assuming that there are two nodes with identities I d i and I d j that want to exchange data in a MANET group. In this MANET group nodes I d i and I d j are already authenticated based on their node identity. Thus, two different cases can occur:

Case 1: the two nodes I d i and I d j are in direct communication in the group, which means no intermediate nodes are available between them in the link. So there is no need to think about attacker nodes. Why? Because even if the network bandwidth is very low our method works effectively due to data compression.

Case 2: the nodes I d i and I d j are not neighbours, but communicated in the same MANET group. Between these two nodes there other nodes that are also authenticated by their node identity. So the transmission occurs on whichever link is near with few hops and authenticated nodes between them. This way, the data is shared securely.

Thus, the MANET group is formed securely with node identifications. This proves that there is a contradiction with the assumption that each group maintains authenticated group members only.

Lemma 3. When an unauthorized node wants to join in the group with fault identity in proxy level, it cannot decrypt the encrypted data.

Proof. By Lemma 2 case 2, if any unauthorized node between authorized nodes I d i and I d j wants to join in the network with fault identity N a at proxy level. Even though this N a node accepted to participate in group communication, the proxy level encrypted data will not be decrypted. Why? Because it doesn't contain the proxy level key of particular attacked node. The proof could drive to a conclusion that every node must have a proxy level encryption key to decrypt proxy level cipher-text.

Lemma 4. When an unauthorized node wants to join in the group with faulty identity in user level, it cannot decrypt the encrypted data and cannot decompress it.

Proof. By Lemma 2 case 2, if any unauthorized node N a wants to join between authorized nodes I d i and I d j in network with fault identity at user level. Even though this N a node accepted to participate in group communication, the user level encrypted data will not be decrypted. Why? Because it doesn't consist the user level secret key of particular attacked node. At peak moment, it got decrypted user level cipher-text, but it doesn't know the meaning of the compressed text message.

The proof could drive to a conclusion that every node must have a secret key to decrypt proxy level cipher-text and must know the data compression technique pattern which is used at source node.

Theorem. In MANETs, group communication, secure data transformation is guaranteed, even in lower data bandwidth.

Proof. This can be done through Lemmas 1, 2, 3 and 4.

6 Conclusions

In this paper, we proposed an efficient scheme for secure data exchange and bandwidth usage in MANETs, based on a Unidirectional Proxy Re-Encryption technique (CBUPRE) to make them secure from DoS attacks. With this scheme, the direct and secure communications between any identified nodes in a group can be implemented. Due to fixed length codes compression technique, the data transformation ratio is

increased and the delay of communication is lowered. Hence the scheme is more flexible for MANETs. Any member node in a group can transform the data securely due to algorithms unidirectionality. Our future work is to simulate CBUPRE based on the hypothesis that we made, and to construct more secure and efficient group communication for MANETs based on CBUPRE.

References

1. Kaur, T., Toor, A., Saluja, K.: Defending manets against flooding attacks for military applications under group mobility. In: 2014 Recent Advances in Engineering and Computational Sciences (RAECS), pp. 1–6 (2014)
2. Konate, K., Abdourahime, G.: Attacks analysis in mobile ad hoc networks: modeling and simulation. In: 2011 Second International Conference on Intelligent Systems, Modeling and Simulation (ISMS), pp. 367–372 (2011)
3. Soryal, J., Saadawi, T.: IEEE 802.11 denial of service attack detection in manet. In: Wireless Telecommunications Symposium (WTS), pp. 1–8 (2012)
4. Pushpalatha, K., Chitra, M.: Gamanet: A ganatic algorithm approach for hierarchical group key management in mobile adhoc network. In: 2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering (PRIME), pp. 368–373 (2013)
5. Reddy, P.R.K., Bouzefrane, S.: Analysis and detection of dos attacks in cloud computing by using qse algorithm. In: 2014 IEEE International Conference on High Performance Computing and Communications, 2014 IEEE 6th International Symposium on Cyberspace Safety and Security, 2014 IEEE 11th International Conference on Embedded Software and Systems (HPCC,CSS,ICESS), pp. 1089–1096 (2014)
6. Lolla, V., Law, L., Krishnamurthy, S., Ravishankar, C., Manjunath, D.: Detecting mac layer back-o timer violations in mobile ad hoc networks. In: 26th IEEE International Conference on Distributed Computing Systems, ICDCS 2006, p. 63 (2006)
7. Jia, Q., Sun, K., Stavrou, A.: Capman: capability-based defense against multi-path denial of service (dos) attacks in manet. In: 2011 Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN), pp. 1–6 (2011)
8. MacVittie, L.: Understanding advanced data compression (2013)
9. Base, O.D.: Oracle advanced compression with oracle database 12c (2015)
10. Oberhumer, M.F.: lzop (2010)
11. Seward, J.: bzip2 and libbzip2, version 1.0.5, a program and library for data compression (2007)
12. loup Gailly, J., Adler, M.: A massively spiffy yet delicately unobtrusive compression library (2013)
13. Wang, H., Cao, Z., Wang, L.: Multi-use and unidirectional identity-based proxy re-encryption schemes. Inf. Sci. **180**, 4042–4059 (2010)
14. Smith, T.: Dvd jon: buy drm-less tracks from apple itunes, 18 March 2005
15. Ateniese, G., Fu, K., Green, M., Hohenberger, S.: Improved proxy re-encryption schemes with applications to secure distributed storage. ACM Trans. Inf. Syst. Secur. (TISSEC) **9**, 1–30 (2006)
16. Ivan, A.A., Dodis, Y.: Proxy cryptography revisited. In: NDSS (2003)
17. Reddy, P.R.K., Sivaramaiah, S., Sesadri, U.: Secure data forwarding in cloud storage system by using umib proxy. Int. J. Comput. Technol. **10**, 1905–1912 (2013)

18. Wu, B., Chen, J., Wu, J., Cardei, M.: A survey of attacks and countermeasures in mobile ad hoc networks. In: Xiao, Y., Shen, X.S., Du, D.-Z. (eds.) Wireless Network Security, pp. 103–135. Springer, USA (2007)
19. Gupta, H., Shrivastav, S., Sharma, S.: Detecting the dos attacks in aomdv using aomdv- ids routing. In: 2013 5th International Conference on Computational Intelligence and Communication Networks (CICN), pp. 380–384 (2013)
20. Rmayti, M., Begriche, Y., Khatoun, R., Khoukhi, L., Gaiti, D.: Denial of service (dos) attacks detection in manets through statistical models. In: Global Information Infrastructure and Networking Symposium (GIIS), pp. 1–3 (2014)
21. Ahir, S., Marathe, N., Padiya, P.: Iamtt - new method for resisting network layer denial of service attack on manet. In: 2014 Fourth International Conference on Communication Systems and Network Technologies (CSNT), pp. 762–766 (2014)
22. Allen Kent, J.G.W.: Artificial Intelligence and ADA to Systems Integration: Concepts: Methods, and Tools (1992)
23. Ateniese, G., Fu, K., Green, M., Hohenberger, S.: Improved proxy re-encryption schemes with applications to secure distributed storage. ACM Trans. Inf. Syst. Secur. (TISSEC) **9**, 1–30 (2006)
24. Ouyang, T., Jin, S., Rabinovich, M.: Dynamic tcp proxies: coping with disadvantaged hosts in manets. In: 29th IEEE International Conference on Distributed Computing Systems Workshops, ICDCS Workshops 2009, pp. 530–536 (2009)
25. El-Sayed, A.: Clustering Based Group Key Management for MANET. In: Awad, A.I., Hassanien, A.E., Baba, K. (eds.) SecNet 2013. CCIS, vol. 381, pp. 11–26. Springer, Heidelberg (2013)