Journal of Theoretical and Applied Information Technology
31st December 2013. Vol. 58 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NOVEL ELLIPTIC CURVE CRYPTOGRAPHY BASED AODV FOR MOBILE AD-HOC NETWORKS FOR ENHANCED SECURITY

[1]M JANARDHANA RAJU, [2]Dr.P.SUBBAIAH, [3]V.RAMESH

[1]Research Scholar, Sathyabhama University, Chennai, TamilNadu, India.
[2]Professor, Dhanalakshmi College of Engineering, Anna University, Chennai, India
[3]Associate Professor, Padmasri Dr.B.V.Raju Institute of Technology & Science, Narsapur, AP, India.
E-mail: [1]raju243102@gmail.com , [2]subbaiah_nani@sifi.com , [3]v2ramesh634@yahoo.co.in

ABSTRACT

The all-pervading nature of communication networks has paved the way for the development of wireless and internet applications, making communication possible all over the world. With the explosion of networks and the huge amount of data transmitted along, securing the data content is becoming more and more important. Mobile ad hoc networks are multi hop wireless networks without fixed infrastructure. So there is a possibility to get several types of attacks including denial of service attacks, which leads to consume the system resources like bandwidth, power and memory. To keep away from these vulnerable attacks researchers proposed many schemes, but still those are possessing massive threats. Hence there is a necessity of new secure routing mechanism. Hence, we have introduced a novel secured Elliptic Curve based Ad-hoc On-demand Distance Vector routing protocol. Here the main advantage with ECC is, it takes less memory provides great security and flawlessly suitable for low power devices like mobile nodes. So the performance of the overall system is good compare with other secure routing mechanisms.

Keywords:- *Elliptic Curve Cryptography, AODV, RSA, SEAD, SRP.*

1. INTRODUCTION

A collection of moveable wireless devices forms the MANET. MANET routing aids to establish the wireless communication among the wireless portable devices[14]. The routing process is used to refer the data packet transmission from the source to the Destination Node. Several routing protocols are suggested in MANET. The routing protocols are used to specify how routers can make the communication with others.

1.1Ad-Hoc On-demand routing protocol (AODV)

In the Reactive AODV protocol, the main features of both DSDV and DSR are inherited. In that, the Reactive approach is used to determine the optimized path and the Proactive approach is used to identifying the fresh paths. It maintains the Sequence Number for every route to compute the recent path like DSR. The AODV also has two phases called Path Discovery and Path Maintenance.[15] During the path discovery process, the Source Node broadcasts the route advertisement packets to the immediate neighbor nodes. The advertised RREQ packets retain source and destination identifier (id), RREQ id, and Time to Live (TTL). If the intermediate node has a valid path to the Destination Node in the cache, the RREP packet is sending back to the Source Node.

The intermediate nodes record the received RREQ packet id and the corresponding Source Node id. It aids to avoid the transmission of duplicate packets. If any node receives the RREQ packet which has already recorded in the cache, the duplicate packets are dropped. Otherwise, the packet is broadcasted to immediate neighbor nodes. Every packet maintains the TTL value in the header. If the RREP is not reached within the TTL time, the route is not valid to transmit the data packet. When the RREQ packet reaches the required Destination Node, the RREP packet is sent back to the Source Node.[15][16] Every node holds the preceding node information to forward the data packets when it receives the RREQ packet. In source routing, every node retains the entire route information of all other nodes, but in AODV, each node holds only the next hop information.

2. RELATED WORK

To detect the resource consumption attacks, Jian-HuaSong, Fan Hong and Yu Zhang in [1] proposed

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

a method to prevent the RREQs flooding by considering three parameters like rate limit, blacklist limit and delay timeout. Rafsanjani, Khavasi and Movaghar [2] proposed an IDS system for selecting the compromised node in the network using non interactive zero knowledge technique. They consider one node as an agent node having more resources like band width, power so that node will take care of compromised nodes. In Dai Hong, Li Haibo [3] proposed Network Intrusion Detection System (NIDS) which uses all data features that are irrelevant and redundant features.

This can influence both the performance of the system and the types of attacks that NIDS detects. Selection algorithm based on Chi-Square and enhanced C4.5 algorithms to build lightweight network intrusion detection are also proposed. Verification test have been carried out by using the KDD Cup 1999[4][5] datasets. Guha, O.Kachirski and D.G.Schwartz [6] proposed a method which utilize cluster and cluster head employs the independent decision making. It also utilizes the mobile agent for communications among nodes. The intrusion detection engine is a case-based agent designed with the principle of artificial intelligence. by this method the efficient, bandwidth conscious, take into account of the distributed nature of MANET. But the disadvantage is Mobile agent's security is hard to implement, packet drop rate increase when network load increase. Sun B, K.wu and U.pooch [7] Implements an IDS which use collaboration mechanism in anomaly detection. In this model, a network is divided into logical zones. Each zone has a gateway node and individual nodes. Individual nodes has IDS agent working and detect intrusion activities individually. Once an individual node detects intrusion, it generates an alert message. Gate way node aggregates and correlates the alerts generated by the nodes in its zone. An algorithm is used in aggregate the alerts based on the similarities in the attributes of the alert. Only gateway nodes can utilize alert to in it alarm Nakayama, Kurosawa, Jamalipour, Nemoto and Nei Kato [8] proposed dynamic anomaly detection by using dynamic learning process. It involves the method to calculate the projection distance and compare it base line profile using PCA.

3. PROPOSED WORK

3.1 Why ECC in MANET Routing

There are many reasons why there have been introduced the concept of ECC in MANET. According to [10] and [11], Compared to traditional cryptosystems like RSA, ECC offers comparable security with smaller key sizes, faster computation, lower power consumption, as well as memory and bandwidth savings. This is especially useful for mobile devices which are characteristically limited in terms of their CPU, power and network connectivity. In [12], it is shown how the energy cost for RSA is greater than that of an ECDSA (a Signature Algorithm of ECC) showing better performance.

Though, in [13][18] although it is renowned that attraction of ECC is that it appears to offer equal security for a far smaller key size, therefore reducing processing overhead, also is showed that: the methods for computing general elliptic curve discrete logarithms are much less efficient than those for factoring or computing conventional discrete logarithms and it indicates that more computation time is required for ECC and considers that the overall performance of ECDLP-based applications needs to be evaluated. However, compared to many other conventional cryptosystems, ECC is a very good alternative to the characteristics of a MANET.[21]

3.2 Elliptic Curve Cryptography for AODV

The ElGamal cryptosystem can be based on any cyclic group and the same system is applied on the elliptic curve group, $E(F_p)$ (Koblitz 1987). As in Section 3.3, addition is not a straight forward process. Moreover, in Section 3.5.3, determining the group order of $E(F_p)$ requires complex algorithms. This means that with ECC, smaller keys are sufficient to provide more security in message or data transmission. For example, the sizes of public and private keys of the ElGamal cryptosystem are 3072-bit and 256-bit respectively, while a 163-bit key size of an elliptic curve cryptosystem provides the same security level as the ElGamal keys (Lauter 2004). It is for these reasons that the NSA recommends EC cryptographic algorithms.

There are several elliptic curve cryptosystems such as the analog of the ElGamal encryption, the analog of the RSA encryption and the Weil Pairing encryption. This thesis emphasises on the elliptic curve ElGamal encryption procedure.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.2.1 Elliptic Curve Key Pairs

An elliptic curve key pair is associated with a particular set of domain parameters $D = (q, FR, S, a, b, G, n, h)$. The public key is a randomly selected point Q in the group generated by G. The corresponding private key is $d = log_G Q$. The entity A generating the key pair must have the assurance that the domain parameters are valid. The association between domain parameters and a public key must be verifiable by all entities who may subsequently use A's public key. In practice, this association can be achieved by cryptographic means (e.g., a certification authority generates a certificate attesting to this association) or by context (e.g., all entities use the same domain parameters).

Algorithm generateKeyPair( )

// **Input** : Domain parameters D
// **Output** : Public key Q, private key d
{
Select $d \in R [1, n-1]$
Compute $Q = dG$
Return (Q, d)
}

The problem of computing a private key d from the public key Q is precisely the ECDLP. Hence, it is crucial that the domain parameters D be selected so that the ECDLP is intractable. Furthermore, it is important that the numbers d generated be random in the sense that the probability of any particular value being selected must be sufficiently small to preclude an adversary from gaining advantage through optimizing a search strategy based on such probability.

The purpose of public key validation is to verify that a public key possesses certain arithmetic properties. Successful execution demonstrates that an associated private key logically exists, although it does not demonstrate that someone has actually computed the private key nor the claimed owner actually possesses it. Public key validation is especially important in DH based key establishment protocols where an entity A derives a shared secret k by combining the private key with a public key received from another entity B, and subsequently uses k in some symmetric key protocol (e.g., encryption or message authentication). A dishonest B might select an invalid public key in such a way that the use of k reveals information about A's private key.

Algorithm validatePublicKey( )

// **Input** : Domain parameters D, public key Q
// **Output** : Acceptance or rejection of the validity of Q
{
Check that $Q \neq \infty$
Verify that x_Q and $y_Q \in F_p$
Check that Q satisfies the EC equation defined by a and b
Verify that $nQ = \infty$
If (Verification fails) then
Return "Invalid"
Else
Return "Valid"
}

There may be much faster methods for verifying that $nQ = \infty$ than performing an expensive point multiplication nQ. For example, if $h = 1$ which is usually the case for ECs over prime fields that are used in practice, then the checks for first three steps of algorithm *'validatePublicKey'* imply that $nQ = \infty$. In some protocols the check that $nQ = \infty$ may be omitted and either embedded in the protocol computations or replaced by the check that $hQ \neq \infty$. The latter check guarantees that Q is not in a small subgroup of $E(F_p)$ of order dividing h.

3.2.2 Elliptic Curve Encryption Scheme

For an elliptic curve ElGamal encryption, all computations are done in the finite field F_p. The encryption and decryption procedures for the elliptic curve analogue on the basic ElGamal encryption scheme are presented as algorithms *'encryptECElGamal'* and *'decryptECElGamal'* respectively. A plaintext m is first represented as a point P_m, and then encrypted by adding it to kQ, where k is a randomly selected integer, and Q is the intended recipient's public key. The sender transmits the points $C_1 = kG$ and $C_2 = P_m + kQ$ to the recipient who uses the private key d to compute $dC_1 = d(kG) = k(dG) = kQ$, and thereafter recovers $P_m = C_2 - kQ$. An eavesdropper who wishes to recover P_m needs to compute kQ. This task of computing kQ from the domain parameters, Q, and $C_1 = kG$, is the elliptic curve analogue of the DH problem

.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Algorithm encryptECElGamal( )
// **Input :** EC domain parameters (p, E, G, n), public key Q, plaintext m
// **Output :** Cipher text (C_1, C_2)
{
Represent the message m as a point P_m in $E(F_p)$
Select $k \in R$ [1, n − 1]
Compute $C_1 = kG$
Compute $C_2 = P_m + kQ$
Return (C_1, C_2)
}

Algorithm decryptECElGamal( )
// **Input :** EC Domain parameters (p, E, G, n), private key d,
cipher text (C_1, C_2)
// **Output :** Plaintext m
{
Compute $P_m = C_2 - dC_1$
Extract m from P_m
Return (m)
}

As in the finite field case, the security of this cryptosystem lies in the fact that if only G and Q are known to the adversary, it is difficult to determine the number of times G has been added to itself to get Q. This property is due to the random additive structure of points. Koblitz (1987) mentioned that the techniques developed to solve the DLP for finite fields often fail to work for the ECDLP. This fact enables this elliptic curve cryptosystem to remain secure while keeping the size of the field small. There exist several methods but without an efficient algorithm to attack the system, the difficulty in solving the ECDLP remains the key advantage of using ECs in cryptography.

3.2.3 Elliptic Curve Diffie Hellman Scheme

Elliptic Curve Diffie Hellman (ECDH) is a key agreement scheme that allows two entities to establish a shared secret key that can be used for private key algorithms. Both entities exchange some public information to each other. Using this public information and their own private information these entities calculate the shared secret.

For generating a shared secret between two entities A and B using ECDH, both have to agree upon elliptic curve domain parameters. Both entities have a key pair consisting of a private key (a randomly selected integer less than n, where n is the order of the curve, an elliptic curve domain parameters) and a public key (G is the generator point, an elliptic curve domain parameter). Let (n_A, P_A) be the private key - public key pair of user A and (n_B, P_B) be the private key - public key pair of user B. Since the shared secret key $k = n_A P_B = n_A n_B G = n_B n_A G = n_B P_A = k$. The algorithm *'ComputeECDHSecretKey'* is used to compute the shared secret key between two users A and B.

Algorithm ComputeECDHSecretKey( )
// **Input :** EC domain parameters (p, E, G, n)
// **Output :** Secret key k
{
User A select $n_A \in R$ [1, n − 1]
User A compute $P_A = n_A G$
User B select $n_B \in R$ [1, n − 1]
User B compute $P_B = n_B G$
User A calculate $k = n_A P_B$
User B calculate $k = n_B P_A$
Return k
}

Since it is practically impossible to find the private key n_A or n_B from the public key P_A or P_B, it is not possible to obtain the shared secret key k for a third party.

4. RESULTS AND DISCUSSION

For demonstration purposes, elliptic curve is represented by $y^2 = x^3 + x + 1$ defined over $E(F_7)$, where $a = 1$, $b = 1$ and $p = 7$. The coefficients a and b are chosen based on the condition that $4a^3 + 27b^2 \bmod p = 31 \bmod 7 = 3 \neq 0$, so $E(F_7)$ is indeed an elliptic curve. The generated points on the EC can be found and they are {O, (0, 1), (0, 6), (2, 2), (2, 5)}.Then consider the elliptic curve $E: y^2 = x^3 + 4x + 20$ defined over F_{29} with the constants $a = 4$ and $b = 20$ which have been checked to satisfy that E is indeed an elliptic curve. The 37 points in $E(F_{29})$ are the following:
{ O, (0, 7), (0, 22), (1, 5), (1, 24), (2, 6), (2, 23), (3, 1), (3, 28), (4, 10), (4, 19), (5, 7), (5, 22), (6, 12), (6, 17), (8, 10), (8, 19), (10, 4), (10, 25), (13, 6), (13, 23), (14, 6), (14, 23), (15, 2), (15, 27), (16, 2), (16, 27), (17, 10), (17, 19), (19, 13), (19, 16), (20, 3), (20, 26), (24, 7), (24, 22), (27, 2), (27, 27) }.
The point (1, 5) in $E(F_{29})$ satisfies the equation (3.2) since:

$$y^2 \bmod p = x^3 + 4x + 20 \bmod p$$
$$25 \bmod 29 = 1 + 4 + 20 \bmod 29$$
$$25 = 25$$

Similarly, other generated EC points also satisfy the equation. These points are graphed in the following Figure 3.3.

ISS 8645 www.jatit.org E-ISSN: 1817-3195

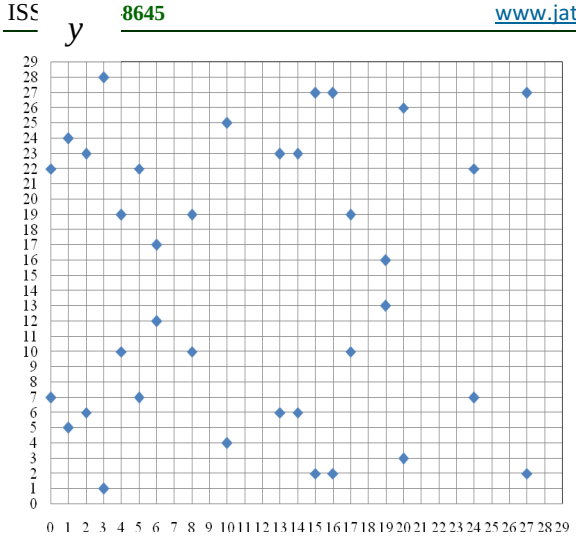

Figure 1: Elliptic Curve Point Representation

Note that there are two points for every x value. Over the field of F_{29}, the negative components in the y-values are taken modulo 29, resulting in a positive number as a difference from 29. Here $-P = (x_1, (-y_1 \bmod 29))$. Based on the field size p, the number of points on the EC can be varied. Point addition and point doubling are the basic EC operations. Given EC, E: $y^2 = x^3 + x + 1$ over F_{13}. The group $E(F_{13})$ has 18 elements.
{ O, (0, 1), (0, 12), (1, 4), (1, 9), (4, 2), (4, 11), (5, 1), (5, 12), (7, 0), (8, 1), (8, 12), (10, 6), (10, 7), (11, 2), (11, 11), (12, 5), (12, 8) }

Consider two points $P = (12, 8)$ and $Q = (1, 9)$ on $E(F_{13})$. The addition of two points $P + Q = (x_3, y_3)$ where $P \neq \pm Q$ is computed as follows: First calculate λ to be:

$\lambda = (9 - 8) / (1 - 12) = 1 / (-11) = 1 / 2 \equiv 7 \pmod{13}$

Then, using the formulae in equation (3.4) to calculate the coordinates as

$x_3 = 72 - (12 + 1) \equiv 10 \pmod{13}$

$y_3 = 7(10) + 2 \equiv 7 \pmod{13}$

So, (12, 8) + (1, 9) = (10, 7) which also lies on the elliptic curve $E(F_{13})$.

Consider a point $P = (11, 2)$ on $E(F_{13})$. To add a point to itself that is double a point $2P = (x_3, y_3)$. First found λ to be:

$\lambda = 3(11^2) + 1 / 2 \times 2 \equiv 0 \pmod{13}$

Then using the formulae in equation (3.5) to calculate the coordinates as

$x_3 = 0 - (2 \times 11) \equiv 4 \pmod{13}$

$y_3 = 0 + 2 \equiv 2 \pmod{13}$

So, $2 \times (11, 2) = (4, 2)$ which lies on $E(F_{13})$.

Cryptographic schemes based on ECC rely on scalar multiplication of EC points. Let P is a point on an EC, and one needs to compute kP, where k is a positive integer. This scalar multiplication can be computed efficiently by a series of doubling and addition of P. For example, given $k = 13$, entails the following sequence of operations, by which the efficiency of the scalar multiplication of the points is improved.

For example, consider the point $P = (12, 8)$ on the elliptic curve $E(F_{13})$. Based on the algorithm *'computeScalarMul'* the value of $13P$ is computed as (1, 4) which also lies on the elliptic curve $E(F_{13})$. The following example demonstrates the encryption and decryption processes using EC. Consider the EC $y^2 = x^3 - 5x + 25 \bmod 487$. Here, $a = -5$, $b = 25$ and $p = 487$ are the parameters of EC. Using equation (3.2) the points are generated. The base point G of an EC is selected as (0, 5). Assume that the user A wants to send the message 48 to user B. First choose the random point on EC as (1, 316). The message is encoded as point on EC as (12, 233).

User B chooses the private key n_B as x The public key P_B is computed accordir the algorithm *'generateKeyPair'* as (260, 48). According to algorithm *'encryptECElGamal'*, user A chooses k as 225 and compute C_1 as (0, 5). Then compute C_2 as (12, 233) + (260, 48) = (384, 288). Therefore, the cipher text $C = (C_1, C_2) = ((0, 5), (384, 288))$. According to algorithm *'decryptECElGamal'*, compute encoded message EC point as (384, 288) - 277(0, 5) = (12, 233). Then extract the message from (12, 233) using discrete logarithm concept as 48.
The following example demonstrates how two users generate a secret key using ECDH. Consider the EC $y^2 = x^3 - 5x + 25 \bmod 487$. Here, $a = -5$, $b = 25$ and $p = 487$ are the parameters of EC. User A select the private key n_A as 719 and its public key is computed according to the algorithm *'generateKeyPair'* as $P_A = n_A G = 719(0, 5) = (213, 351)$. Similarly, user B choose the private key n_B as 967 and its public key is calculated as $P_B = n_B G = 967(0, 5) = (114, 364)$. Based on the algorithm *'ComputeECDHSecretKey'*, the shared secret key $k = n_A P_B = n_B P_A = 719(114, 364) = 967(213, 351) = (195, 469)$.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4.1 Simulation and Evaluation of Results

This section describes the simulation tool and parameters chosen to simulate the routing protocols. For simulation software, NS2.34 is used to evaluate the performance of SEAD (Secure Efficient Ad-Hoc Distance Vector Routing Protocol), SRP and AODV with ECC routing protocols.

Simulation Parameter	Value
Simulator	NS-2
Node Movement Model	Random Waypoint
Speed	0-25m/s
Traffic Type	UDP
Bandwidth	2Mb/s
Transmission Range	250m

Parameters used in simulations are shown in Table-1. We compare access control mechanism on elliptic curve cryptography with popular RSA algorithm. ECC is giving better security when compared with RSA, which takes fewer bits key and providing good security. Elliptic Curve Cryptography may be the standard for the next generation cryptographic technology. The reason is that ECC can achieve the better level of security with smaller key sizes. It has been shown that 160-bit ECC provides comparable security to 1024-bit RSA and 224-bit ECC provides comparable security to 2048-bit RSA. Under the same security level, smaller key sizes of ECC offer merits of faster computational efficiency, as well as memory, energy and bandwidth savings.

Table 1: Comparison of AODV with ECC, SEAD and SRP

Parameter	AODV with ECC	SEAD	SRP
Number of CBR data packets produced	1144	681	653
Number of UDP data packets generate	18021	1497	5
Number of CBR data packet sent	586	584	584
Number of UDP data packets sent	9090	760	05
Number of dropped Packets	328	513	510
Packet delivery ratio (CBR and UDP) in %	0.98	0.62	0.12
Average Delay	1.582	2.468	1.692
Normalized routing load	7.95	6.69	40.775

4.2 Performance evaluation of AODV with ECC

4.2.1 Simulation Metrics

Packet Delivery Fraction (PDF):

It is the ratio of total number of packets successfully received at the Destination Nodes to the number of packets are forwarded from the Source Nodes throughout the simulation.

$$PDF = \frac{\text{Number of Received Packets}}{\text{Number of Sent Packets}}$$

Average End to end delay of data packets:

$$\text{AED} = \sum_{i=0}^{n} \frac{(\text{Time of packet Recived} - \text{Time of Packet Sent})}{\text{Total Number of Packets Received}}$$

The average End-to-End delay is defined as the average time from the beginning of a packet transmission at a Source Node until the packet is delivered to a destination.

Normalized Routing Load (NRL):

This is calculated as the ratio between the no. of routing packets transmitted to the number of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

packets actually received (thus accounting for any dropped packets).

4.2.1.1 Effect of varying the number of nodes

The number of nodes are varied from 50 to 100 and the effect of PDF, NRL and AED is studied. The results can be found in figure 2, 3 and 4. It has been observed that the packet delivery ratio decreases as the number of nodes increases in the network. This is due to the fact that as number of nodes increases, the congestion in the network also increases and hence the number of lost packets due to retransmission also increases.

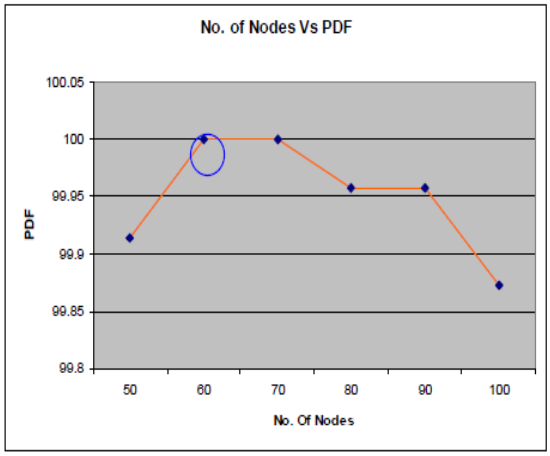

Figure 2: Effect Of Varying The Number Of Nodes On The Pause Time

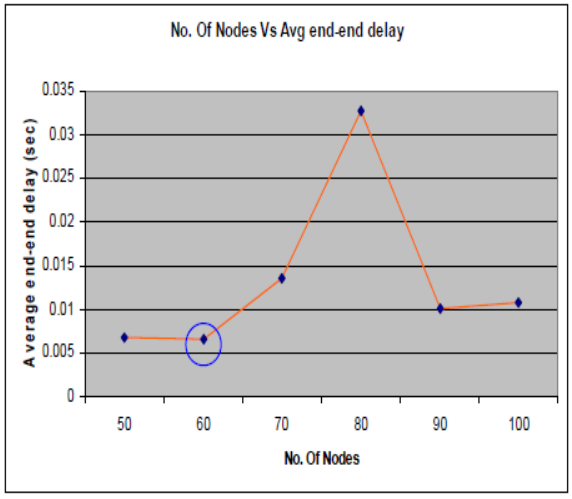

Figure 3: Effect Of Varying The Number Of Nodes On The Average End-End Delay

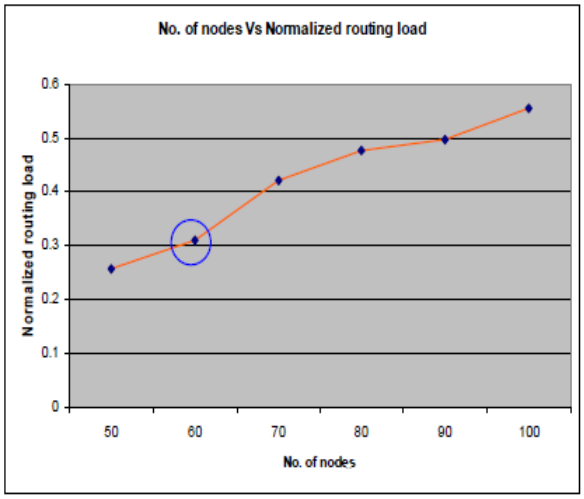

Figure 4: Effect Of Varying The Number Of Nodes On The Normalized Routing Load

The circles in figures 2, 3 and 4 represent the "optimal points" which corresponds to the highest PDF, lowest end-to-end delay and the lowest normalized routing load. It has been observed that for 60 nodes we achieve this optimal point.

4.2.1.2 Effect of varying the pause time

The effect of varying the pause time on these metrics are shown in figures 5,6 and 7. It is found that as pause time varies, the packet delivery fraction also increases. It is due to the reality that as pause time increases, the relative mobility of the nodes decreases, and hence the congestion also decreases in the network.

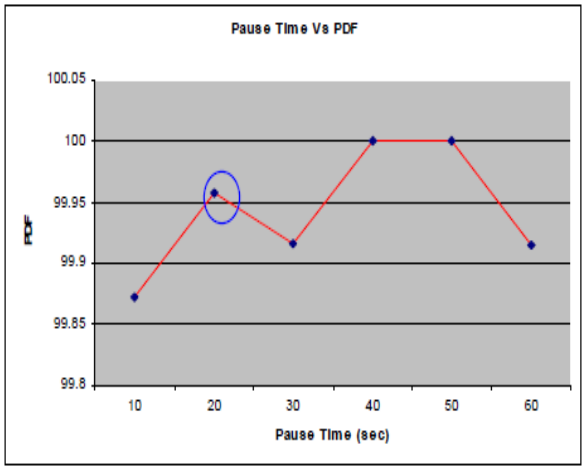

Figure 5: Effect Of Varying The Pause Time On PDF

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

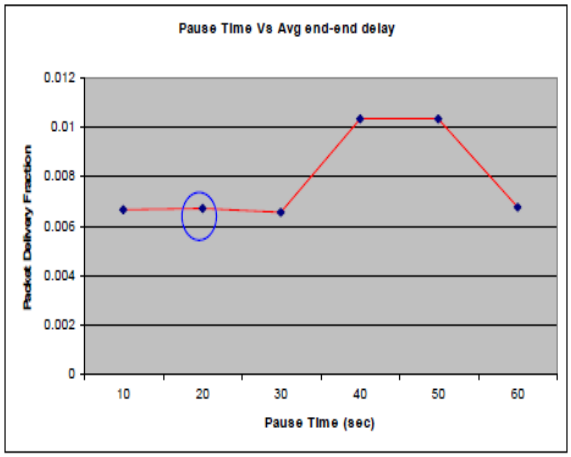

Figure 6: Effect Of Varying The Pause Time On Average End To End Delay

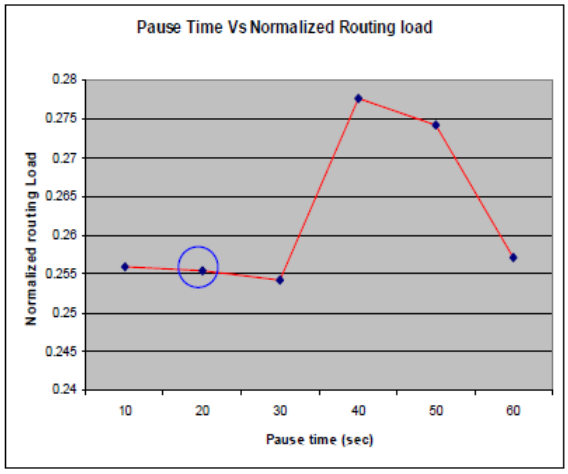

Figure 7: Effect Of Varying The Pause Time On NRL

5. CONCLUSION

Secure routing is crucial to the acceptance and use for many MANET network applications. In this research work, we have AODV routing protocol with integration of Elliptic Curve Cryptography to guarantee security for the routing information. It has given wide-ranging security over RSA encryption algorithm and also it is suitable for low power devices like MANETs and sensor nodes. It increase the life time of the network and it provides efficient secure transmission of data over SEAD and SRP protocols.

REFERENCES

[1] Jian-HuaSong, Fan Hong and Yu Zhang," Effective Filtering Scheme against RREQ Flooding Attack in Mobile Ad Hoc Networks", Proceedings of the Seventh International Conference on Parallel and Distributed Computing, Applications and Technologies, pp. 497-502 ,2006.

[2] Marjan Kuchaki Rafsanjani, Ali Asghar Khavasi and Ali Movaghar,"An Efficient Method for Identifying IDS Agent Nodes by Discovering Compromised Nodes in MANET" in proc ICCEE , Vol.01., pp.625-629, December 2009.

[3] H. Liu, Setiono and R., "Chi2: feature selection and discretization of numeric attributes", in Proc of the Seventh International Conference on Tools with Artificial Intelligence, pp. 388 - 391, 1995.

[4] Wanli Ma, Dat Tran, Dharmendra Sharma,"A Study on the Feature Selection of Network Traffic for Intrusion Detection Purpose"p.p 245-247,2008, Taipei, Taiwan.

[5] ACM. KDD CUP 1999 data. [Cited 12 January 2007]; Available from: http://kdd.ics.uci.edu//databases/kddcup99/kddcup99.html.

[6] Dai Hong, Li Haibo.A Lightweight Network Intrusion Detection Model Based on Feature Selection. DOI 10.1109/PRDC.2009.34,p.p 165-168.

[7] C.K.Toh,"Ad-Hoc mobile wireless network protocol and system", Pearson Education, 2009.

[8] Hidehisa Nakayama,Satosi Kurosawa, Abbas Jamalipour,Yoshiaki Nemoto and Nei Kato, " A Dynamic Anomaly Detection Scheme For AODV- Based Mobile AdHoc Networks", IEEE Transactions On Vehicular Technology,Vol.58, No. 5, pp.2471-2481, June 2009.

[9] Yih-Chun Hu, David B. Johnson, Adrian Perrig. "SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks", Fourth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '02), pp: 3-13, Jun 2002.

[10] V. Katiyar, K. Dutta, S. Gupta; "A Survey on Elliptic Curve Cryptography for Pervasive Computing Environment." *International Journal of Computer Applications* 11(10):41–46, December 2010.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[11] Xu Huang; Shah, P.G.; Sharma, D.; , "Protecting from Attacking the Man-in-Middle in Wireless Sensor Networks with Elliptic Curve Cryptography Key Exchange," *Network and System Security (NSS), 2010 4th International Conference on* , vol., no., pp.588- 593, 1-3 Sept. 2010.

[12] Jia Xiangyu; Wang Chao; , "The application of elliptic curve cryptosystem in wireless communication," *Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications, 2005. MAPE 2005. IEEE International Symposium on* , vol.2, no., pp. 1602- 1605 Vol. 2, 8-12 Aug. 2005.

[13] Yong Wang; Ramamurthy, B.; Xukai Zou; , "The Performance of Elliptic Curve Based Group Diffie-Hellman Protocols for Secure Group Communication over Ad Hoc Networks," *Communications, 2006. ICC '06. IEEE International Conference on* , vol.5, no., pp.2243-2248, June 2006.

[14] A. Al-Maashri, M. Ould-Khaoua, Performance Analysis of MANET Routing Protocols in the Presence of Self-Similar Traffic, Proceedings of 31st IEEE Conference on Local Computer Networks, 14-16 Nov. 2006, pp. 801–807.

[15] R. Bai, M. Singhal, DOA: DSR over AODV Routing for Mobile Ad Hoc Networks, IEEE Transactions on Mobile Computing, Vol. 5, No. 10, pp. 1403– 1416, 2006.

[16] A. Chaplot, A Simulation Study of Multi-Hop Wireless Network, IEEE International Conference on Personal Wireless Communications, pp. 86–89, December 15-17, 2002.

[17] T.-C. Huang, C.-C. Chan, Caching Strategies for Dynamic Source Routing in Mobile Ad Hoc Networks, IEEE Wireless Communications and Networking Conference (WCNC) 2007, 11-15 March 2007, pp. 4239 – 4243.

[18] Song, N.; Qian, L.; Li, X.; , "Wormhole attacks detection in wireless ad hoc networks: a statistical analysis approach," *Parallel and Distributed Processing Symposium, 2005. Proceedings. 19th IEEE International* , vol., no., pp. 8 pp., 4-8 April 2005.

[19] Yun Zhou, Yanchao Zhang, Yuguang Fang, "Access control in wireless sensor networks," Ad - hoc Networks 5 (2007) 3–13.

[20] Dong-Won Kum, Jin-Su Park, You-Ze Cho and Byoung- Yoon Cheon," Performance Evaluation Of AODV and DYMO Routing Protocols in MANET", in proc IEEE CCNC, Las Vegas, Nevada, USA, pp.1046-1047, Jan.2010.

[21] Yun Wang; Zhongke Zhang; Jie Wu; , "A Distributed Approach for Hidden Wormhole Detection with Neighborhood Information," *Networking, Architecture and Storage (NAS), 2010 IEEE Fifth International Conference on* , vol., no., pp.63-72, 15-17 July 2010.