SECURING AODV ROUTING PROTOCOL IN MANET BASED ON CRYPTOGRAPHIC AUTHENTICATION MECHANISM

Author Name **Hifza M Anwar (63110)**
COCIS
PAF-KIET,
Karachi, Pakistan
E-mail hifzaansari999@gmail.com

Author Name **Qasim Hassan(62357)**
COCIS
PAF-KIET,
Karachi, Pakistan
E-mail qasimhassan708@gmail.com

Author Name **Areeba Qamar (61898)**
COCIS
PAF-KIET,
Karachi, Pakistan
E-mail areebaqamar@gmail.com

INTRODUCTION:

During these modern eras, Mobile Ad-hoc Network (MANET) remains one of these typical conservative research areas in software engineering. MANET implies an emerging technology in which node assists as WIFI as well as the processor to communicate amongst per other in a peer to peer way in multi-hope externally enduring every foundation [1]. That expects that it is also recognized because an infrastructure-less network in which there occurs no necessity for either base station or access point is expected in infrastructure-based networks [2]. Ad Hoc system operates externally established infrastructure. Versatility, multi-hop large network capacity blends among the bandwidth and battery strength restrictions. People can use this network swiftly and efficiently. Users stay joins to the network if these users propelled around. In both military and civilian systems, the wireless network operates an essential function. Users can combine with PCs, laptops, ship network, vehicles quickly used in a pressure situation of all sorts of network applications. Network topologies remain irregular and powerful. Universal Networks can't be directly correlated just as wireless networks because of the most maximum of are wired networks.[4] To remain separate from several defenseless attacks researchers suggested many projects, but even so, these are holding heavy warnings. Thus, there is a constraint of a modern protected routing device. Therefore, we have proposed RSA, AES, Discrete and Elliptic Curve based Ad-hoc On-demand Distance Vector routing protocol. Here we will analyze certain algorithms to reveal which algorithm is most beneficial to guard Ad-hoc on-demand Distance Vector Routing Protocol.

LITRETURE REVIE:

AODV routing protocol is basically a combination of the protocols DSDV and Data Source Routing (DSR) [10 , 11]. AODV (Ad Hoc on Demand distance vector) is a reactive protocol that acquires some essential on-demand means of path discovery and routes sustaining of DSR. The aim of this is to devise a route for lowering route load. It is bi-directional from source and destination, and when needed it sends the message[4]. The routing messages in AODV only contain source and destination information. AODV defines message types as Route Requests (RREQs), Route Replies (RREPs), and Route Errors (RERRs)[12]. Often, when appropriate, AODV finds ways that do not seem to keep routes from each connection to each other. Each connection intends its tediously rising number of courses.

The node report variation in the neighborhood topology which implies sequence figure is progressing every time. AODV utilizes routing tables to collect routing data. This route table reserves data in every form: < address, next-hope address, destination order number, life time>. AODV relies upon a performance learning mechanism and route preservation. For example, sender 'S' broadcast a piece of information to all its neighbors, every node getting the information from 'S' delivers the information to its individual neighbors. The information approaches destination 'D' presented that 'D' is reachable from the sender 'S'. This method of conveying the information from origin node 'S' to destination node 'D' a chain system, till the information arrives at the final destination 'D'. Node 'D' commences up a reverse route reply (RREP) for the origin node 'S' in its route table. As large as this route persists active, it will remain to be prepared. A route is regarded as working as large as information packages systematically are moving from one origin to the destination simultaneously the route. Once that origin prevents transmitting information packages, the connections will be time out also finally be removed from these common node routing tables [13]. If a connection failure happens while that design is active, the node upstream of the failure creates a route error (RERR)

message to the origin node to notify among regard to the distant destinations [14]. Later getting these RERR, if one origin node besides want that path, it can relaunch route spotting.

Protection Problems of AODV:

AODV routing rules seem not to give any security instruments to make preparation for assault. The significant susceptivity started in AODV rules are:

1. An Intruder can imitate an origin link A by producing an RREQ including its IP address as IP
2. Address of source node S.
3. Attacker can imitate a destination node D by producing a RREP with its IP address as
4. IP address like each target node D.
5. Diminishing hop calculation in RREQ/RREP.
6. Rising order figure in RREQ/RREP.
7. Advancing the RRER information.

AODV routing order needs at minutest two protection assistance: Data starting point validation on

various gathering nodes and routing message integrity. Message integrity is of the most firm in

AODV routing. A cruel node or negotiation node may change sequence number or hop

Count Fields in RREQ /RREP messages or imitate the sender of routing packets.

Amendment of routing data may direct to the disparity in the network. The routing table may

include incorrect data on network topology. Variation in order amount may appear in

routing circuits etc.

AODV Security by RSA:

RSA is a fully-known cryptography structure. That is used for safety goals within the expanse scope of operations. In that RSA, that end of protection ought to be built. This public including the private key-generation algorithm remains the usual several difficult pieces of RSA cryptography.

Key creation: RSA requires a public key and a private key. This public key can be understood by everyone also is used to encrypting information. Information encrypted by the public key can just be decrypted within a fair number of times utilizing the private key. The keys for the RSA algorithm do produce the subsequent way:

1. Pick two different prime figures p and q.

· With regard to the constancies of protection, the numbers p and q should be selected on the blind and should remain in the related bit-range. Effectively, prime numbers can be calculated to use a primary test.

2. Calculate the value n = p.q

· N implies the use of both public and private keys since the module. The key length is its range, which is usually expressed in bits.

3. Calculate f(n) = f(p) f(q) = (p − 1)(q − 1) = n-(p + q -1), where f is the function of Euler's totient. It's kept the interest secret.

4. Use a number e to ann-prime $1 < e < f(n)$ and gcd(e, f(n)) = 1; i.e., e and f(n).

E is set to release as an exponent of the public key.
E has a short bit and a low Hamming weight resulting in more effective encryption – most
216 + 1 commonly = 65.537. However in some settings, much smaller e values (such as 3) have been shown to be less safe.
5. Evaluate d as d order e−1 (mod f(n)); i.e. d is the multiplicative reverse modular of e (modulo f(n)).

· This is stated more clearly as: d solution given d

.

E iba 1 (mod f(n))

Using the extended euclidean method, it is often computed. Usage of the Modular Pseudo Code

Section of the prime numbers, interfaces a and n relate directly to e and f(n), respectively.

Item d is held as an exponent of the private key.
The public key consists of the modulus n and the public (or encryption) exponent e. The private key consists of the modulus n and the secret (or decryption) exponent d, which must be kept secret. p, q, and φ(n) must also be kept secret because they can be used to calculate d.

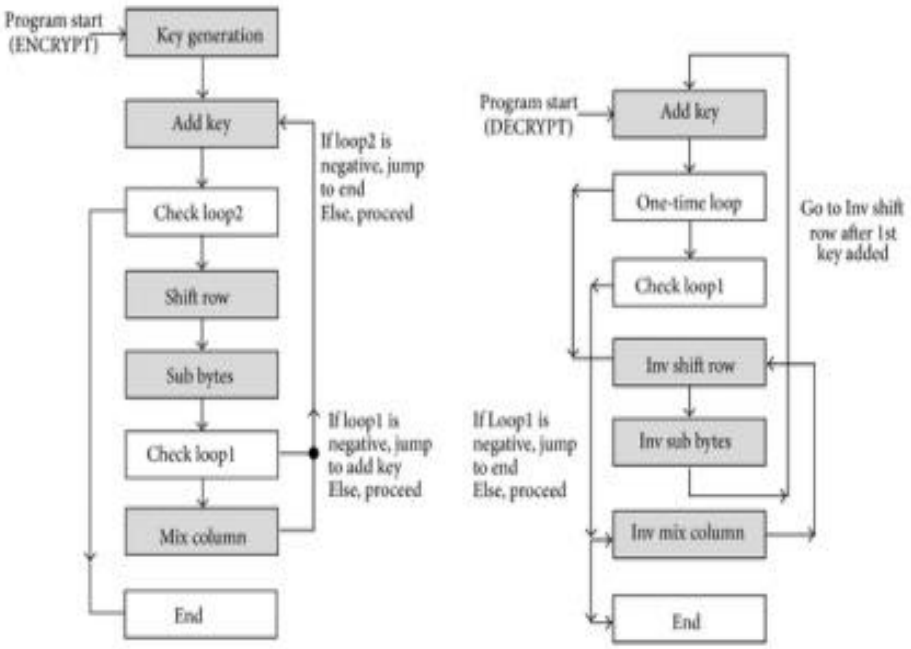
Since any common factors of (p-1) and (q-1) are present in the factorization of p*q-1, it is recommended that (p-1) and (q-1) have only very small common factors, if any besides the necessary 28]. Encryption: Alice transmits her public key (n, e) to Bob and keeps the private key d secret. Bob then wishes to send message M to Alice. He first turns M into an integer m, such that $0 \leq m < n$ and gcd (m, n) = 1 by using an agreed-upon reversible protocol known as a padding scheme. He then computes the cipher text C corresponding .This can be done competently, even for 500-bit numbers, using Modular exponentiation. Bob then transmits C to Alice.
Note that at least nine values of m will give in a cipher text C equal to m Decryption: Alice can recover m from C by using her private key exponent d via computing

AES:

(AES) algorithm depends upon a plan belief recognized as a substitution-permutation network, which means a mixture of change and transformation, including further that is quick in both software and device.
Some actions of AES algorithm into which a plain text is transformed in the ciphertext as follows:
(a) It uses 128-bit large plain text input block as input also keys range perhaps 128, 192 or 256-bits large also produce a product of 128- bit block.
(b) This 128-bit plain text allows an first step in which every byte of this case is connected among the round key utilizing XOR.
(c) Following producing the first round, plain text resides 10 steps if the key range of size 128-bit, 12 steps if key range of size 192-bit or 14 steps if key range of volume 256-bit. There are developing measures for each step: -
(i) Sub-Bytes change- it is a non-linear change, wherever every byte is substituted amidst different according to a lookup table.
(ii) Shift-Rows conversion- a change round wherever every row of this event is moved regularly an unknown amount of rounds.

(iii) Mix Column transformation- it is a mixing process which works on each column of the case, connecting four bytes in every column.

(iv) Add Round key in last, resultant matrix will endure a final round where sub byte, shift rows transformation are performed and round key is added into the resultant matrix. The complete process is shown in Figure.



Flow Chart Of AES:

ECC methodology for AODV:

A Mobile Adhoc Network (MANET) consists of a variety of nodes, neighbor nodes that are linked through radio connections. With limited power, bandwidth and scalability each node is mandatory. Several researchers are working to establish various techniques to ensure a safe route in MANET. As a consequence, an elliptic curve cryptography (ECC) technique is used to build a reliable shortest method for navigating the packets from origin to destination, it is famous because of its smaller focal duration and far less numerical there over encryption / decryption operations. It is more appropriate for power oblige devices. But it depends on secure routing protocol which requires a large number of Route Request (RREQ) packet in Ad-hoc on Demand Distance Vector (AODV) protocol and devours more power.

It works for:

•Advertised hop count includes multiple loop free routes.

• Preserves link-disjoint path or node-disjoint path.

• Processes several paths to carry each RREQ packet.

• Chooses routing path by considering the maximum number of sequences and thus keeps fresh route.

• Consider minimum hop count in case of the same sequence number.

Elliptic Curve Cryptography (ECC) is used to define a group of cryptographic tools and protocols where protection is based on the question of a discrete logarithm.

This is based on numeral sets and equations associated with elliptic curves[6].

The ECC operates at the following phases:

Key Generation

Global Public Elements a) Ep(a,b) elliptic curve with parameters a, b & p in the equation: Y2 mod p=(X3 +aX+b) mod p b) G is the base point on elliptic curve

1. User A Key Generation a) Select private key nA ; nA< n b) Calculate public, P = nA × G

2. User B Key Generation a) Select private key nB ; nB < n b) Calculate public , M = nB × G

3. Generation of Secret Key by user A P1 = k = nA × M

4. Generation of Secret Key by user B P2 = k = nB × P The two calculations produce the same result because nA × M = nA × (nB × G) = nB × (nA × G) = nB × P.

2) ECC Encryption

Consider a message ‘Pm’ sent from A to B.

‘A’ chooses a random positive integer ‘k’, a private key ‘nA’ and generates the public key PA = nA × G .

Chooses G, the base point selected on the Elliptic Curve Ep (a, b).

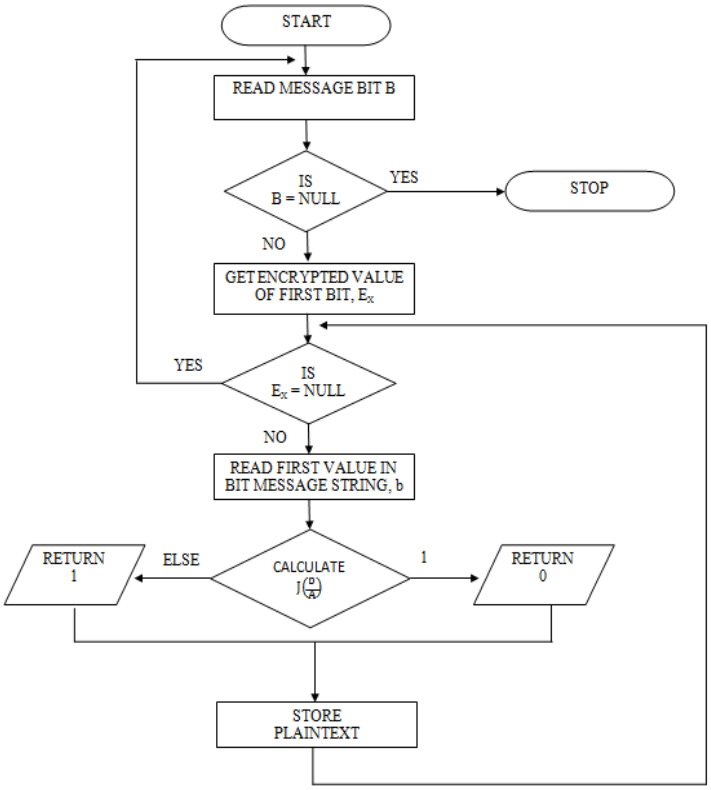
Produces the ciphertext ‘Cm’ consisting of pair of points Cm = { kG , Pm + kPB } = (C1,C2) where, PB= nB × G ,

The public key of B with private key ‘nB’ .

3)ECC Decryption

To decrypt the ciphertext, Cm, B multiplies the first point in the pair, C1 by B’s secret key.

Subtracts the result from the second point, C2, Pm + kPB - nB (kG) = Pm + k (nB G) – nB (kG) = Pm



DISCRETE LOGRITHM:

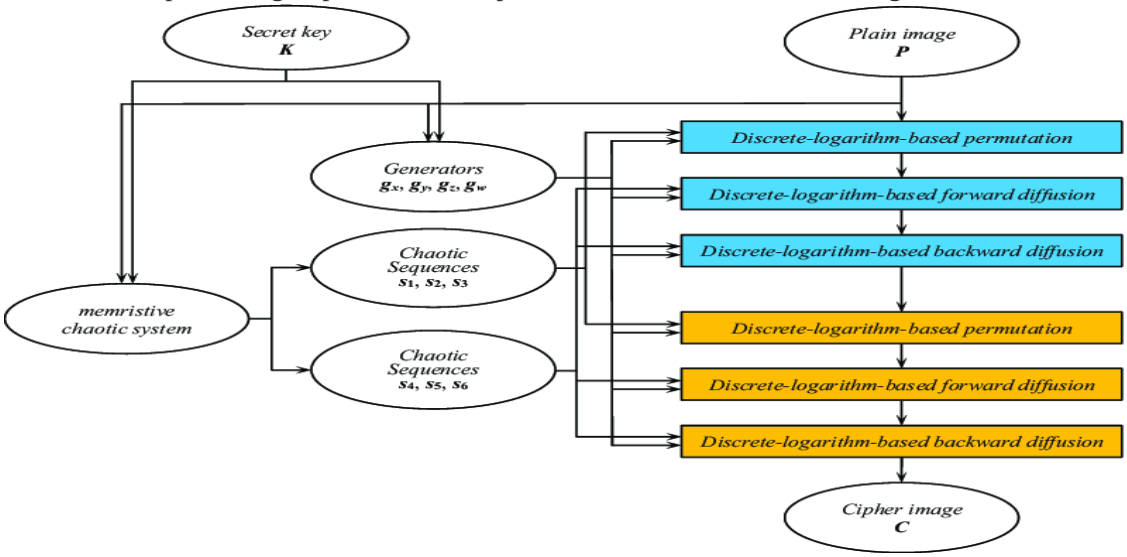
Discrete logarithms are fast measurable in some specific samples. But, neither effective technique is recognized for measuring them in common. In public-key cryptography, the security of several important algorithms based on assuming that the discrete logarithm query over thoroughly preferred combinations produces no effective explication.

This discrete logarithm query is thought to be computationally unmanageable. Such is, no effective standard algorithm is recognized for calculating discrete logarithms in common.

Some common algorithms for computing logarithm, a tremendous collection G is to establish b to more extensive and higher capabilities k till the wanted a is ascertained. This algorithm is seldom called test repetition. It needs working time linear in the volume of the group G and therefore exponential while the number of digits in this size of the combination. Accordingly, it is an exponential-time algorithm, useful only for petite collections G.

There are more sophisticated algorithms exist which is normally motivated by comparable algorithms as total integer factorization. Those algorithms work quicker than the naïve algorithm, any of them equivalent to the

square root of the scope of the group, and hence exponential in share the number of digits in the size of the



group.

Problem Statement:

Q1) What cryptographic method can we use for secure AODV Protocol?

Q2) How can you choose a good algorithm for how well you compare algorithms?

METHODOLOGY:

Attacks and Ad Hoc Networks Solutions

1) Replay attack on MANET

Intruder observes node recording appropriate broadcast messages and subsequently resends it to destination. MANET routing operations can be misused or interrupted by replay attack.

2) Attack by wormhole on MANET

Amongst the most complex and severe side attacks in MANET, attacker captures packets from one destination after the attacker uses private destination node to rebroadcast them to some other destination. This attacks would provide credibility and confidentiality to all communications.

a) ***Solution of Replay Attack***

b) The replay attack solution to protect in MANET is using asymmetric key as time stamp. Replay attack involve the receive message because once comparing the current time and time stamp. Time stamp setting is too far from the current time, so attacker can not be disturbed MANET operations.

c) ***Solution of Wormhole Attack***

Required location-based information, and public key infrastructure deployment. Transmitter sends HELLO message with current position and time. Receiver receives HELLO message to their neighbors and calculates distance from their neighbours. If the distance is greater than the discard or reject message within the transmission range.[16]

2) ***BlackHole Attack On AODV***

Black hole routing attack, a suspicious (attacker) sends fake routing information towards the other node, claiming that this would be a real route to destination through suspicious one that causes other real / good node data packets. Attacker may send out fake

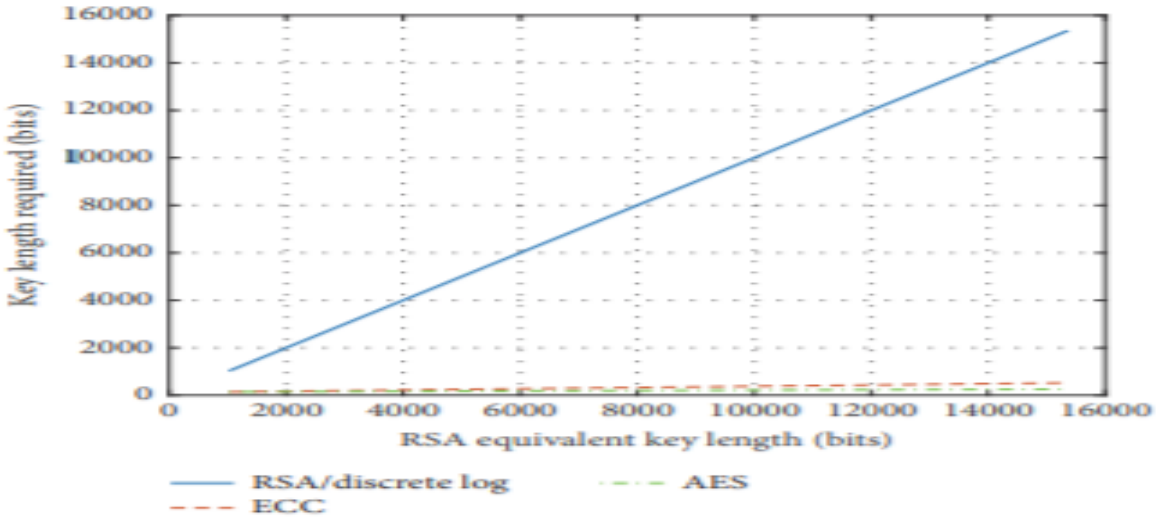
RREP to the source node (generate fake destination sequence number) and malicious node say that this is a real route to the destination. This can result in source node selecting fake path and passing through malicious (attacker) nodes. Attacker may misuse or discard the traffic nodes.[7]

MANET Attacks	Preventions/Solution
Replay Attack on MANET: **Replay attack can be misused or disturb routing operations on MANET.**	**Prevention:** Use Asymmetric Key.
Wormhole Attack on MANET: **This attack will launched against all the communication provide and confidentiality.[16]**	**Prevention:** Use Hello Message and calculate neighbor distance.

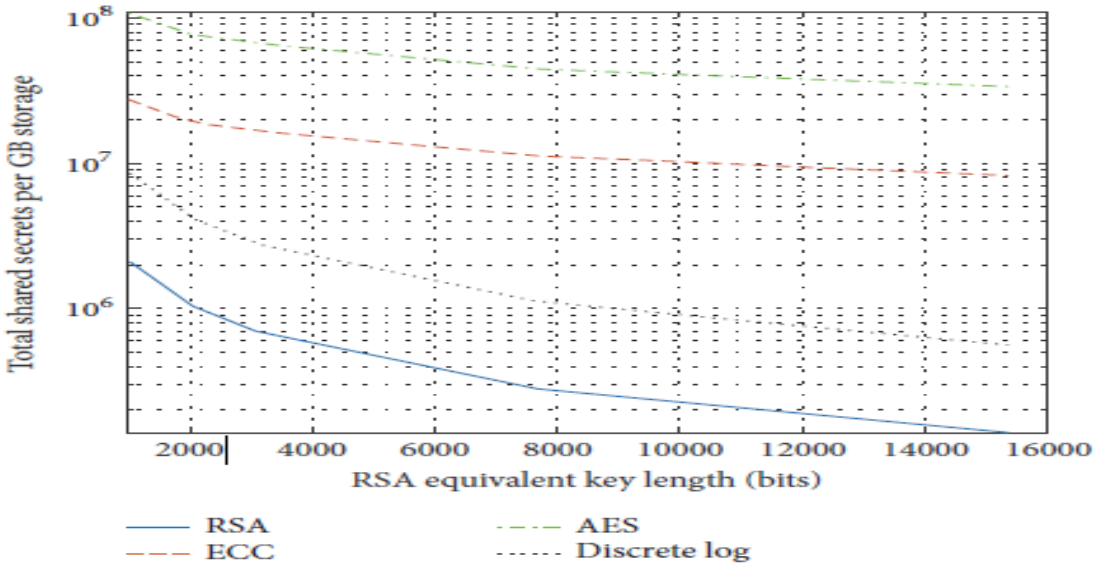
AODV Attack	Prevention/Solution
Black Hole Attack: **Attacker can misuse the node or discard the traffic.**	**Preventions:** Introduce request (CREQ) and (CREP) for confirm validity.

The RSA, ECC, AES, and discrete logarithm customs may all give an absolute level of protection, limited by the range of the encryption ciphers applied for any algorithm [8]. visually represents the necessary key range required by many encryption algorithms in a position to gain a level of protection equivalent to a detailed RSA key range. Within this state of some discrete logarithm classification, this similar key range of that prime p applied was defined utilizing the overall figure range algorithm as analyzed to RSA key lengths and was found to be approximately equal in requirement with RSA key N equivalent to a discrete log key 0.84 . ECC and AES uphold distinct benefits hereabouts over RSA and discrete log orders, as key extents for the last couple spread quickly as extended protection is required, while that key range: protection rate persists almost straight for ECC and AES. These more extended key measures of RSA and discrete log will likewise need extra bandwidth for public key change,
related to more abbreviated ECC public keys, including neither supplementary bandwidth cost is needed for AES. Accommodation necessities for pre-shared confidential data through a router, as described through the revised algorithms explained prior, are as follows:
(1) n-bit RSA wants a culmination of $4n$ bits per secret.
(2) n-bit ECC wants a whole of $2n$ bits per secret.

(3) n-bit AES wants a particular saved n-bit key.
(4) n-bit discrete log process includes a pre-shared secret
R, considered to hold of highest range n. Applying those conditions, in order with the key length wants to be depicted in Figure 1, it is likely to determine the smallest area obligations of every router for pre assigned confidential data. For instance, from Figure 1, we understand that a 2048-bit RSA or discrete logarithm key is the specific equivalent of a 224-bit ECC key or a 112-bit AES key. All shared secret collected by the router at that safety level would therefore need a preponderance of $(2048 * 4) = 8192$ bits for RSA and 2048 bits for discrete log but only $(224 * 2) = 448$ bits for ECC, or 112 bits for AES. Using these calculations, F Figure 2 shows the total amount of pre assigned intrigues which may be collected per gigabyte of retention for each assigned protection level and encryption algorithm(e.g., 8,000,000,000/8192 = 976,562 partook secrets per GB for 2048-bit RSA, or above 70 million received secrets per GB for the similar 112-bit AES).



Key length versus security for AES, ECC, RSA, and discrete log



Router pre shared secret storage requirements.

As anticipated in this Figure, a separate GB of router hidden data Storage allocation can contain hundreds of thousands of shared secrets, even when inefficient use of RSA algorithm at the 7680-bitLevel Security. Millions of shared secrets may be stored in that space when using AES. Whatever implementation options, shared hidden data storage is impossible to be a critical factor in the realistic implementation of routers. Encryption / decryption productivity is hard to measure for both the different algorithms, and has been influenced heavily by system design and software / hardware standardization. Usually, however, asymmetric cryptography ciphers such as AES would then give the speediest encryption/decryption time intervals. ECC provides significantly outstanding pair of keys generation productivity 6 networking and surveillance suitable material to RSA, with large prime numbers produced for RSA needing a few magnitudes the most time due to a much narrower ECC key, particularly at the 2048 or above RSA bit sizes. This could present a considerable problem in router systems with repeated key recharges. Consequently, production equipment can manage to fill only a moderately priced processing processor to distinctive preshared RSA keys (even a 1 GB chip could even hang thousands of pre-shared RSA certificates!), whilst also thousands of preshared RSA certificates are shared!
Symmetric key encryption keys will also involve actually trying to fill out extremely similar random data in the same chip couple. In particular, RSA encryption is probably quicker than ECC, while ECC decryption can be a few times faster than RSA, and both are generally sufficiently efficient to avoid a practical system bottleneck[15, 16]. Due to similarities in algorithm implementation, the discrete log method is assumed to offer a similar processing time as the RSA but is likely to take longer due to the multiple exchanges involved.

Security Bits	RSA	ECC	Key Generation (ms) ECC	Key Generation (ms) RSA
80	512	192	18	39
112	768	196	22	68
128	1024	239	22	241
192	2048	256	26	1542

Time Based Comparison for Encryption and Decryption:

Here we take encryption and decryption time of our proposed algorithms and observed the time which one is taking less time for encryption and decryption.

	Encryption	Decryption
AES	00:00:00.0107039	00:00:00.0005628
RSA	00:00:00.0260376	00:00:00.0221366
ECC	00:00:00.0010577	00:00:00.0003487
DL	0.002474784851072188	0.002550363540649414

We compare these four algorithms by giving those data set and implementation of these algorithms created in C# and Python Programming Language. System setup is just normal because we operate at a local level and not a high level we can use multiple or one python software platform depending on us sometimes here we use Jupiter notebook online and visual studio for our code implementation.

Following the attributes / parameters we use in our comparison:

Time: Here calculate the time taken for encryption and decryption the less time taken for encryption and decryption is considered as the best algorithm.

Avalanche effect: When a bit is changed or replaced with a critical alternative every minute the data is completely changed "the input change is a bit of a change". Avalanche High Result is ready for the algorithm.

Key Size: Here we also compare these four algorithms by their key sizes and analyze which one is best.

Time complexity: Displays the amount of time taken by the implementation algorithm. Reduce the complexity of the time betting algorithm.

Attributes	AES	RSA	ECC	DL
Avalanche effect	0.690879	0.456789	0.5333333	0.4545

Time took for encryption	00:00:00.0107039	00:00:00.0260376	00:00:00.0010577	0.002474784851072188
Time took for decryption	00:00:00.0005628	00:00:00.0221366	00:00:00.0003487	0.002550363540649414
Key Size	128,192,256	512,1024,2048	160,192,224,256	1024
Rounds	10,12,14	1	1	1
Time complexity	O(1)	O(N^3)	O(N^2)	$2^O((\log k)^3)$

Conclusion:

AES key pair data, efficiently composed of a random bitstream, could be produced and pre - loaded on to the devices far quicker unlike RSA, ECC or discrete logarithm key pairs as well as provide greater security than asymmetric ciphers of equivalent lengths. Alternatively, it is possible to use a fusion of both structures, providing on-demand verification where required and efficient then no verified encrypted connection.

Based on a comparison we discovered that the ECC is smarter than most of the other methodologies as we contrasted it with the time

Cryptography of the elliptic curve is probably better with most reasons but not for all.

The major benefit of a ECC is that you'll have smaller keys for much the same degree of protection, especially at high levels of safety (AES-256 ~ ECC-512 ~ RSA-15424). That's also due to flashy factoring methodologies, such as the Number Field Sieve

.

Like with any data security, the certain encryption network is exposed to negotiation to computer machine as a very last considering. Whenever an intruder is able to view the secret key stored data mostly on security hardware of the router, then even the most robust encryption mechanism can be exposed and precautions must be taken during hardware development and delivery to ensure that such keys are not copied or accessed prematurely

References:

[1]https://mafiadoc.com/a-comparative-evaluation-of-algorithms-in-the-implementation-of-an-_5baa469c097c47586d8b4612.html94 93 25.59

[2]https://www.hindawi.com/journals/scn/2017/1467614 88 87 19.11

[3]http://downloads.hindawi.com/journals/scn/2017/1467614.xml 85 84 26.78

[4]https://www.researchgate.net/publication/306018436_ADVANCED_SECURE_METHOD_FOR_DATA_TRANSMISSION_IN_MANET_USING_RSA_ALGORITHM 45 43 8.61

[5]https://www.scribd.com/document/251910634/Enhancing-data-mining-techniques-for-secured-data-sharing-and-privacy-preserving-on-web-mining 41 40 10.8

[6]https://www.slideshare.net/J4R/homomorphic-authenticable-ring-signature-mechanism-for-public-auditing-on-shared-data-in-the-cloud 41 39 16.42

[7]https://www.researchgate.net/publication/330653876_Elliptic_Curve_Cryptography_Based_Data_Transmission_against_Blackhole_Attack_in_MANET 37 34 7.4

[8]https://infogalactic.com/info/RSA_(cryptosystem) 37 34 13.38

[9]http://voip.kryptotel.net/rsa-encryption 35 34 18.32

[10]https://www.cnblogs.com/chucklu/p/5197231.html 32 31 18.39

[11]https://www.eandbsoftware.org/rsa-cryptosystem-and-the-prime-numbers 31 29 15.47

[12]https://pastebin.com/wpQxyegi 26 24 11.29

[13]https://www.ijser.org/researchpaper/A-Comparative-Analysis-of-AES-and-RSA-Algorithms.pdf 24 20 14.69

[14]https://www.cnblogs.com/yuanjiangw/p/10108686.html 22 21 13.6

[15]https://www.thefreelibrary.com/A+Comparative+Evaluation+of+Algorithms+in+the+Implementation+of+an...-a0548321280 21 20 19.25