Secure Group Key Management in Adhoc networks for peer node to group nodes communication

Candidate: Kasi Sravani (Mtech Computer Science and Engineering) kasi.520@gmail.com
Guide: Mrs.YVD Pushpa latha, M.Tech
Maharaj vijayanagaram gajapathi raj college or engineering http://www.mvgrce.com

Abstract:

The theme of this paper (*SGMNpNGN)* is to create the static/dynamic peers (in simulated adhoc network), which will be with the control of one master administrator / super node. This super node will be having following features over the sub nodes.

- **Cast creation**: This can create target cast (GROUP) nodes.
- Dynamic source node for data transmission.
- Encryption and decryption based on algorithm (Rota).
- Key generation control algorithm based on SHA-2.

Basically the user can create source and destinations (casting nodes) for encrypted (all possible selected formats) for data transmission. The encryption and decryption is user's choice(according to mode of transmission). And this is implemented here.That is.

- Rota

Whenever the transmission is proposed the key will be generated automatically by using SHA-2 for transmission. Here this will be controlled by only super node. Creating the trusted peer as administrator or super peer in wireless adhoc sensor networks is a big challenge. So admin will choose trusted node (in cast of manet) for transmission. Basically nodes can be generated by users request by super node but controlled by on super node. Hence all encryption or decryption and key generation will be at super node level only.

Index Terms:*Cast, Group, Peer, Peer Level, Secure communication, SNMP, SMTP*

1. Introduction:

Securing a computer network requires mastery of many skills and concepts are proper design of network, device and deployment, protocols, etc.

Existing concerns in network security:

- The issues can arise if there are IP based phones on a network that uses the same port!
- It will be possible to get the following data from IDS.

1. Total number of packets that node receives from one single destination only.
2. Total number of packets that node receives from all sources.
3. The time between two received packets.

- Wheatear to check the information from a security aspect, or the focus is towards other data nodes
- If public SMTP servers are located in a perimeter network, what is the procedure for tunneling internal mail through the firewall?
- Under what parameters or boundaries would you recommend building your own intrusion detection system (IDS)?
- Which way the client will connect/reconnect without using secure socket layers able to connect to remote secure server?

To overcome above concerns this paper proposed this DYN Peer proposal algorithm to have tree structure adhocnetwork (can be migrated to other topology with in no time of i.e. migrating to other networks instantly. DYN Peer proposal generates prior static peer adhoc network and after that it will instantly acts as dynamic peer addition/deletion at any instant of working network. This Three level key protection is available in this algorithm for secure packet TCP/IP transmission over the tree adhoc network.

The main purpose of DYN Peer proposal to overcome the instant casting issues as target nodes. The source node is always can be any one in available adhoc network and casing nodes(group nodes as target nodes) to transmit data and both will be controlled by administrator peer which is ROOT node in tree network. The node which is sending the data will be protected by keys to transmit the data. The following are the available keys in our proposal.

2. Key Management types:

Private:

- This will be tracked and owned by root/Administrator peer.
- This is one time generation for each and every process/session.
- Generated by SHA-2 algorithm.
- The format of the key is like this.

Public:

- Visible/sharable to peers in a group.
- This key will be generated uniquely whenever a peer is added or removed dynamically.
- This key will be tracked by root node (which generates the key for all peers).
- Generated by SHA-2 algorithm.
- A group key cannot be visible to other groups (not sharable).
- The format of the key is like this.

Protected:

- Sharable to all the participants in data transmission over network.
- This key will be generated randomly/uniquely whenever the network starts data transmission.
- Generated by SHA-2 algorithm.
- The format of the key is like this.

Node generation (DYN peer proposal 3-stage) Stages:

- Node Generation
- Source / destination casting nodes.
- Transmission

MANET Generation:

Algorithm:

1. $m \leftarrow 0\, masternodegeneration$
2. $m \in \sum_{n-1}^{0} staticnodegeneration$
3. $m_c \rightarrow n - 1\, masternodecontrolingallchildtreenodes$
4. $m \leftarrow n - 1\, mastergenerationfordynamicpeers$
5. $L0 \leftarrow m_c$
6. *L0 L1 L2 … Li (Levels are generated in tree structure)*
7. $m(n - 1) \sum_{i-1}^{0} dynamicnodegeneration$
8. $m \pm \Longrightarrow {}^{(n-1)}_{(i-1)}$
9. $S \leftarrow Sourcenode(Si)$
10. $D \in d\,{}^{i}_{n}$ //
11. $Trans(S, D)$

Node generation:

Master node generation initialize the nodeDYN peer to *m* as 0. Statically we estimated n-1 peers which pre tree is framing. Dynamically nodes are generated at any level

of tree (already framed static tree). m_c Will control all the statical/dynamical peers at any given time interval. Here, $m(n-1)\sum_{i-1}^{0}$,peers starts from 1 to i-1 traversal would be feasible if the peers controlled by m at any level(*L1, L2,….LI*).

Transmission stage:

The peers generation will be controlled by m.(which is root node and) this root node have full access to control i-1 levels and n-1 peers to transmit data. D is destination casting nodes can be any level and any available nodes for transmission. The casting is done instantly for one transmission with log information further.

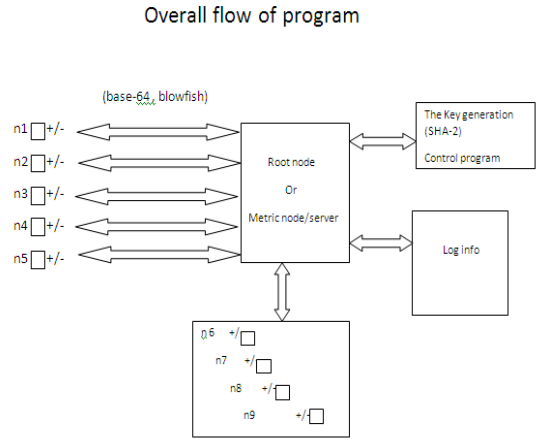
Transmission:

The transmission is always secured with available secure algorithms. The **Trans**(S, D) function in step11 contains secure transmission stage.

Challenge:

The feasible transmission is always in this adhoc network for further transmission and the available challenge is still this adhoc network is static nodes generation (starting tree formation) and later this will be added with dynamic peers additions. This is future enhancement for DYN Peer proposal.

The flow of *SGMNpNGN:*



SGMNpNGN issecure group transmission for the data sharing (communication) to dynamically generated peers in the *adhoc* network. The master peer will have all control over the generated/added peers (statically and dynamically). The peer network will be framed in tree structure with root node (peer) is the master node. The peers can be added and deleted in the tree structure without disturbing available structure of the network for communication.

Once the transmission is enabled, the source node will frame the selected documents (any supportive formats). The source node will get the key from master node for transmission. The transmission can be for multiple peers under one umbrella, which is grouped for secure communication. The grouping is done for selected peers from any level in the adhoc tree. The communication will be parallel communication from source peer to destination group (peers).Once the grouping is done, the master peer will generate key for secure transmission. Here we proposed a secure algorithm: SHA – 2.

The key has to be entered to have proper communication. And the transmission logs will be maintained in the log information. The algorithm used to create the key:

- **SHA-2:**

In cryptography, SHA-2 is a set of cryptographic hash functions (SHA-224, SHA-256, SHA-384, SHA-512) designed by the National Security Agency (NSA) and published in 2001 by the NIST as a U.S. Federal Information Processing Standards. SHA stands forSecure with Hash based algorithm. SHA-2 includes a meaningful number of changes from its next predecessor, SHA-1. SHA-2 consists combination of four hash functions with digests ie 224, 256, 384 or 512 bits.

In 2005, security flaws were determined in SHA-1, namely that a calculation wise weakness might exist, indicating that a powerful hash function would be required.[2]. Though SHA-2 express some similarity to the SHA-1 algorithm, these intrusions have not been successfully handed over to SHA-2.

Examples of SHA-2 variants:

SHA-2 Algorithm:

1. Initialize First 8 Prime numbers of fractional square root values into hexadecimals.

   (2, 3, 5, 7, 11, 13, 17, 19)

   H0:
   Sqrt(1/2)=1.414213562=0x6a09e667.

   H1:sqrt(1/3)
   =1.73205080=0xbb67ae85.

   H2:0x3c6ef372.

   H3:0xa54ff53a.

   H4:0x510e527f.

   H5:0x9b05688c.

   H6:0x1f83d9ab.

   H7:0x5be0cd19.

2. Next we have to take first 32 bits of the fractional cube roots to the first 64 prime numbers into an array k[].
3. Append the bit '1' to the message.
4. If k >=0 then append '0'.
5. Message will be partitioned into 512 bits chunks.
6. Each 512 bit separated into 64 bit. First 16 words will be in array. The remaining 48 words are pre processed.
7. A[i]=A[i-16]+b0+A[i-7]+b1

   Where b0= (A [i-15]>>>7) ^ (A [i-15]>>>18) ^(A [i-15]>>>3).

   b1= (A [i-2]>>>17) ^ (A [i-2]>>>19) ^(A [i-2]>>>10).
8. a←H0,b←H1,c←H2,d←H3,e←h4,f←H5,g←H6,h←H7
9. Loop starts I =0-63

   B1= (e>>>6) ^ (e>>>11) ^ (e>>>24)

   Ch= (e Λ f) ^ (⌐ e Λ g)

   Temp1=h+b1+ch+k[i] +A[i]

   B0= (e>>>2) ^(e>>>13) ^(e>>>22)

   Maj=(aΛb)^(aΛc)^(bΛc)

   Temp2=b0+maj;
10. Replace values as while appending

    h=g

    g=f

    f=e

    e=d+temp1;

    d=c

    c=b

    b=a

    a=temp1+temp2;

Examples:

Hash values of empty string.

Text: hi sravani how are you

Encryption:

```
25a7d5bb53a5647a0283629d56777eda24a5c
386aa372534d73406d4e5cdead04a6b7eecef
bb133873422f1d8aa249554a041dd6e68a42d
221f5d0e5e0930920
```

Text: Secure Group Key Management in Adhoc networks for peer node to group nodes communication

Encryption:

```
6570a94570655b45f41fdaf8c2d4236a75aaf
2a6ed017319b65139ab6bc1f2825199faecc2
074be36f932a77f90de501aea31ced6601a16
d9ab9859808fa0ff0
```

Text:

Text to convert to a SHA-512 hash:

Encryption:

```
e6d0ca61a30170581867ec10345ebe4a12b92
3c65bb1d35a465eb174691401fe50864d851e
9d8db1b64ea3e3475edb9395a414272f986aa
dc054d9b89315c848
```

Even a small change in the message will (with overwhelming probability) result in a mostly other hash, due to the sliding effect. For example, adding a slot to the end of the sentence.

The types of keys:

- **Private Key:** This is totally used by Root peer for checking the log information.
- **Group Key:** Whenever the transmission is established, this key is generated by root node and shares only for one communication (source peer to destination group (casted peers) peers).
- **Shared Key:** This key can be shared among the casted peers but generated and controlled by root peer.

Note: all the above keys will be generated dynamically and maintained by root peer.

Rota Encrypt/Decrypt Algorithm:

Encryption:

Input: Plain text

Output: CipherText

Initialization:

n←0 //total number of characters.

∑D←0 //total data

Cipher←1

∑E←0 //total Encrypted data

t1←null //temp variable

t2←null //temp variable

Loop for each c in D

n=0

t1←LSHIFT(c,n,CIPHER)

t2←RSHIFT(c,n+1,CIPHER)

n+1

E←APPEND(t1)

E←APPEND(t2-1)

End loop

Decryption:

Input: Cipher text

Output: Plain text

Initialization:

N←0 //initialization for total number of characters.

∑E←0 //Cipher texts

∑N←0 // plain text

T1←null

T2←null

Loop for each c in E

n=0

t1←RSHIFT(c,n,CIPHER)

t2←LSHIFT(c,n+1,CIPHER)

n+1

D←APPEND(t1)

D←APPEND(t2+1)

End loop

In Rota Encryption Algorithm the Data Files will be encrypted in the form of cipher text. When transmitting the data from node to node the source file data was encrypted by the sender using Rota Encryption Algorithm.

Here the Data Characters will be converted into based ASCII Values. These values are internally turned into binary bits. We perform operations such as Left Shift and Right Shift on Binary Data. If the ASCII values are Even number digit then perform 1LeftShift operation on Even ASCII value of given character(B<<1).its generate Cipher text, and then we again perform 1RightShift operation on cipher text.noe we got Plane text. If the ASCII values are odd number digit then perform 1RightShift operation on oddASCII value of given character (A>>1).Its generate Cipher text, and then we again perform 1LeftShift operation on cipher text.Now we got Plane text.

Example3:

Enter Your String

pleasedonot touch steves pet aligator

Original Test is

pleasedonot touch steves pet aligator

Encrypt Test is

àØ1/81@È6Ü6è@è690Ð@8è1ì18@à1è@/Ø32/è6ä

Decryption Test is

pleasedonot touch steves pet aligator

Advantages:

- Rota encryption provides high security for data files while transmitting data from node to node in manet.
- The main advantage of sha-2 algorithm is it generates same size of key for any size of plain text. so we can specify the size for key.
- In SNMP multiple peers can be managed but very few adhoc networks have the following features: (Dynamic peer generation with static peer network, data sharing among unlimited casting peers as destination group, SHA-2 for secure key generation).
- Logs will be maintained instantly with micro level information.

Secure transmission by using group key:(algorthm2)

1. $\int \leftarrow sel \binom{(n-1)}{(i-1)}$
2. $D \leftarrow \left.\binom{(n-1)(a)}{(i-1)(a)}\right\} cast$
3. $\int \rightarrow t \rightarrow D$

$$m = t - \left[\int \rightarrow t \rightarrow D\surd\right]$$

$$m \neq t - \left[\int \rightarrow t \rightarrow D \times\right]$$

Networking

1. *Networking Infrastructure:*

Networking infrastructure frames the base for the networks on top of which the higher-level

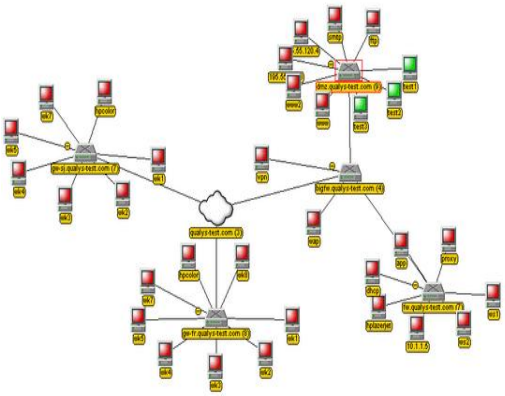
Services can be built. The core of the networking infrastructure is formed by the physical

Topology and the logical structure of the network of which the latter is implemented andmaintained with routing As discussed in[5], there are two approaches in networking:

a).Flat or "zero-tier" infrastructure hierarchical, b).multiple- or N-tier infrastructure.

In flat networks there are no hierarchies of nodes; all nodes have equivalent roles from theperspective of routing. In this aspect, in hierarchical networks there are nodes that have differentCharacteristics of Ad-Hoc Networks

Architecture



Ad-hoc networks do not have any kind of pre-existing infrastructure but they are dynamic by nature and consist of nodes communicating on peer-to-peer basis. Usually, such networks are not very long lived being temporary and constructed on the spot when needed. The nodes forming an ad-hoc network are usually mobile as well as small and limited in their resources, for instance battery size, memory orcomputational power. Fixed networks (like the Internet in most parts) contain static hardware and software that is dedicated to routing messages through the network. Ad-hoc networks lack these kinds of dedicated router networks but the communication is carried by direct peer-to-peer connections between nodes in the ad-hoc network. These peer-to-peer connections may break down suddenly because of many reasons like nodes getting out of reach of the network or running out of battery power [4]. In fixed networks, it is easy to assign static servers for different tasks like being a name server, a certificate authority or a key distribution center. Due to the high reliability of the server platforms and static, fixed networks these centralized servers are practically always reachable. This luxury is absent in an ad-hoc environment since the devices are generally limited and the network is very dynamic. Thus, it is very hard to assign any fixed and static roles in ad-hoc networks.

Modules:

- Manufacturing administration peer in wireless Adhoc sensor network (Math part).
- Group/key Management
    1. Adding key
    2. Deleting key
    3. Rekeying/refreshing key
- Data Transmission/tracking management using algorithm(s). Metrics table (bw,size,time)
- Test Cases(existing/expected/proposed) Issues and salvations.

Our proposed mechanism to have public key over broad casting networks, shared key over multi casting networks, private key over unicasting networks.

Our Base station creation will be on broad cast networks which is base station for all other networks / peers, which are getting dynamically generated in multicasting and single casting.

Base station will be having the root control for generated networks/peers. Base station will be acting as a trusted KDC (Key distribution center) which is having control over all networks/peers.

The transmission over the entire adhoc network will be tracked by administrator for secure data transmission. We can have multiple transmissions over networks, the source will change per transmission depends on selected node to transmit.The destination node(s) will be having restrictions for transmission from base station.To generate the secure key for public, private, protected types we have used SHA-2 algorithm. For transmission security (data traffic encryption/decryption) we have used **Rota** algorithm. This encryption/decryption done by source selection.

The transmission logs will be maintained by Base Station. The log will contain

source/destination peer information, amount of transmitted data, routing/band width information, processing time to transmit per transmission(s).The generated peers will be recognized as *pi(1) – pi(n-1)* where i belongs to level of the peers.

Once the base station chooses the number of peers to initiate/cast to adhoc network, the peers will be dynamically placed in network which will be tracked or controlled by base station. These peers will be in the top level which is closed to base station (BS).

Whenever the new peer is joined will be under current least level. The privileges will be allotted by base station for transmission. The group key will be generated instantly for every new join. The peers appear in tree structure. The peer which is selected as source will be fixed per transmission and destination nodes will be more for that assigned source.

Leaving of the peer is possible at any level with the network by base station. The entire adhoc network will be refreshed after every leaving of the node. Even assigned group key will be rekeyed and refreshed with the leaving of the peer.

Once the transmission process is started and selected nodes to transmit the base station will allot a shared key, then the transmission starts from a source to a single or multiple destinations using this shared key. Every transmission will be independent with the other i.e. there won't be any impact on the current transmission with previous or next. In order to improve the transmission security within the group we implemented two algorithms called Rota Encryption Algorithm. The source node which initiated the transmission can opt for any of these algorithms to encrypt and decrypt the data. The shared key will be uniquely generated for every transmission.

EXPERIMENT RESULTS:

Joining and leaving of casting networks, with dynamic peers:

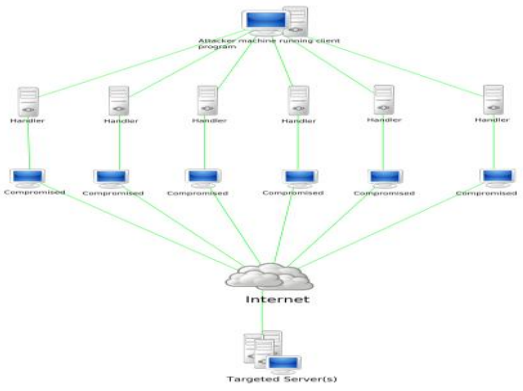
Test Case 1
Action : Data Transmission from p1 to p4 and p8
Requirements: (1) Group Key[fghjfdhkj] (2) Shared Key[jnfdjkndjg] (3)Algorithm (Rota)
Input : GK: ghjfdhkj SK: fdjhnkdvgn
Output : test pass

After this test case, node p9 left the group and a refreshed GK is generated.

Test Case 2
Action : Data Transmission from p1 to p4 and p8
Requirements: (1) Group Key[dfkjhdgnm] (2) Shared Key[jnfdjkndjg] (3) Algorithm (Rota)
Input : GK:ghjfdhkj SK: fdjhnkdvgn
Output : test Fail
Reason: the test failed 'coz the group key is not valid in this group,

Experiment-2: need to write a table Security:

- DOS/DDOS



In computing,a denial-of-service attack (DoS attack) or distributed denial-of-service attack (DDoS attack) is an attempt to make a

machine or network resource unavailable to its intended users. Although the means to carry out, motives for, and targets of a DoS attack may vary, it generally consists of the efforts of one or more people to temporarily or indefinitely interrupt or suspend services of a host connected to the Internet.

Perpetrators of DoS attacks typically target sites or services hosted on high-profile web servers such as banks, credit card payment gateways, and even root name servers. The term is generally used relating to computer networks, but is not limited to this field; for example, it is also used in reference to CPU resource management.

One common method of attack involves saturating the target machine with external communications requests, such that it cannot respond to legitimate traffic, or responds so slowly as to be rendered effectively unavailable. Such attacks usually lead to a server overload. In general terms, DoS attacks are implemented by either forcing the targeted computer(s) to reset, or consuming its resources so that it can no longer provide its intended service or obstructing the communication media between the intended users and the victim so that they can no longer communicate adequately.

Denial-of-service attacks are considered violations of the IAB's Internet proper use policy, and also violate the acceptable use policies of virtually all Internet service providers. They also commonly constitute violations of the laws of individual nations.

Methods of attack in DOS:

A "denial-of-service" attack is characterized by an explicit attempt by attackers to prevent legitimate users of a service from using that service. There are two general forms of DoS attacks: those that crash services and those that flood services.

A DoS attack can be perpetrated in a number of ways. The five basic types of attack are Consumption of computational resources, such as bandwidth, disk space, or processor time.

1. Disruption of configuration information, such as routing information.
2. Disruption of state information, such as unsolicited resetting of TCP sessions.
3. Disruption of physical network components.
4. Obstructing the communication media between the intended users and the victim so that they can no longer communicate adequately.

A DoS attack may include execution of malware intended to Max out the processor's usage, preventing any work from occurring.

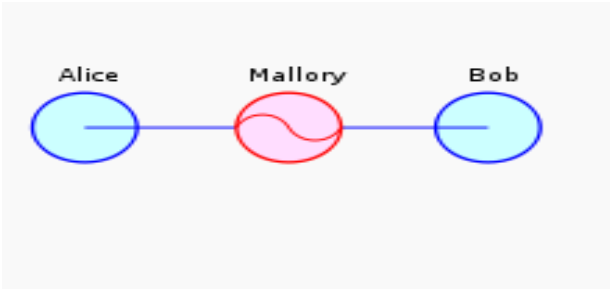
- Trigger errors in the microcode of the machine.
- Trigger errors in the sequencing of instructions, so as to force the computer into an unstable state or lock-up.
- Exploit errors in the operating system, causing resource starvation and/or thrashing, i.e. to use up all available facilities so no real work can be accomplished.
- Crash the operating system itself.

- MIM (Man in the Middle)

The man-in-the-middle attack (often abbreviated MITM, MitM, MIM, MiM,also known as a bucket brigade attack, or sometimesjanus attack) in cryptography and computer security is a form of active eavesdropping in which the attacker makes independent connections with the victims and relays messages between them, making them believe that they are talking directly to each other over a private connection, when in fact the entire conversation is controlled by the attacker. The attacker must be able to intercept all messages going between the two victims and inject new ones, which is straightforward in many circumstances (for example, an attacker within reception range of an unencrypted Wi-Fi wireless access point, can insert himself as a man-in-the-middle).

A man-in-the-middle attack can succeed only when the attacker can impersonate each endpoint to the satisfaction of the other — it is an attack on mutual authentication (or lack thereof). Most cryptographic protocols include some form of endpoint authentication specifically to prevent MITM attacks. For example, SSL can authenticate one or both parties using a mutually trusted certification authority. The Illustration of man-in-the-middle attack is below.

Example of an attack



Suppose Alice wishes to communicate with Bob. Meanwhile, Mallory wishes to intercept the conversation to eavesdrop and possibly deliver a false message to Bob.

First, Alice asks Bob for his public key. If Bob sends his public key to Alice, but Mallory is able to intercept it, a man-in-the-middle attack can begin. Mallory sends a forged message to Alice that claims to be from Bob, but instead includes Mallory's public key.

Alice, believing this public key to be Bob's, encrypts her message with Mallory's key and sends the enciphered message back to Bob. Mallory again intercepts, deciphers the message using her private key, possibly alters it if she wants, and re-enciphers it using the public key Bob originally sent to Alice. When Bob receives the newly enciphered message, he believes it came from Alice.

1. Alice sends a message to Bob, which is intercepted by Mallory:

Alice*"Hi Bob, it's Alice. Give me your key"*-->MalloryBob

2. Mallory relays this message to Bob; Bob cannot tell it is not really from Alice:

AliceMallory*"Hi Bob, it's Alice. Give me your key"*-->Bob

3. Bob responds with his encryption key:

AliceMallory<--*[Bob's_key]*Bob

4. Mallory replaces Bob's key with her own, and relays this to Alice, claiming that it is Bob's key:

Alice<--*[Mallory's_key]*MalloryBob

5. Alice encrypts a message with what she believes to be Bob's key, thinking that only Bob can read it:

Alice*"Meet me at the bus stop!"[encrypted with Mallory's key]*-->MalloryBob

6. However, because it was actually encrypted with Mallory's key, Mallory can decrypt it, read it, modify it (if desired), re-encrypt with Bob's key, and forward it to Bob:

AliceMallory*"Meet me at 22nd Ave!"[encrypted with Bob's key]*-->Bob

7. Bob thinks that this message is a secure communication from Alice.

This example shows the need for Alice and Bob to have some way to ensure that they are truly using each other's public keys, rather than the public key of an attacker. Otherwise, such attacks are generally possible, in principle, against any message sent using public-key technology. Fortunately, there are a variety of techniques that help defend against MITM attacks.

Conclusion:

Secure Group Management would explain us to maintain the security relations between peer in manet. Here we can permit the nodes by providing the Root key for adding node and for removing nodes. The region of wireless nodes is communicated by transmitting the data with encryption format by using the advanced Algorithms what we had implemented in this paper. While implementation we got best results and performance.

References:

[1] A. Perrig, "Efficient Collaborative Key Management Protocols for Secure Autonomous Group Communication", 1999.

[2] T. Dunigan and C. Cao, "Group Key Management", 1998.

[3] A. Ballardie, "Scalable Multicast Key Distribution", 1996.

[4] R. Canetti and B. Pinkas, "A Taxonomy of Multicast Security Issues", 1998.

[5] M. Waldvogel, G. Caronni, D. Sun, N. Weiler and B. Plattner, "The VersaKey Framework: Versatile Group Key Management", 1999.

[6] G. Caronni, M. Waldvogel, D. Sun and B. Plattner, "Efficient Security for Large and Dynamic Multicast Groups", 1998.

[7] I. Chang, R. Engel, D. Kandlur, D. Pendarakis and D. Saha, "Key Management for Secure Internet Multicast using Boolean Function Minimization Techniques", 1999.

[8] T. Hardjono, M. Baugher and H. Harney, "Group Security Association (GSA) Management in IP Multicast", 2000.

[9] O. Rodeh, K. Birman and D. Dolev,

"Using AVL Trees for Fault Tolerant Group Key Management", 2000.

[10] B. Bhargava, S. B. Kamisetty and S. K. Madria, "Fault-tolerant Authentication and Group Key Management in Mobile Computing", 2000.

[11] SCIM, 2002.

[12] P-C Cheng, J. A. Garay, A. Herzberg and H. Krawczyk, "Design and Implementation of Modular Key Management Protocol and IP Secure Tunnel on AIX", 1995