MODIFIED TRUSTED ON DEMAND ROUTING PROTOCOL FOR SECURE SPONTANEOUS WIRELESS AD-HOC NETWORK

Rojith Thomas[1], Sreeja J. Kumar[2], Abhiram S[3], Aneesya C[4]
PG Scholars, Communication Engineering, Caarmel Engineering College, Pathanamthitta, Kerala, India.

Abstract— This paper presents a modified trusted on demand routing protocol for spontaneous wireless ad-hoc networks which uses asymmetric/symmetric scheme and a secured trust between the users for exchanging data and to exchange the public/private keys that will be used to encrypt the data. Trust is established when a user or mobile node joins a network or during the first contact between the user. Our proposal is a self-configured holistic protocol that is able to create the network and also share the secure services between users without any pre-defined infrastructure. This protocol allows network users to share resources, offer services among users in a highly secured environment and also operate without any control from external resources or from a server. Our proposal has been implemented by using NS2. Finally, we compare the protocol with other traditional on demand routing protocol in order to highlight its features. Our results show that the cumulative utilities of cooperative nodes are increased steadily. The Modified Trust Scheme evaluates the behavior of all nodes by establishing a trust value for each node in the network that represents the trustworthiness of each one thereby identifies and eliminates the malicious nodes. It also observes number of packets generated and forwarded by the neighboring nodes, number of packets received and sent by the nodes and also end to end delay of the packet.

Index Terms— Spontaneous networks, Wireless ad-hoc networks, On demand routing protocol, Secured trust.

I. INTRODUCTION

In recent years, there is a booming growth in the field of wireless networks and telecommunications and which is mainly due to the mobility offered to the users, providing access to information on everywhere, easy installation and user friendly atmosphere. And also, flexibility and mobility provides it more popularity and increases users numbers and efficiency of system. Spontaneous network is a newly emerged idea with which a set of mobile terminals that are placed in a close location communicating with each other, sharing resources or services during a limited period of time and in a limited space, as like human interaction [1]. Peoples are attached to a group or meeting for a while and after that they leave like that, so the management of network should be flexible for the user. These types of networks usually have independent or haven't a centralized administration. Spontaneous network can be wired or wireless. Here we are dealing about wireless spontaneous ad-hoc networks and which needed well-defined, effective and user-friendly security mechanisms. Because these networks are generally implemented in devices such as laptops or mobile phones and so we have to provide a holistic flexible and secure protocol to manage, control and integrate them. Spontaneous networks have an ability to adapt to new situations and during occurrence of fault in the system i.e. they act as human. A service configuration in spontaneous network depends on size of the network, number of users in the network and services to provide for users. Spontaneous ad-hoc networks require a well-defined, efficient and secured user friendly protocol. Different functions that have to be performed by the protocol are identifying a user, provide authorization to newer nodes, assignment of addresses, providing services to users and ensure safety to the network. The salient features of spontaneous networks are:

- Poorly defined network boundaries.
- Not have a predefined infrastructure and preconfigured hosts.
- Not have a Centralized Authority [CA].
- No need for users to be experts.

Generally, in wireless network setup, we use Certificate Authority (CA) servers to manage node trust and authentication [2]. Also these systems have been used in wireless ad-hoc networks. But it creates some practical limitations like difficulty to manage the CA node online every time and also CA node must have higher computing capacity. In such networks, the key share mechanisms for node authorization and user authentication are needed to achieve a dependable communication and node authorization in ad hoc networks. So here we introduced a secured self-configured environment for data distribution, resources sharing and services sharing among users by using an efficient and secured routing protocol and network. Here the security is based on the users service needs. Also to obtain a distributed certification authority, it is necessary to build a trusted network. The network allows users to join because it belongs to someone who knows it. Hence, the new user is trusted by the certification authority. This allows the network to have a distributed service and also distribution of network management. We are used to apply asymmetric cryptography, where each device has a public-private key pair for device identification and symmetric cryptography to share session keys between nodes. There are also some unidentified users present in the network since authentication, validity and privacy is based on the user identification. A major issue in spontaneous ad- hoc network is security. The security mechanism for network must require low computational complexity and small number of appended messages to save the node energy. It should also be competitive and effective in preventing misbehaviors or identifying misbehaving nodes from normal ones. Two approaches in protecting spontaneous ad- hoc networks are:

- Proactive Protocol: Traditional distributed shortest-path protocols, based on periodic updates. It has high routing overhead and also prevents an attacker from launching attacks through various cryptographic schemes.
- Reactive Protocol: Seeks to detect security threats and react accordingly. Discover routes when needed. Source-initiated route discovery, because of this we go for reactive protocol.

The two types of reactive protocol are Dynamic Source Routing (DSR) and Ad-Hoc On Demand Distance Vector Routing (AODV). Both are routing protocols for wireless mesh/ad hoc networks. They are demand-driven protocols which form a route on demand when a transmitting computer desires a route. DSR is based on source routing in which all the routing information is maintained at the mobile nodes. The DSR computes the routes and also updates them. The main drawbacks in this are:

- Maintains additional table entries, causing a larger memory overhead.
- Not capable of handling congestion.
- It does not remove the broken path, hence routing is time consuming.
- Routing packets are large.
- Relatively small network diameter.

Therefore we go for AODV protocol. The AODV uses a combination of a DSR and DSDV [3] mechanism. It uses the route discovery and route maintenance from DSR, hop-by-hop routing, periodic advertisements, and sequence numbers from DSDV. The AODV easily overcomes the counting to infinity and Bellman Ford problems, and it also provides quick convergence whenever the ad-hoc network topology is altered. In our work to be described in the thesis, we focused on designing a secure routing mechanism for spontaneous networks in a self- organized way instead of using centralized servers since these centralized servers or trusted parties make the network more controllable but they destroy the self-organizing nature of spontaneous networks and reduce the network scalability. Our solution is to introduce the idea of “trust” to solve this problem. Based on this trust model, we design our secure routing protocol for spontaneous ad-hoc networks according to Ad-hoc On-demand Distance Vector (AODV) routing protocol. The new protocol, called MTAODV (Modified Trusted AODV), has several salient features:

- Nodes perform trusted routing behaviors mainly according to the trust relationship among them.
- A node which performs malicious behaviors will eventually be detected and denied to the whole network.
- System performance is improved by avoiding requesting and verifying certificates at every routing step.

The rest of paper is organized as follows: Section II presents the AODV routing protocols and security mechanisms that can be applied to them. Section III describes about secured spontaneous network in details. Different stages in network creation are detailed in Section IV. Section V explains about experimental setup and comparing proposed protocol with existing one. Performance analysis is shown in Section VI and finally, Section VII gives the conclusion and future work.

II. AODV ROUTING PROTOCOLS

A. Secure Ad-hoc on Demand Distance Vector Routing

SAODV is a modified version of AODV. It uses asymmetric cryptography for routing messages and uses Digital Signatures to protect the non-mutable data in the RREQ (Route Request) and RREP (Route Reply) messages. The four basic operations performed for the route establishment are route discovery, route request, route reply and route maintenance. Before entering the network, each node obtains a public key certificate from a trusted certificate server. There are end-to-end authentication between source and destination and hop-to-hop authentication between intermediate nodes.

Hash chains are used in this to authenticate the hop count of the AODV routing. Source broadcasts signed RDM (Route Discovery Message) [4] along with its own certificate. RDM contains the source IP address, along with a source-specific nonce (to detect duplicates). First hop adds its own signature and certificate. Each hop verifies signature of previous hop and replaces it with its own signature. It also adds a reverse route to source. Destination also verifies the source signature. In route reply the destination sends back a signed reply (RRM) to the first RDM.

The discovered route may not be the shortest, but is the “quickest”. Route Maintenance Nodes send signed error messages (RERR) to indicate link breaks, and packets arriving on deactivated paths. Hop count authentication by using hash chains is not perfect since a malign node might forward a message without increasing the hop count. Tunneling attacks are not solved by SAODV. The processing power requirements of SAODV should be reduced due to the use of asymmetric cryptography.

B. Adaptive SAODV (A-SAODV)

Adaptive mechanism that tunes its behavior for optimizing the performance of routing operation is called Adaptive SAODV (A-SAODV) [5] which is a multi-threaded application. Cryptographic operations are performed by a dedicated thread to avoid blocking the processing of other message and other thread to all other functions.

Each node has to maintain a queue length field for all neighbouring node along with the list of neighbourhood nodes which they may update and exchange with the help of hello message periodically. When an intermediate node receives a RREQ and finds that it has the valid route to the destination, it check it’s time to leave field (TTL) then simply forwards RREQ only to this neighboring node, otherwise it replies to the source using method involved in SAODV. If RREQ packet is less than the TTL threshold value the request packet is simply forwarded to all neighboring nodes. This may significantly reduce the queue length of any intermediate node. The prototype also maintains a cache of latest signed and verified messages in order to avoid signing and verifying the same message twice.

C. Security Aware Ad-hoc Routing (SAR)

SAR is an approach to routing, which incorporates security levels of nodes into traditional routing metrics. The goal of SAR is to characterize and explicitly represent the trust values and trust relationships associated with ad-hoc nodes and use these values to make routing decisions. The route discovery mechanism will then find nodes that match particular security attributes and trust levels.

Only nodes that provide the required level of security can generate or propagate route requests, updates, or replies. If the node cannot provide the required security, the RREQ (Route Request) [6] is dropped. However SAR is able to find a route with guarantee of security. If one or more routes that satisfy the require security attributes exist, SAR will find the shortest such route. If all the nodes on the shortest path between two nodes can satisfy the security requirements, SAR will find routes that are optimal. Timeliness, ordering, authenticity, integrity, confidentiality are some of its properties. The main drawback in SAR is it requires excessive encryption and decryption.

D. Reliable Ad-hoc on Demand Distance Vector Routing

Reliable Ad-hoc on Demand Distance Vector Routing (RAODV) also uses RRDU (Route Discovery Request) and RRDU_REP (Route Discovery Request Reply) [6] to help discover the path and for reliability maintenance. Path discovery in RAODV [5] can be thought of as consisting of two phases. Phase I is same as that in AODV. That is, when a node wishes to communicate with another node it looks for a route in its table. If a valid entry is found for the destination it uses that path else the node broadcasts the RREQ to its neighbors to locate the destination.

The process continues until either the destination or an intermediate node with a fresh route to the destination is located. At each intermediate node, a reverse path is created for the source. The source receives RREPs from all these paths In Phase II the source node sends an RRDU packet to all the nodes from which it gets the RREPs. Now since replies to RRDU, i.e. RRDU_REP packets are generated only by the destination and there is no impersonation, the source node will receive a unique RRDU_REP and the path discovery is completed.

III. SPONTANEOUS NETWORK CREATION

Our protocol is a secure protocol for routing purposes, based on trust that allows the creation and management of decentralized spontaneous networks with little intervention from the user and integration of different devices. The services and network scenario provided for users must be volatile so that user can free to join and leave the network. This section deals about the creation of spontaneous network and following steps should be completed for creation of network.

A. Nodes Joining Procedure

In this step we are enabling devices to communicate, including the automatic configuration of logical and physical parameters. The system is based on the use of an IDentity Card (IDC) and a certificate. An IDC contains public and private components, Logic IDentity (LID), which is unique for each node and which allows other nodes to identify it. It may include information such as name, photograph or other type of user identification. This LID is the main portion of public component. It also contains the user's public key (Ki), the creation and expiration dates, an IP proposed by the user, and the user signature.

The user signature is generated by using Secured Hashing Algorithm on the previous data to obtain the data summary. Then, the data summary is signed with the user's private key. The private component contains the private key (ki). The user introduces its personal data (LID) the first time he/she uses the system because the security information is generated then. Security data are stored persistently in the device for future use. Certificate Cij of the user i consists of a validated IDC, signed by a user j that gives its validity. To obtain IDC signature of user i, the summary function obtained by SHA- 1 is signed with j's private key. No central certification authority is used to validate IDC. Validation of integrity and authentication is done automatically in each node.

The certification authority for a node could be any of the trusted nodes. This system enables us to build a distributed certification authority between trusted nodes. When node A wants to communicate with another node B and it does not have the certificate for B, it requests it from its trusted nodes. After obtaining this certificate the system will validate the data, if correct then it will sign this node as a valid node. All nodes can be both clients and servers, can request or serve requests for information or authentication from other nodes. The first node creates the spontaneous network and generates a random session key, which will be exchanged with new nodes after the authentication phase. Figure 1 shows phases of a node joining the network, node authentication and authorization, agreement on session key, transmission protocol and speed, IP address and routing. When node B wants to join an existing network, it must choose a node within communication range to authenticate with (e.g., node A). A will send its public key. Then, B will send its IDC signed by A's public key. Next, A validates the received data and verifies the hash of the message in order to check that the data has not been modified. In this step, A establishes the trust level of B by looking physically at B (they are physically close), depending on whether A knows B or not.

Finally, A will send its IDC data to B (it may do so even if it decides not to trust B). This data will be signed by B's public key (which has been received on B's IDC). B will validate A's IDC and will establish the trust and validity in A only by integrity verification and authentication.

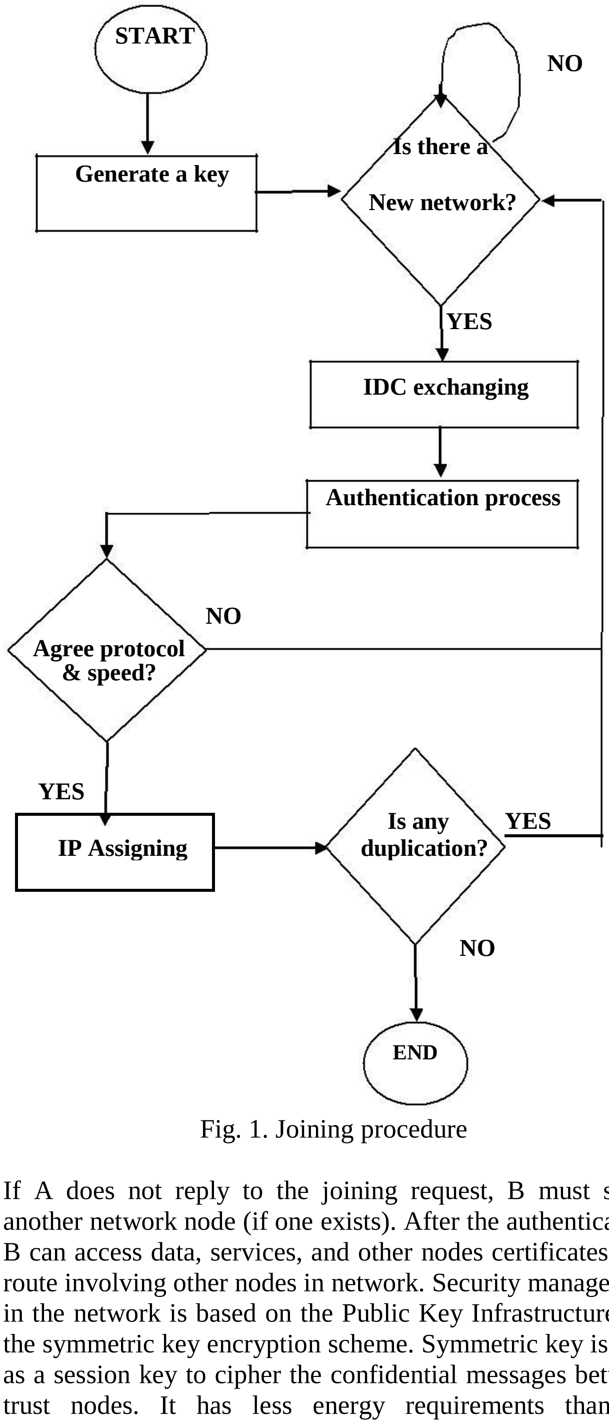


Fig. 1. Joining procedure

If A does not reply to the joining request, B must select another network node (if one exists). After the authentication, B can access data, services, and other nodes certificates by a route involving other nodes in network. Security management in the network is based on the Public Key Infrastructure and the symmetric key encryption scheme. Symmetric key is used as a session key to cipher the confidential messages between trust nodes. It has less energy requirements than the asymmetric key [7]. We have used the Advanced Encryption Standard (AES) algorithm for the symmetric encryption scheme [8]. It offers high security because its design structure removes subkey symmetry. Moreover, execution times and energy consumption in cryptography processes are adequate for low-power devices. The asymmetric key encryption scheme is used for distribution of the session key and for the user authentication process. We used the Rivest, Shamir & Adleman cryptographic algorithm (RSA) [9] for asymmetric encryption scheme. After the mutual authentication, A will encrypt the session key with B's public key and will send it to B. Then, they will agree the transmission protocols and the wireless connection speed. Finally, B will configure IP address and routing information. B generates an IP address which has a fixed part in the first two bytes and the rest is formed by a random number which depends on the user's data. Then, B will send the data to process the routing information to A. A will check whether the IP is duplicated in the network. When B sends data to other network nodes, e.g., node C, these data will be validated by C (using hashing and authentication methods). Afterwards, C will establish the trust level with B, by looking physically. If no trust level is established, it will be done afterwards by using trusted chains.

B. Discovering Services

If B asks for the available services. Services can be discovered using Web Services Description Language (WSDL). A user can ask other devices in order to know the available services. It has an agreement to allow access to its services and to access the services offered by other nodes. Services have a large number of parameters which are not transparent to the user and require manual configuration. One issue is to manage the automatic integration tasks and use, for example, service agents. Other is to manage secure access to the services offered by the nodes in the network. The fault tolerance of the network is based on the routing protocol used to send information between users. Services provided by B are available only if there is a path to B, and disappear when B leaves the network.

C. Trust Establishing

There are only two trust levels in the system. Node A either trusts or does not trust another node B. The software application installed in the device asks B to trust A when it receives the validated IDC from B. Trust relationship can be asymmetric. If node A did not establish trust level with node B directly, it can be established through trusted chains, e.g., if A trusts C and C trusts B, then A may trust B. Trust level can change over time depending on the node's behavior. Thus, node A may decide not to trust node B although A still trusts C and C trusts B. It can also stop trusting if it discovers that previous trust chain does not exist anymore.

IV. NETWORK MANAGEMENT AND PROTOCOL IMPLEMENTATION

In the network formation, nodes perform an initial exchange of configuration information and security using the mechanism of authentication. This mechanism avoids the need for a central server, making the tasks of building the network and adding new members very easy. The network is created using the information provided by users, thus, each node is identified by an IP address. Services are shared using UDP connections. The network is built using IEEE 802.11b/g technology which has high data rates to share resources. After the authentication process, each node learns the Identity card of other known nodes, a public key and a LID. This information will be updated and completed throughout the network nodes. This structure provides an authenticated service that verifies the integrity of the data from each node because there is a distributed CA. Each node requests the services from all the nodes that it trusts, or from all known nodes in the network, depending on the type of service. A request to multiple nodes is made through diffusion processes. The protocol prioritizes access to information through trusted nodes. When the information cannot be obtained through these nodes, it can then ask other nodes. Nodes can also send requests to update network information. The reply will contain the identity cards of all nodes in the network. The node replying to this request must sign this data ensuring the authenticity of the shipment. If it is a trusted node, its validity is also ensured, since trusted nodes have been responsible for validating their previous certificates. Under this network, any type of service or application can be implemented. The services offered by our protocol will be secure.

A. Creation of New Network

The first node in the network will be responsible for setting the global settings of the spontaneous network (SSID, session key). However, each node must configure its own data (including the first node): IP, port, data security, and user data. This information will allow the node to become part of the network. After this data are set in the first node, it changes to standby mode.

1) Joining New Members: The second node first configures its user data and network security. Then, the greeting process starts. It authenticates against the first node. Our protocol relies on a sub layer protocol. The connection is created through a short-range link technology, to provide flexibility and ease of detection and selection of nodes, and visual contact with the user of the node. Furthermore, minimal involvement of the user is required to configure the device, mainly to establish trust. This technology also limits the scope and the consumption of involved nodes. Each additional node authenticates with any node in the network.

B. Protocol Operation

Once the validation/registration process of the user in the device has been done, he/she must determine whether to create a new network or participate in an existing one. If he/she decides to create a new network, it begins the procedure shown in Figure 2. First, a session key will be generated. Then, the node will start its services (including the network and authentication services). Finally, it will wait for requests from other devices that want to join the network. If the user wants to become part of an existing network, the node follows algorithm from earlier section to find a device that will give trust to it, save corresponding data and will able to begin communications. The node that belongs to the network, and is responsible for validating the new node's data [10], will perform a diffusion process to the nodes that are within its communication range. These nodes will forward the received packets to their neighbours until the data reach all nodes in the network. This process allows verifying the validity and uniqueness of the new node's data. The authentication process for new device B is shown in Figure 3. The receiver node A validates the received data and sends a broadcast message to B to check if these data are not used in the network (even the IP address). This IP checking packet is sent randomly twice in order to avoid simultaneous checks and reach all devices. When the authentication device receives the IP checking reply, it sends the authentication reply to the new device. If any step is wrong, an error message is sent to the new device. When the node is authenticated, it is able to perform several tasks. Some of them are performed transparently for the user, but others are used by the user to perform some operations in the network. They are the user application options.

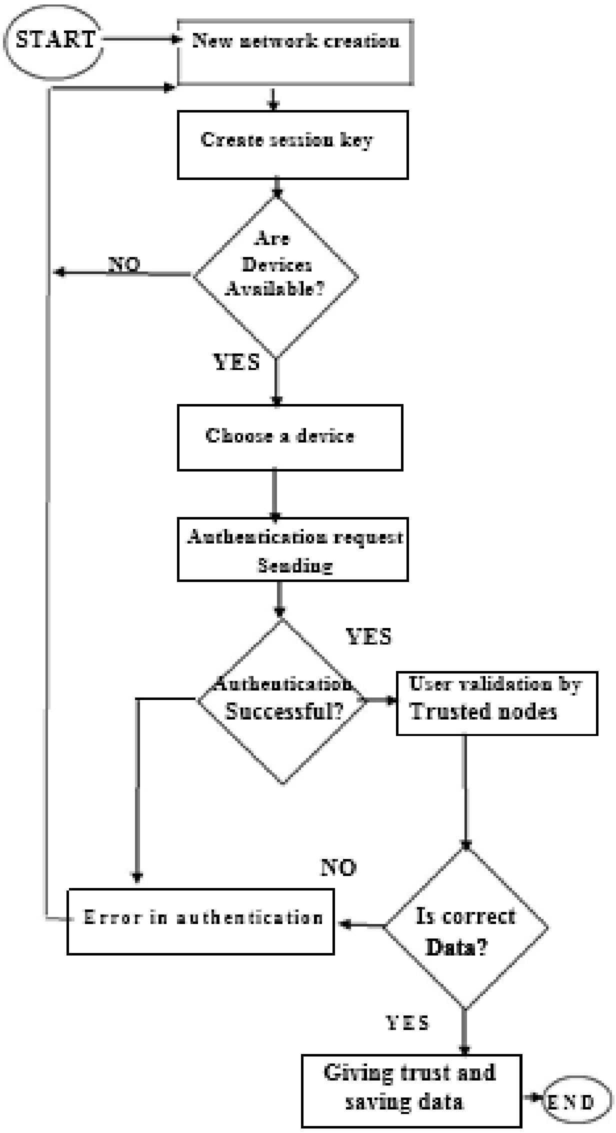


Fig. 2. Creation of new network

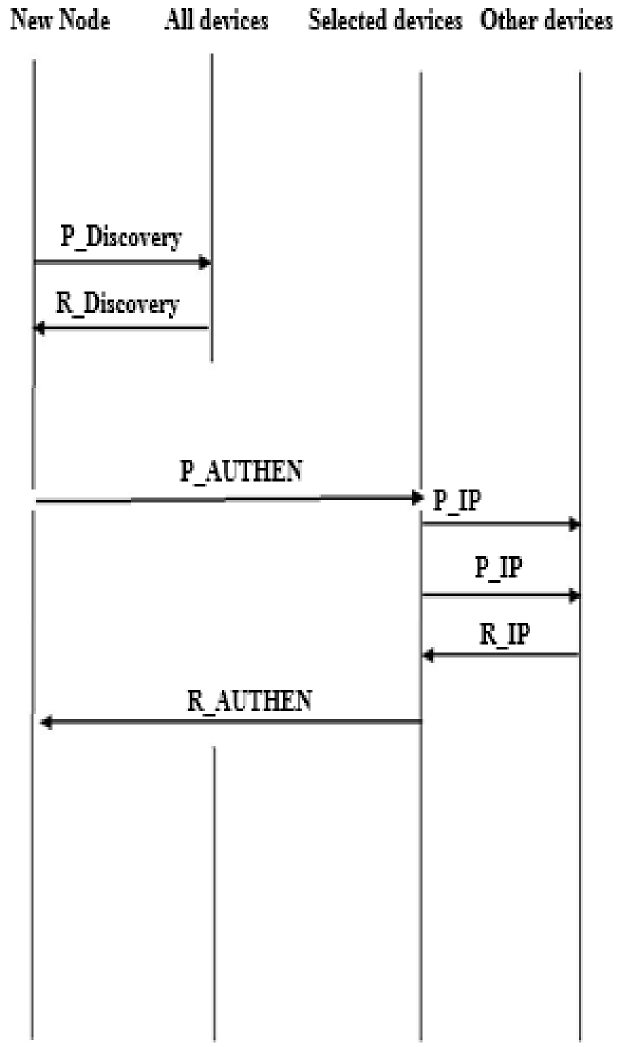


Fig. 3. Procedure for authentication

The authenticated node can perform the following tasks:

- Display the nodes.
- Modify the trust of the nodes.
- Update the information: It allows a node to learn about other nodes in the network and also to send its data to the network. This update could be for only one user or for all users in the network through a controlled diffusion process.
- Other nodes certificate request: A node could be requested from other node, from all nodes or from all known nodes. In case of all known nodes, the node that replies trusted to the request will always sign the data. The data will be considered validated if a trusted node has signed them.
- Process an authentication request: The node authenticates a requesting node by validating the received information, user authentication, and verifying the non-duplication of the LID data and the proposed IP.

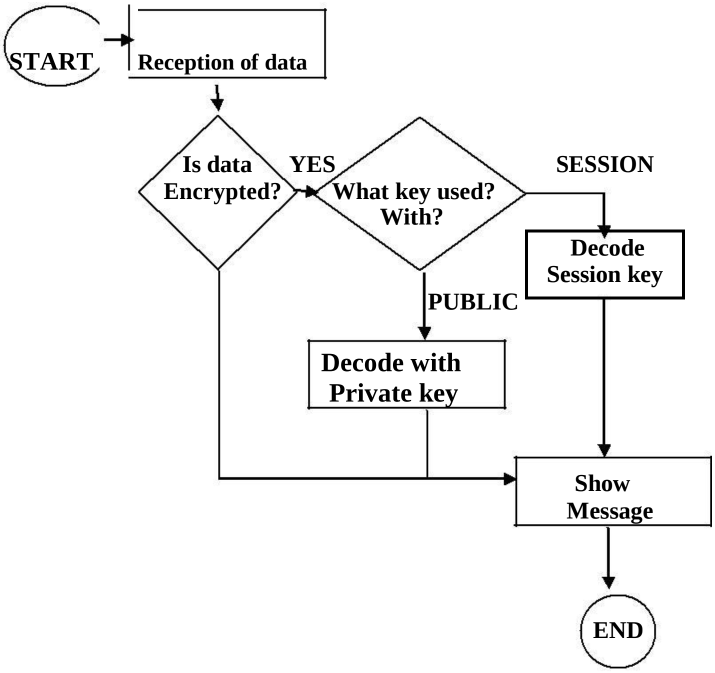


Fig. 4. Packet reception algorithm

- Reply to an information request: the requested information will be sent directly to the requesting node or routed if the node is not on the communication range.
- Forward an information request: The request will be forwarded if it is a broadcast message.
- Send data to one node: It can be sent symmetrically or asymmetrically encrypted, or decrypted.
- Send data to all nodes: This process is doing by a flooding system. Each node retransmits the data only the first it receives the data. It can be sent symmetrically encrypted or unencrypted.
- Modify Data: User data can be modified and the password changed.
- Leave the network.

To request a certificate, the node sends a request certificate message to its trusted nodes. The application generates a packet to request the certificate to its trust nodes which are selected from the database. To process the received request, the node checks if it can reply to the request, if not, the node sends the search to other nodes (that it trusts or known nodes). Then, the node has to validate the certificate and sends it to the requesting node. When the server process receives the packet, it processes the packet in order to take the certificate and checks its validity access to the certificate data.

To send data encrypted with the public key to a node, the user has to select the remote node and write the data. Then, the message is encrypted using the remote node’s public key. The application encrypts the data with the public key, generates the packet and sends it to the selected node. Each node has to check every received data packet.

TABLE I. PROT OCOL PACKET S

ID	PACKET NAME	DESCRIPTION
01	P_DISCOVERY	Discovery request
02	R_DISCOVERY	Discovery reply
03	P_AUTHENT	Authentication request
04	R_AUTHENT	Authentication reply
05	P_IP	IP checking
06	R_IP	IP checking reply
07	P_ACTUALIZA	Update node request
08	R_ACTUALIZA	Update reply request
09	P_BROADCAST	Update request to all network nodes
10	R_BROADCAST	Update reply to all network nodes
11	P_NODO_CONF	Certificate request to trusted nodes
12	P_NODO	Certificate request to known nodes
13	R_NODO	Certificate reply
14	P_DATAS	Packet for sending data
15	P_ERROR	Error
16	P_ACK	Acknowledge

If the received packet is not encrypted, it is shown directly to the user, but if it is encrypted, the packet will be deciphered using the encryption model used by the sender. The algorithm followed by the node is shown in Figure 4.

C. Protocol Implementation

We have developed 16 packets for the proper running of the protocol. Table I shows these packets. Some of them have been shown in Figure 3. When a device wants to join a spontaneous network it has to start the process by sending a Discovery request packet (01), which contains the Logical IDentity of the user in order to let the destinations know the sender device. The receivers will reply with the Discovery reply packet (02) with their Logical IDentity, their IP address, and network mask. This information is then used to learn the selected device to authenticate and to propose an IP inside that network IP range. The authentication request packet (03) is used for the new device authentication. The authentication reply packet (04) confirms that the proposed IP and email are unique in network, so new device is officially authenticated. In case of duplication, an error packet is sent. The IP and e-mail checking packet (05) is used by the authenticator device to verify that no one in the network has the same email or IP address as the one proposed by the new device.

The IP and email checking reply packet (06) is sent to authenticator device in order to verify that the IP and email are unique. If the IP is duplicated, the device must restart the authentication process after the generation of a new IP.

The update request to one node (07) is used to request information to a specific known node and the update reply from one node (08) is used to reply with the information requested by the update request packet to one node. Unknown information can be requested from all nodes in the network by sending the update request to all network nodes packet by flooding (09) the reply with the information requested is called update reply to all network nodes packet (10). The Certificate request to trusted nodes (11) and the Certificate request to known nodes (12) are used to request the certificate from all trusted and all known nodes, respectively. Both packets are replied to by the certificate reply packet (13). Data are sent using the Packet for sending data (14). This packet is sent when the user decides to communicate with one or more nodes. These data could be sent in explain or encrypted text. The error packet (15) can be sent to indicate that this operation is not possible, because the authentication has failed, or because the node does not have the required data. The acknowledge packet (16) is used to confirm to the sender that the packet has arrived at its destination correctly.

TABLE II. HARDWARE AND SOFT WARE DESCRIPTION

HARDWARE USED	
PROCESSOR	INTEL CORE 2 DUO
PROCESSOR SPEED	1.5 GHz
RAM	3 Gb
HARD DISK	320 Gb
SOFTWARE USED	
OPERATING SYSTEM	UBUNDU 12.04 LTS
LANGUAGES	TCL SCRIPT,C++
SOFTWARE	ns2.34

V. EXPERIMENTAL SETUP AND COMPARISON

In this section we are going to analyzing proposed system with existing Ad-hoc on demand routing protocol. Before that we just explaining our experimental set up. We used ns 2.35 for simulating the network and Table II shows the software and hardware setup details.

Implementation study begins with simulation of Network Environment. This requires setting of simulation network parameters. These parameters are depicted in the Table III. The simulation results from running the script in NS-2 include one or more text based output files and an input to a graphical simulation display tool called Network Animator (NAM).

TABLE III. SIMULAT ION PARAMET ERS

Parameter	Values
Examined Protocols	AODV,MTAODV
Traffic Type	UDP
Transmission Area	500X500 m²
Packet Size	512 bytes
Data Rate	600 kb/s
SImulation Time	60s

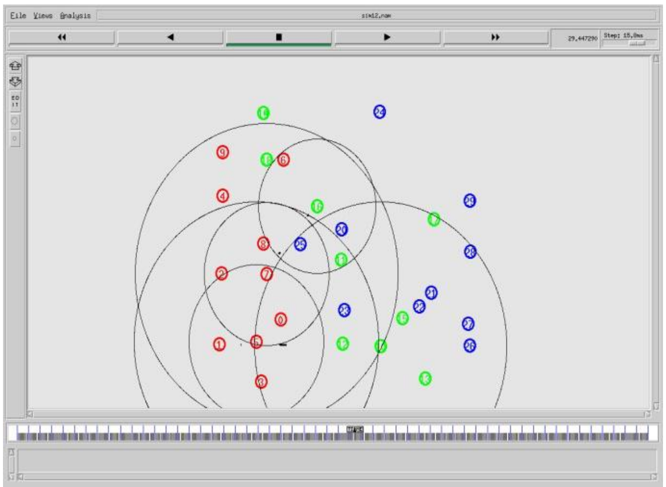
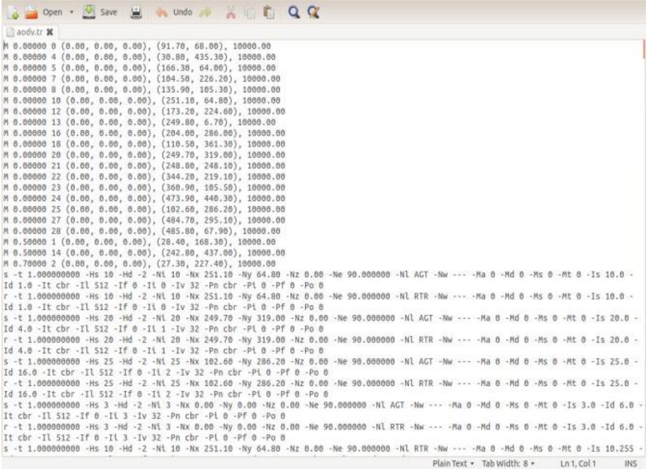


Fig. 5. Screen shot of NAM window



```
M 0.00000 0 (0.00, 0.00, 0.00), (91.70, 68.00), 10000.00
M 0.00000 4 (0.00, 0.00, 0.00), (30.80, 435.30), 10000.00
M 0.00000 5 (0.00, 0.00, 0.00), (166.30, 64.00), 10000.00
M 0.00000 7 (0.00, 0.00, 0.00), (104.50, 226.20), 10000.00
M 0.00000 8 (0.00, 0.00, 0.00), (135.90, 105.30), 10000.00
M 0.00000 10 (0.00, 0.00, 0.00), (251.10, 64.80), 10000.00
M 0.00000 12 (0.00, 0.00, 0.00), (173.20, 224.60), 10000.00
M 0.00000 13 (0.00, 0.00, 0.00), (249.80, 6.70), 10000.00
M 0.00000 16 (0.00, 0.00, 0.00), (204.00, 286.00), 10000.00
M 0.00000 18 (0.00, 0.00, 0.00), (110.50, 361.30), 10000.00
M 0.00000 20 (0.00, 0.00, 0.00), (249.70, 319.00), 10000.00
M 0.00000 21 (0.00, 0.00, 0.00), (248.80, 248.10), 10000.00
M 0.00000 22 (0.00, 0.00, 0.00), (344.20, 219.10), 10000.00
M 0.00000 23 (0.00, 0.00, 0.00), (360.90, 105.50), 10000.00
M 0.00000 24 (0.00, 0.00, 0.00), (473.90, 440.30), 10000.00
M 0.00000 25 (0.00, 0.00, 0.00), (102.60, 286.20), 10000.00
M 0.00000 27 (0.00, 0.00, 0.00), (484.70, 295.10), 10000.00
M 0.00000 28 (0.00, 0.00, 0.00), (485.80, 67.90), 10000.00
M 0.50000 1 (0.00, 0.00, 0.00), (28.40, 168.30), 10000.00
M 0.50000 14 (0.00, 0.00, 0.00), (242.80, 437.00), 10000.00
M 0.70000 2 (0.00, 0.00, 0.00), (27.30, 227.40), 10000.00
s -t 1.000000000 -Hs 10 -Hd -2 -Ni 10 -Nx 251.10 -Ny 64.80 -Nz 0.00 -Ne 90.000000 -Nl AGT -Nw --- -Ma 0 -Md 0 -Ms 0 -Mt 0 -Is 10.0 -
Id 1.0 -It cbr -Il 512 -If 0 -Ii 0 -Iv 32 -Pn cbr -Pi 0 -Pf 0 -Po 0
r -t 1.000000000 -Hs 10 -Hd -2 -Ni 10 -Nx 251.10 -Ny 64.80 -Nz 0.00 -Ne 90.000000 -Nl RTR -Nw --- -Ma 0 -Md 0 -Ms 0 -Mt 0 -Is 10.0 -
Id 1.0 -It cbr -Il 512 -If 0 -Ii 0 -Iv 32 -Pn cbr -Pi 0 -Pf 0 -Po 0
s -t 1.000000000 -Hs 20 -Hd -2 -Ni 20 -Nx 249.70 -Ny 319.00 -Nz 0.00 -Ne 90.000000 -Nl AGT -Nw --- -Ma 0 -Md 0 -Ms 0 -Mt 0 -Is 20.0 -
Id 4.0 -It cbr -Il 512 -If 0 -Ii 1 -Iv 32 -Pn cbr -Pi 0 -Pf 0 -Po 0
r -t 1.000000000 -Hs 20 -Hd -2 -Ni 20 -Nx 249.70 -Ny 319.00 -Nz 0.00 -Ne 90.000000 -Nl RTR -Nw --- -Ma 0 -Md 0 -Ms 0 -Mt 0 -Is 20.0 -
Id 4.0 -It cbr -Il 512 -If 0 -Ii 1 -Iv 32 -Pn cbr -Pi 0 -Pf 0 -Po 0
s -t 1.000000000 -Hs 25 -Hd -2 -Ni 25 -Nx 102.60 -Ny 286.20 -Nz 0.00 -Ne 90.000000 -Nl AGT -Nw --- -Ma 0 -Md 0 -Ms 0 -Mt 0 -Is 25.0 -
Id 16.0 -It cbr -Il 512 -If 0 -Ii 2 -Iv 32 -Pn cbr -Pi 0 -Pf 0 -Po 0
r -t 1.000000000 -Hs 25 -Hd -2 -Ni 25 -Nx 102.60 -Ny 286.20 -Nz 0.00 -Ne 90.000000 -Nl RTR -Nw --- -Ma 0 -Md 0 -Ms 0 -Mt 0 -Is 25.0 -
Id 16.0 -It cbr -Il 512 -If 0 -Ii 2 -Iv 32 -Pn cbr -Pi 0 -Pf 0 -Po 0
s -t 1.000000000 -Hs 3 -Hd -2 -Ni 3 -Nx 0.00 -Ny 0.00 -Nz 0.00 -Ne 90.000000 -Nl AGT -Nw --- -Ma 0 -Md 0 -Ms 0 -Mt 0 -Is 3.0 -Id 6.0 -
It cbr -Il 512 -If 0 -Ii 3 -Iv 32 -Pn cbr -Pi 0 -Pf 0 -Po 0
r -t 1.000000000 -Hs 3 -Hd -2 -Ni 3 -Nx 0.00 -Ny 0.00 -Nz 0.00 -Ne 90.000000 -Nl RTR -Nw --- -Ma 0 -Md 0 -Ms 0 -Mt 0 -Is 3.0 -Id 6.0 -
It cbr -Il 512 -If 0 -Ii 3 -Iv 32 -Pn cbr -Pi 0 -Pf 0 -Po 0
s -t 1.000000000 -Hs 10 -Hd -2 -Ni 10 -Nx 251.10 -Ny 64.80 -Nz 0.00 -Ne 90.000000 -Nl RTR -Nw --- -Ma 0 -Md 0 -Ms 0 -Mt 0 -Is 10.255 -
```

Fig. 6. Screen Shot of Trace File.

It is analyzed by separate programs written in “perl” or it can also analyzed by “awk” program to extract six network performance metrics. NAM is an animation tool for viewing network simulation traces and real world packet traces. Fig 5 & 6 shows snapshot of NAM and trace file generated for a 30 node network with data rate of 600kb/s X graph is a graphical representation tool which is used for representing nodes properties with respect to the simulation time. This is attributed to only one route per destination maintained by AODV. Each packet that the MAC layer is unable to deliver is dropped since there are no alternate routes.

MTAODV allow packets to stay in the send buffer for 30 seconds for route discovery and once the route is discovered, data packets are sent on that route to be delivered at the destination. This is attributed to only one route per destination maintained by AODV. Each packet that the MAC layer is unable to deliver is dropped since there are no alternate routes. MTAODV allow packets to stay in the send buffer for 30 seconds for route discovery and once the route is discovered, data packets are sent on that route to be delivered at the destination. If route fails, MTAODV find new path within 30 seconds thereby minimizing the possibility of packet drop. As MTAODV always holds optimal paths to all other destinations in their routing, delay involved in sending data packets at lower traffic load is very less. As traffic load increases AODV perform better as it adopts hop-by-hop routing. In MTAODV at higher traffic load packet delay is higher and but which has overall delay not much more than AODV.

TABLE IV. PERFORMANCE MET RICES OF PROPOSED AND EXISTING SYST EMS.

METRICES	AODV	MTAODV
No. of Nodes	30	30
Throughput [kbps]	274.27	318.04
Send Packets	4307	4320
Received Packets	3617	4193
Routing Packets	2110	453
Packet Delivery Fraction	83.98	97.06
Packet Routing Load	0.58	0.11
End to end Delay (ms)	39.02	216
No of dropped data (packets)	1123	139
No of dropped data (bytes)	597436	73948
Mean Jitter	55.80	327.45
Current Jitter	61.51	312.34

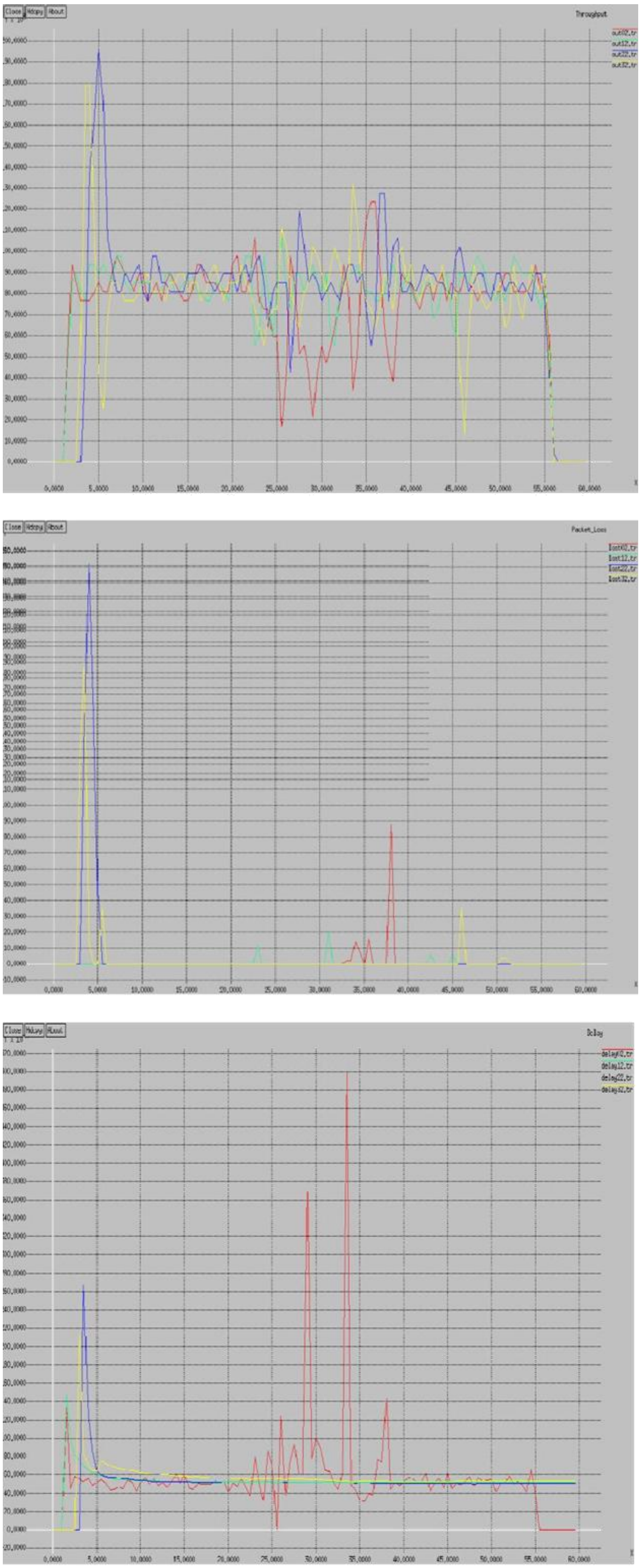


Fig. 7. Various X Graphs of throughput, packet dropped and delay of MTAODV.

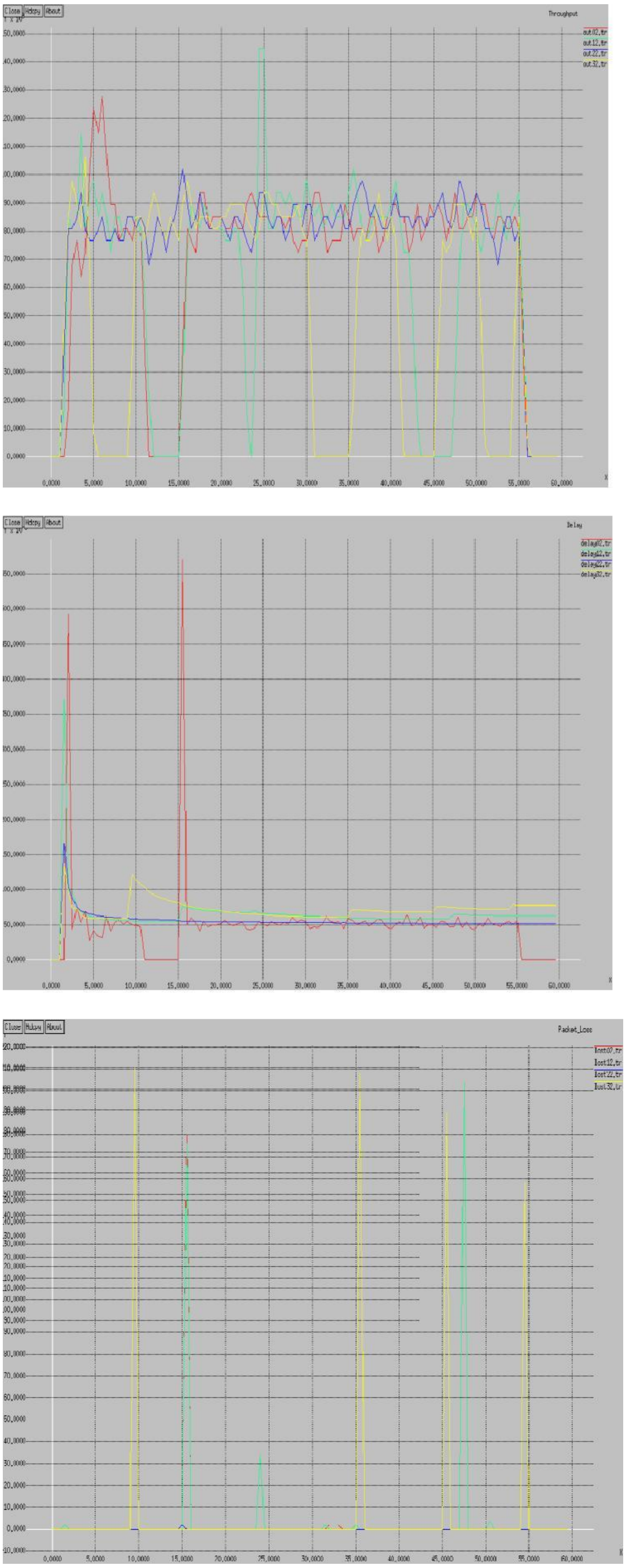


Fig. 8. Various X Graphs of throughput, packet dropped and delay of AODV.

VI. PERFORMANCE ANALYSIS

AWK script is a high level C++ program which is used to calculate performance evaluation metrics from the trace file that generated during the simulation of a network. In AWK script the input parameters and variables is from the trace file and by using this it will calculate the overall performance evaluation metrics of a network. As by using AWK script, we got the performance matrices as shown in Table IV. Now we simulate networks for different number of nodes to find out the performance of our proposed system under various routing conditions and we plot it as graph. Figure 10 shows comparison of throughput, packet delay fraction, end to end delay and normalized routing load of proposed and existing system. On comparing with AODV, MTODV has higher throughput and which means that it has high mobility at high routing loads and by increasing number of nodes the throughput will increases. Similarly packet delivery ratio is the number of packet received to the number of packet sent. Here comparing with AODV, MTAODV has higher packet delivery fraction which has almost near to 97 in some cases. While comparing with AODV, AODV has lower delay of packet transmission from source to destination. But in the case of higher number of nodes, the delay is same as that of AODV. Also, periodic broadcasts of control packets of AODV increase routing load in the network and hence AODV has more routing overhead irrespective traffic load. Further this worsens with increasing number of node. Even understand still condition of network AODV keeps on sending periodic updates at regular intervals among the nodes. MTAODV performs better at high traffic since it computes route as and when needed and it adapts hop-by-hop routing. Now figure 9 shows the routing protocol performance during intruders attacks. Intruders are the attacker which enters to the network and during the data transmission, the packet that received to the attacker nodes will not forward to destination and finally the packet will drop and make the network collapse. Here MTAODV will provide much better performance evaluation metrics values than traditional AODV.

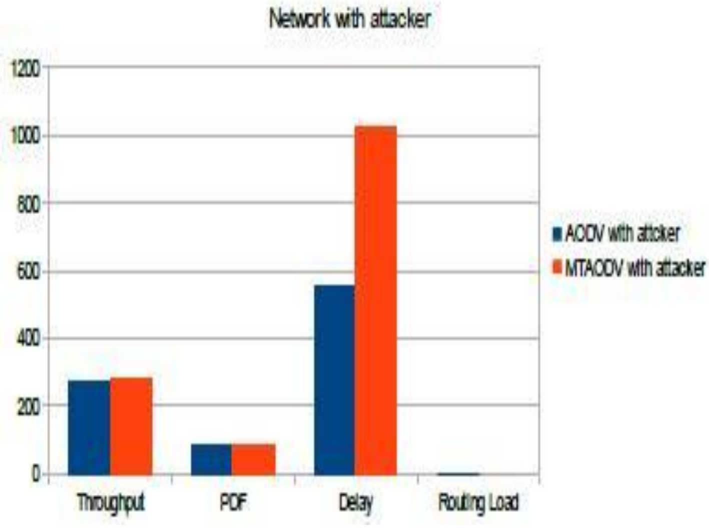


Fig. 9. Performance analysis of intruders attack.

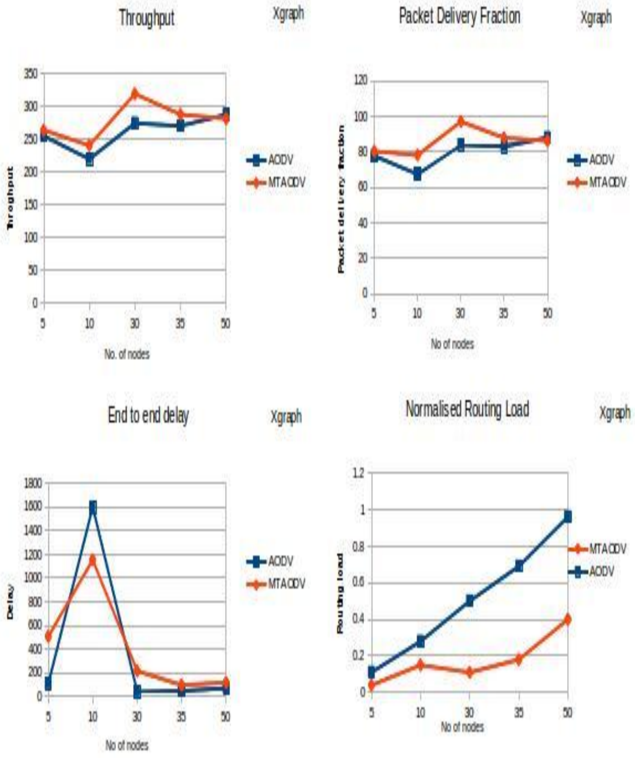


Fig. 10. Performance analysis using various parameters.

VII. CONCLUSION

In this paper, we have shown the design of a protocol that allows the creation and management of a spontaneous wireless ad hoc network. It is based on a social network imitating the behavior of human relationships. Thus, each user will work to maintain the network, improve the services offered, and provide information to other network users. We have provided some procedures for self-configuration, a unique IP address is assigned to each device, the DNS can be managed efficiently and the services can be discovered automatically. We have also created a user-friendly application that has minimal interaction with the user. A user without advanced technical knowledge can set up and participate in a spontaneous network. The security schemes included in the protocol allow secure communication between end users (bearing in mind the resource, processing, and energy limitations of ad hoc devices). We have performed several simulations to validate the protocol operation using NS2. They showed us the benefits of using this self-configuring ad-hoc spontaneous network. And while comparing with the existing systems, the performance evaluation metrics values are higher than traditional one. In future, we intend to add some new features to the user application (such as sharing other types of resources, etc.) and to the protocol, such as an intrusion detection mechanism and a distributed Domain Name Service by using the LID and IP of the nodes. Now, we are working on adding other types of nodes that are able to share their services in the spontaneous network. The new nodes will not depend on a user, but on an entity such as a shop, a restaurant, or other types of services.

REFERENCES

[1] L.M. Feeney, B. Ahlgren, and A. Westerlund, “Spontaneous Networking: An Application-Oriented Approach to Ad-hoc Networking,” IEEE Comm. Magazine, vol. 39, no. 6, pp. 176-181, June 2001.

[2] V. Kumar and M.L. Das, “Securing Wireless Sensor Networks with Public Key Techniques,” Ad Hoc and Sensor Wireless Networks, vol. 5, nos. 3/4, pp. 189-201, 2008.

[3] Suchita Gupta, Ashish Chourey, “Performance Evaluation of AODV Protocol Under Packet Drop Attacks in Manet”, International Journal of Research in Computer Science eISSN 2249-8265 Volume 2 Issue 1 (2011) pp. 21-2.

[4] Dalip Kamboj and Pankaj Kumar Sehgal, “A Comparative Study of various Secure Routing Protocols based on AODV”, International Journal of Advanced Computer Science and Applications,Vol. 2, No. 7, 2011, pp 80-85.

[5] Mohd Anuar Jaafar and Zuriati Ahmad Zukarnain, “Performance Comparisons of AODV, SecureAODV and Adaptive Secure AODV Routing Protocols in Free Attack Simulation Environment”, European Journal of Scientific Research ISSN 1450-216X Vol.32 No.3 (2009), pp.430-443.

[6] Songbai Lu1, Longxuan Li and Kwok-Yan Lam, LingyanJia, “SAODV: A MANET Routing Protocol that can Withstand Black Hole Attack”, IEEE 2009 International Conference on Computational Intelligence and Security, pp 421-425.

[7] N.R. Potlapally, S. Ravi, A. Raghunathan, and N.K. Jha, “Analyzing the Energy Consumption of Security Protocols,” Proc. Int’l Symp. Low Power Electronics and Design (ISLPED ’03), 2003.

[8] S Landau, “Communications Security for the Twenty-First Century: The Advanced Encryption Standard,” Notices of the Am. Math. Soc., vol. 47, no. 4, pp 450-459, Apr. 2000.

[9] R. Mayrhofer, F. Ortner, A. Ferscha, and M. Hechinger, “Securing Passive Objects in Mobile Ad-hoc Peer-to-Peer Networks,” Electronic Notes in Theoretical Computer Science, vol. 85, no. 3, pp. 105-121, Aug. 2003.

[10] Raquel Lacuesta, Jaime Lloret, Miguel Garcia, and Lourdes Penalver “A Secure Protocol for Spontaneous Wireless Ad Hoc Networks Creation” IEEE transactions on parallel and distributed systems, vol. 24, no. 4,pp. 629-641, Apr. 2013.