ISSN: 2229-6948(ONLINE)
DOI: 10.21917/ijct.2012.0072

EFFICIENCY AND EFFECTIVENESS ANALYSIS OVER ECC-BASED DIRECT AND INDIRECT AUTHENTICATION PROTOCOLS: AN EXTENSIVE COMPARATIVE STUDY

K. Thilagavathi[1] and P.G.Rajeswari[2]
[1]*Department of Mathematics, Kongunadu Arts and Science College, India*
[2]*Department of Science and Humanities, Easa College of Engineering and Technology, India*
E-mail: drpgrajeswari@gmail.com

Abstract
Elliptic curve cryptography finds enormous applications because of its security offering using the remarkable property of elliptic curve. The Elliptic curve cryptography finds enormous applications in almost all the emerging areas. However in mobile networks, the usage of elliptic curve cryptography is limited. Moreover, the operation of mobile networks in an un-trusted environment increases the significance of the usage of security protocols. To provide a secure environment, an improved authentication protocols are required as the menacing effects increasing. Hence, in the previous works, we have proposed two authentication protocols. One of the protocols performs direct authentication and the other one performs indirect authentication. However, the performance of both of them has to be analyzed. Hence in this paper, a comparative analysis is made between the two authentication protocols. The analysis is done empirically as well experimentally. For performance analysis, the efficiency measures such as computational overhead, communication overhead, storage overhead and total computational complexity and the effectiveness measures such as replay attack, guessing attack and Stolen-Verifier attack are considered.

Keywords:
Elliptic Curve Cryptography (ECC), Direct Authentication Protocol, Indirect Authentication Protocol, Efficiency, Effectiveness

1. INTRODUCTION

The rapid progress in wireless mobile communication technology and personal communication systems has prompted new security questions. Since open air is used as the communication channel, the content of the communication may be exposed to an eavesdropper, or system services can be used fraudulently. In order to have reliable proper security over the wireless communication channel, certain security measures need to be provided [12].The mobile environment aggravates some of the security concerns and threats. Mobile users will use resources at various locations and this may be provided by different service providers. Integrity and confidentiality of information stored on the mobile appliance is another important concern. A competing system that has emerged recently is ECC [10].

ECC is a public key cryptography system superior to the well-known RSA cryptography: for the same key size, it gives a higher security level than RSA [3] [14]. From the time when the use of elliptic curves in public key cryptography was suggested in 1985, increasingly effective implementations of ECC systems have been developed. Today, these systems are as fast as systems based on integer factoring with same key length [1]. Elliptic curves have been broadly used in the design of cryptosystems [2] [17]. ECC has been adopted in a wide variety of applications from digital certificates in web server authentication to embedded processors in wearable devices [4] [19]. Elliptic curve cryptography plays an important role in authentication and encryption protocols [5] [18].

ECCs are used commonly in constrained environments, such as portable and wireless devices, as a small-area, low-energy alternative to the RSA cryptosystem. The primary application of ECC is secure key agreement and digital signature generation and verification [6]. In both of these applications the primary optimization criterion from the implementation point of view is the minimum latency (rather than the maximum throughput) [7]. An elliptic curve is a type of cubic curve whose solutions are confined to a region of space that is topologically equivalent to a torus [8]. The crucial property of an elliptic curve is that we can define a rule for adding two points which are on the curve, to obtain a third point which is also on the curve. This addition rule satisfies the normal properties of addition [9]. Elliptic curve cryptosystems require less computational power, memory, communication bandwidth and network connectivity [15].

The main attraction of ECC over RSA and DSA is because they take sub-exponential time to solve the underlying hard mathematical problem in ECC (the elliptic curve discrete logarithm problem (ECDLP) while the best known algorithm takes full exponential time [11]. The ECC is intended to be used in the security layer to automatically encrypt/decrypt all data that flows to or from the application layer. We develop a front-end program to demonstrate the functionality of the ECC. This front-end program utilizes the ECC to encrypt a plain text data file. The program can be used on /computing devices in order to store confidential data securely onto the device. In addition to encryption and decryption, ECC can be applied to other applications such as Digital Signatures, Mutual Authentication, and Secure Data Transmission [10]. ECC is becoming the mainstream cryptographic scheme in all mobile and wireless devices. Smart cards are one of the most popular devices for the use of ECC and many manufacturing companies produce smart cards that make use of elliptic curve digital signature algorithms [13]. Elliptic curve cryptography has become the cryptography of choice for mobile computing and communications devices due to its size and efficiency benefits [16]. Some of the works that have been done with Elliptic Curve Cryptography is reviewed in the following section.

2. RELATED WORKS

Pathak et al. [20] have proposed a new modified algorithm called ‘Direct Recoding Method’ for computation of signed binary representation. Their proposed method has been more efficient compared to other standard methods such as NAF, MOF and complementary recoding method. Rahila Bilal et al. [21] has discussed that Elliptic Curve Cryptography has been

one of the most interesting research topic in VLSI. FPGA based architecture for elliptic curve cryptography coprocessor, which has promising performance in terms of both Space Complexity and Time Complexity has been proposed in their paper. The modules have been simulated using Modelsim SE software and synthesized using Xilinx ISE 9.2i software. Experimental results have shown that ECC coprocessor realized in their architecture can speed up an elliptic curve scalar multiplication suitable for low area constraint applications and very high speed applications.

Adnan Abdul-Aziz Gutub [22] has designed and modeled an improved parallel elliptic curve processor. The Jacobian coordinates system has been adjusted by interacting point double and point add operations. Results have shown that their proposed modified Jacobian design gives higher speed and cost (AT2) showing attractive research direction. Rahila Bilal et al. [23] have presented an article on the design of a crypto processor to implement the Elliptic curve point multiplication .They have investigated the potential of the hardware/software co-design to realize a flexible – low resource Elliptic Curve Cryptography (ECC) processor over binary fields GF(2163) on FPGA platforms. The implemented processor has presented a good performance, which is very suitable for applications that require high speed. Portilla et. al.[24] has described how the reconfiguration possibilities of the system could be used to adapt ECC parameters in order to increase or reduce the security level depending on the application scenario or the energy budget. According to the results, the FPGA-based ECC implementation has required three orders of magnitude less energy, compared to a low power microcontroller implementation, even considering the power consumption overhead introduced by the hardware reconfiguration.

Kumar et al. [25] have proposed a Region-Based structure that enables efficient and secure peer-to-peer information sharing over MANETs. The implementation has shown that the proposed scheme as secure, scalable, efficient, and adaptive to node mobility and provider of reliable information sharing. Rajaram Ramasamy et. al. [26] have illustrated encryption / decryption involving the ASCII value of the characters constituting the message, and then it has been subjected to the knapsack algorithm. They have compared their proposed algorithm with RSA algorithm and shown that their algorithm is better due to the high degree of sophistication and complexity involved. It has been almost infeasible to attempt a brute force attack. PrasannaGanesan [27] has highlighted that the existing authentication protocols, based on RSA asymmetric cryptography, have not been appropriate for such devices due to their limitations in computing power, memory capacity, key sizes and cryptographic support. Therefore, an efficient protocol for resource constrained platforms that achieves a level of security similar to the one achieved by the protocols that are then in use has been designed and implemented. This protocol has been based solely on Elliptic curve asymmetric cryptography and the results have proved that the performance achieved has been good compared to RSA.

From the above literature review, it can be seen that the ECC is utilized in number of applications. But, the works that are done for networks security, especially mobile networks is less. Hence, in our research we have utilized ECC to develop authentication protocols for secure mobile networks. In the first work, we have proposed a direct authentication protocol and in the second work, we have proposed an indirect authentication protocol. Both the protocols have their own advantages and disadvantages over the others. A very brief description about the two protocols is given in the subsequent Section.

3. PROPOSED DIRECT AND INDIRECT AUTHENTICATION PROTOCOLS

The proposed direct [38] and indirect authentication protocols [39] have utilized ECC for key generation and the protocols are developed in such way that it can authenticate the user or information requester very effectively.

3.1 DIRECT AUTHENTICATION PROTOCOL

In the situation of requesting information by base station to the user node, the user node need to authenticate and then only it has to send the information to base station only if it is valid. The protocol flow is given in Fig.1 and the procedure is described below.

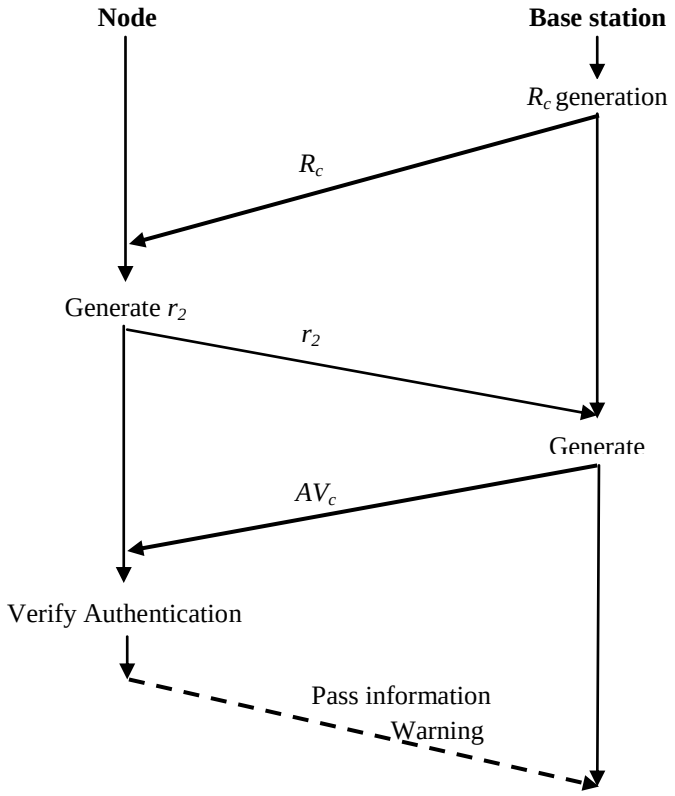

Fig.1. Protocol flow for Direct Authentication

The set of procedure followed in the Direct Authentication are as follows,

- Initially, base station generates a random number r_1
- Then, base station calculates the requesting code R_c as,

$$R_c = r_1 * B \quad (1)$$

- Base station sends R_c to user node
- Node generates a random number r_2 and sends it to base station

- Base station generates authentication-verifying code AV_c, which can be calculated as

$$AV_c = r_1 + (r_2 * K_s) \quad (2)$$

- Node performs authentication as,

$$(AV_c * B) - (r_2 * K_p) = R_c \quad (3)$$

3.2 INDIRECT AUTHENTICATION PROTOCOL

In the Indirect authentication protocol, two servers are utilized to perform authentication, one is main server and the other one is authentication server. The protocol flow is given in Fig.2 and the procedures are described below.

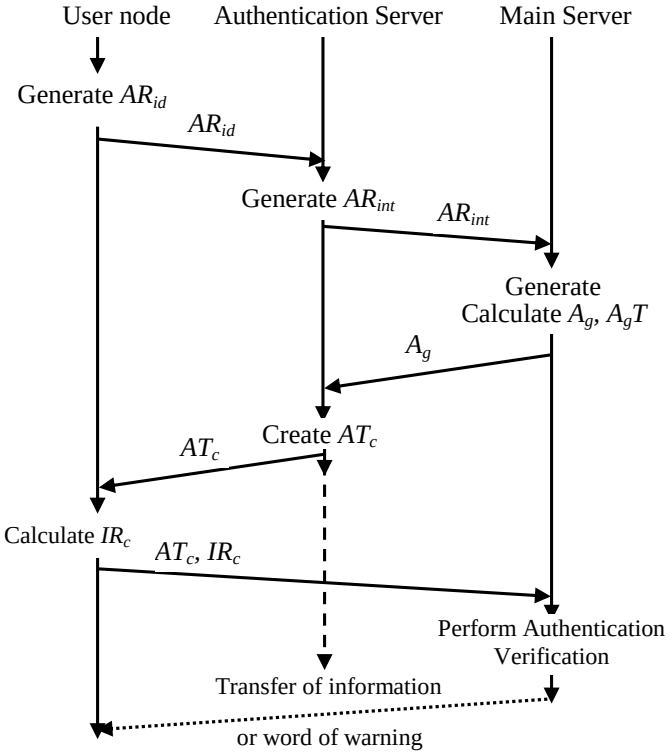

Fig.2. Protocol flow of Indirect Authentication protocol

The procedures are given below:

- User generates AR_{id}, which is a random number, and sends it to Authentication Server
- Authentication Server generates AR_{int}, another random number, and sends it to Main Server
- Main Server generates Ack_{rand} and calculates A_g and A_gT as follows,

$$A_g = B[AR_{int} * K_s(M.S.)] \quad (4)$$

$$A_gT = [Ack_{rand} * K_p(A.S.) - AR_{int} * K_p(M.S.)] \quad (5)$$

- Main Server sends A_g to Authentication server
- Authentication Server calculates AT_c as given below and sends it to user node

$$AT_c = [AR_{id} + Ack_{rand} * K_s(A.S.)]B - A_g \quad (6)$$

- User node calculates IR_c as given in Eq.(7) and sends IR_c and AT_c to Authentication Server,

$$IR_c = AR_{id} * B \quad (7)$$

- Main Server performs authentication as,

$$AT_c - A_gT = IR_c \quad (8)$$

Hence the authentication process is done using the proposed indirect authentication protocol. A comparative analysis between the direct and indirect authentication protocol empirically as well experimentally is detailed in the following Section.

4. THE DIRECT AND INDIRECT AUTHENTICATION PROTOCOL: A COMPARATIVE STUDY

A wide experimental analysis as well as empirical analysis was made between the proposed techniques. An overall picture about the analysis performed in the proposed technique is given in Fig.1. As in Fig.1, the comparative analysis can be divided into two, namely, efficiency validation and effectiveness validation. Generally, the term efficiency, which is a significant performance measure, is directly related to the temporal performance of any technique. The effectiveness, which is another performance measure, directly related to the performance of the technique in fulfilling the purpose/requirement.

As discussed in [39] [38], the elliptic curve cryptography is better than the RSA public key cryptography because even though the RSA exploits the largest prime numbers for providing security that can also be hacked by the hackers. In order to provide high security than RSA, the elliptic curve cryptography is utilized in this mechanism. In this ECC the elliptical point is utilized and here both the protocols are utilizes the ECC mechanism. In direct protocol, the node can be access the base station directly without any intermediary nodes, it may provide easy access but security may less due to its direct access. But in the indirect protocol, two servers are involved and hence it is difficult to hack the information even though the hackers hacked a severs' information or they hacked both the servers they are unaware of the relationship among them. Hence the indirect authentication protocol provides more security than the direct protocol.

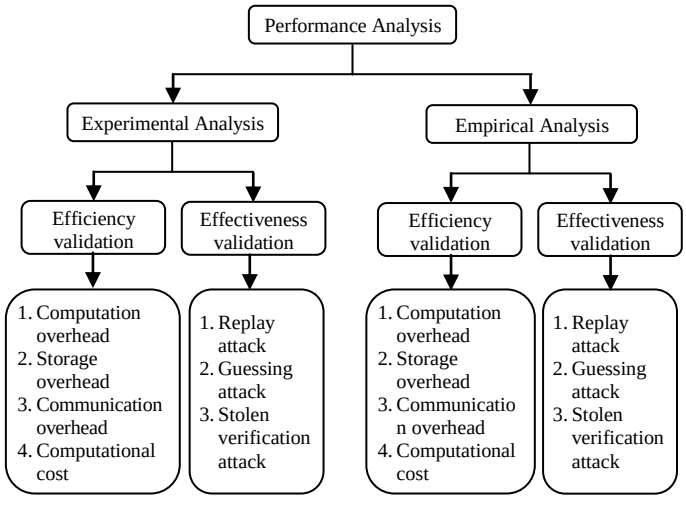

Fig.3. Structural view of the performance metrics to analyze the protocols

Here, we empirically as well as experimentally analyze the performance of the technique in terms of efficiency and effectiveness. To accomplish this, we consider four significant efficiency measures. They are,

- Computational Overhead
- Total Computational Cost
- Communication Overhead
- Storage Overhead

For effectiveness validation, we consider three major attacks in any networking environment. They are

- Replay attack
- Guessing attack
- Stolen-Verifier attack

Prior to analyzing the techniques using these measures, a general as well as technical description of these measures is discussed in the following.

Computational Overhead:

Overhead is normally said to be some mix of too much or implicit computation time, memory, bandwidth, or alternative resources that are necessary to accomplish a specific objective. The traffics are computed using protocols by considering the signaling traffic of the protocols. It is observed that the base protocol has the lowest traffic, which implies that the complexity is low both at MNs and Mobility Agents. In addition the bandwidth required is low. But, it has poor security. Yang's protocol and self certified time invariant protocols have highest overhead and strong security [29]. Real time communication encryption is based on public key, so the protocol has lower communication and computation overheads. In symmetric message authentication scheme, the MAC of the sent encrypted data is computed by the receiver, packet modification during transit is prevented to restrict the pollution attack. Then a random number k is obtained by decrypting. After that, this k is compared with the random number k sent by the receiver. This achieves data authentication as a particular sender is identified by the set index. When members join the multicast group, the group manager circulates the used symmetric key created along with the private decryption key to the group members. Large computation overhead is required by this protocol [32].

Communication Overhead:

The amount of data transmitted between the reader and authentication server determines the cost of communication. A low communication cost will improve the performance of the application by lessening the network traffic and workload on the server [29]. Both computation and communication performance is improved. Due to the possibility of using resource broker for batching authentication sessions, significant improvement is achieved in communication [28]. The tag-reader is found to be most efficient when the number of interchanged messages for accomplishing mutual authentication is considered. Four rounds may be regarded as the appropriate number of rounds for mutual authentication in RFID environments due to the fact that low cost tags are passive and communication can only be started by a reader [30]. Their model incurs large communication overhead in node re-authentication, though sink or base station is not necessitated by the mobile node and the authentication protocol supporting node mobility, for authentication and key distribution. The communication overhead between a sink and the base station can be decreased by an efficient untraceable re-authentication and key distribution protocol. [33]. An ACK message containing two fields: a node id and a MAC, is included by the communication overhead for confirming a pair wise key [34]. Only four messages are used by Gossamer for accomplishing mutual authentication and integrity protection. A "hello" and IDS message are sent by means of the channel in the identification phase. The authentication phase transmits the messages A||B||C and D. Therefore, if 5 bytes are assumed for the "hello" message, then an aggregate of 424 bits are transferred over the channel [35].

Storage Overhead:

Assuming L-bits as the size of all components, an L-bit index pseudonym (IDS) and a four L-bit component associate key (K) has to be stored by each tag. In addition, a distinct L-bit identification number (ID) has to be stored by the tag. It necessitates a memory of 6L bits as the reader has to store the same information [30]. Their use on devices having restricted resource is impractical because asymmetric cryptographic mechanisms have increased computation, communication, and storage overhead [36]. A priori high execution time of the CPU, battery consumption and storage capacity of the mobile device are necessary for its use, even though the use of cryptographic operations improve network security [37].

Total Computational Cost:

The overall execution time taken by the any protocol/process can be simply defined as total computational cost. In some point, the computational cost includes the requirements needed to execute the protocol. The requirement may be hardware/software modules.

Replay Attack:

A replay attack is a kind of network attack which fraudulently or maliciously repeats or delays the legitimate data transmission. A replay attack happens when a stream of messages between two parties is copied by the attacker and the stream is replayed to one or more of the parties.

Guessing Attack:

A password guessing attack happens when log on to a computer or network is repetitiously attempted by an illegal user through guessed username and password. Several password guessing programs are available in the internet which tries to break passwords. The diverse types of password guessing attacks are as follows:

Brute force attack:

A brute force attack or exhaustive key search is an approach that can hypothetically be used against any encrypted data by an attacker [1] if he is not able to make his/her job easier by exploiting some weak-point in the encryption system.

Dictionary attack:

A dictionary attack in cryptanalysis and computer security is a method for defeating a cipher or authentication system that attempts identification of decryption key or pass-phrase of authorized system by searching the probable possibilities.

Stolen-Verifier attack:

Attackers always target the servers because several secrets of customers are stored in their databases Hence, majority of the available password authentication schemes, stores the verifier of the user (e.g., plaintext passwords or hashed passwords) instead of the bare password of the user in the server to decrease the

security breach in case the server is compromised. Stolen-verifier attack is said to be the masquerading attempt made by the adversary as a legitimate user by directly using the password-verifier which is stolen from the server. Stolen-Verifier attack is considered as a critical problem in authentication schemes. So, instead of the clear text of passwords, verifiers of the passwords of users are stored by servers. Stolen-Verifier attack is an objectionable action performed by an attacker who has obtained a verifier for a particular user by compromising the password database. Security schemes are strongly needed to defy this attack as attacks carried out by internal users have become increased and more critical nowadays. Alleviating the pressing danger to the authenticate user is the major objective of any authentication scheme that safeguards against the Stolen-Verifier attack. The launching of a guessing attack is common by an adversary who has a password-verifier. This attack scheme is not good for masquerading as the legal user or system. The merit of verifier-based authentication mechanism is due to the fact that password guessing consumes the time of the attacker when the verifier is stolen. Although it can resist the stolen-verifier attack, it succumbs to other easier attacks such as denial-of-service attack and replay attack. On the other hand, the strong-password authentication schemes are prone to stolen verifier attacks and guessing attacks. If verifiable information cannot be stolen if verification table or password table that contain this information are not stored in servers or registration center. So such methods can resist against the stolen verifier attack. Many protocols and methods are proposed to protect the stolen-verifier problem.

4.1 THE PROPOSED PROTOCOLS: AN EMPIRICAL ANALYSIS

In this section, the protocols are empirically analyzed for the efficiency and effectiveness measures.

4.1.1 Efficiency Measures:

The primary intention of the proposed authentication protocols is to authenticate the base stations with reduced computational complexity. As the aforesaid efficiency measures play a vital role in analyzing the computational complexity of the techniques, an empirical analysis of these techniques is given below.

Computational Overhead:

Practically, the computational overhead can be defined as the average complexity that occurs in computing every authentication parameters and steps that are to be performed in the protocols. The direct authentication protocol [38] is very simple and it involves performance of extremely small steps and computations using the ECC concepts. However, the indirect authentication protocol [39] involves performance of numerous steps and computations, both in the main server and the authentication server. Consequently, compared to the direct authentication protocol, the computation overhead is really high for the indirect authentication protocol.

Total Computational Cost:

Here, the total computational cost can be defined as the total executable time to compute and execute the entire authentication protocol. In [38], the protocol accomplishes the generation/calculation of authentication variables/authentication parameters in three steps, transfer of authentication variables/authentication parameters between the node and the base station in three steps and finally an authentication process and the resultant transfer. However, in [39] seven authentication parameters/variables have been calculated. Two steps of transfers have been performed between Authentication server and Main server as well as between the Authentication server and the user node. In addition, a single step of authentication parameters/variables is carried out between the user node and the Main server. Eventually, the authentication process is performed at the Main server and the outcome is transferred to the user node. Hence, the total computation complexity (or) computational cost estimated can be very high for the protocol proposed in [39] compared to that of the protocol proposed in [38].

Communication Overhead:

In our case, the communication overhead can be stated as the mean time taken in transferring every authentication parameter/variable that is involved in the authentication protocol. In [38], only three steps are carried out in transferring the authentication protocols/variables between the base station and the user node. However in [39], four steps are carried out between the Authentication server and Main server as well as between the Authentication server and the user node. In addition, a single step of parameters transfer is done in between the user node and the Main server. Thus, totally five steps of communication is carried out between the authentication members i.e. user node, Main server and the Authentication server. Ideally, it can be estimated that the protocol, which is proposed in [39], has an increased communication overhead of around 60% over the protocol that is proposed in [38].

Storage Overhead:

The protocol proposed in [38] intends to generate four authentication parameters and determine three authentication parameters. In [39], five authentication parameters are generated and then six authentication parameters are determined in every protocol member. Thus generated and determined parameters needs to be stored in the concerned protocol member, which increases the storage overhead as storage overhead is considered as the complexity due to the storage of the parameters that are involved in the authentication protocol. This makes the storage overhead of [39] as much as 63% (estimated) more than that of protocol [38].

The empirical analysis results are tabulated in Table.1, which indicates the parameters and processes that are involved in the protocols.

Table.1. Parameters/Processes influencing in the efficiency measures

Performance measures	Direct Authentication Protocol	Indirect Authentication Protocol
Computational overhead	1)Generation of B 2)Generation of r_1 3)Generation of K_s 4)Calculation of R_c 5)Calculation of K_p 6)Generation of r_2 7)Calculation of AV_c 8)Verify	1) B Generation 2) Generation of AR_{id} 3) K_s Generation 4) AR_{int} Generation 5) K_pCalculation 6) Ack_{rand} Generation 7) A_g Calculation 8) A_gT Calculation

	Authentication	9) AT_cCalculation 10) IR_c Calculation 11)Authentication verification
Communication overhead	1) R_c from Base station to Node 2) r_2 from Node to Base station 3) AV_c from Base station to Node 4) Resultant for Authentication verification	1) AR_{id} from User node to Authentication server 2) AR_{int}Authentication server to Main server 3) A_g Main server to Authentication server 4) AT_cAuthentication server to User node 5) AT_c' IR_cUser node to Authentication server 6) Resultant for Authentication verification
Storage overhead	B, r_1, K_s, R_c, K_p, r_2, AV_c	B, AR_{id}, K_s, AR_{int}, K_p, Ack_{rand}, A_g, A_gT , AT_c, IR_c
Total computational cost	B, r_1, K_s, R_c, K_p, r_2, AV_c, R_c, r_2, AV_c	B, AR_{id}, K_s, AR_{int}, K_p, Ack_{rand}, A_g, A_gT , AT_c, IR_c, AR_{id} , AR_{int}, A_g, AT_c, AT_c' IR_c

4.1.2 Effectiveness Measures:

An empirical analysis of the proposed protocols in terms of the effectiveness measures is described in the following sections

Replay attack:

It is well known that the replay attack is an attack that is done by hacking certain information during the time of conversation between two communicating partners and then using the hacked information in the subsequent communications. In protocol [38], information hacking can be done possibly in any of the three following steps of communications,

i. Transmission of R_c from base station to user node
ii. Transmission of r_2 from user node to base station
iii. Transmission of AV_c from base station to user node

In case (i) and (ii), the parameters are arbitrary. Hence the probability of using these parameters, after it is hacked, is very less. As case (iii) parameter is the final parameter that needs to be transferred in the authentication protocol it is not possible to use this information in the same protocol even if it is hacked. Hence, it can be absolutely confirmed that the proposed protocol [38] is robust against replay attack.

In protocol [38], information hacking can be done during the transfer of the following parameters

i. Transmission of AR_{id} from user node to Authentication Server
ii. Transmission of AR_{int} from Authentication Server to Main Server
iii. Transmission of A_g from Main server to Authentication Server
iv. Transmission of AT_c from Authentication server to user node
v. Transmission of AT_c and IR_c from user node to Main Server

Cases (i) and (ii) deal with the transfer of arbitrary parameters and so probability of hacking is very low. In cases (iii), (iv) and (v), the transferred parameters are determined using contribution of arbitrary parameters. Hence, even if these parameters are hacked, they cannot be used for replay attack. However, the robustness is relatively lower than that of the protocol proposed in [38].

Guessing Attack:

In protocol [38], guessing of arbitrary parameters is very difficult. The only parameter that can be guessed is AT_c, however it is a contribution of arbitrary numbers. Hence, it is robust as long as AT_c is not guessed.

Protocol [39] exhibits more robustness than protocol [38] because it performs authentication not only with the parameters of the user node but also with the parameters of the Authentication server. Though the hacker guesses a parameter/ user credential, it cannot be used for pretending him/herself as the authenticated user. This is mainly because of the involvement of two servers in the authentication process. The strong point is that it is practically impossible to simultaneously hack information from the user, Authentication server and the Main server.

Stolen-Verifier Attack:

In both the protocols, no parameters are constant for any user when they are trying to access the information. This can be asserted ideally that the protocols are more robust for the Stolen-Verifier attack. Moreover, the protocol [39] performs the authentication using two servers. Even any credential of the information requester is hacked by either of the server; it is not acceptable when working with the other servers. This further claims that the protocol [39] is more robust than the protocol [38] against the Stolen-Verifier attack.

4.2 THE PROPOSED PROTOCOLS: AN EXPERIMENTAL ANALYSIS

In order to experimentally evaluate the proposed protocols [38] and [39], several efficiency performance measures are calculated and compared. The efficiency measures that are determined for the protocols are given in Table.2, 3 and 4.

Table.2. The efficiency measures (i) Computational Overhead, (ii) Communication Overhead, (iii) Storage Overhead for Direct Authentication protocol

Table.2(i)

Computational Overhead	
Parameters	**Time (sec)**
R_c	0.0
r_2	0.0
AV_c	0.0

Table.2(ii)

Communication Overhead	
Parameters	**Time (sec)**
R_c transfer from Base station to Node	$3.01050914 \times 10^{-11}$
r_2 transfer from Node to Base station	$2.51100929 \times 10^{-10}$
AV_ctransfer from Base station to Node	$2.81294329 \times 10^{-10}$

Table.2(iii)

Storage Overhead	
Parameters	**Bytes**
$R_c$generation	8
Generate r_2	8
Generate AV_c	16

Table.3. The efficiency measures (i) Computational Overhead, (ii) Communication Overhead, (iii) Storage Overhead for Indirect Authentication protocol

Table.3(i)

Computation Overhead	
Parameters	**Time (sec)**
AR_{id}	0.0
AR_{int}	0.0
Ack_{rand}	0.0
A_g	0.0
A_gT	0.0
AT_c	0.0
IR_c	0.0

Table.3(ii)

Communication Overhead	
Parameters	**Time (sec)**
AR_{id} transfer from User node to Authentication server	$3.10000914 \times 10^{-11}$
AR_{int} transfer from Authentication server to Main server	$2.18999929 \times 10^{-10}$
A_g transfer from Main server Authentication server	$1.50000012 \times 10^{-11}$
AT_c transfer from Authentication server to User node	$3.14568212 \times 10^{-11}$
AT_c, IR_c transfer from User node to Main server	$2.74572542 \times 10^{-11}$

Table.3(iii)

Storage Overhead	
Parameters	**Bytes**
Generate AR_{id}	8
Generate AR_{int}	8
Generate Ack_{rand}	8
Calculate A_g, A_gT	32
Create AT_c	16
Calculate IR_c	16

In determining the performance measures that are given in Table.2 and Table.3, we have not determined the measures of keys as well as authentication process, because these measures are common for any protocols. When determining the other parameters, it can be visualized that the direct authentication protocol shows more efficiency rather than the indirect authentication protocol in terms of communication overhead and the storage overhead. In reference to computational overhead, both the protocols show similar performance. The performance in terms of total computational complexity is tabulated below.

Table.4. Total Computational Complexity for the proposed Direct and Indirect Authentication Protocols

Protocols	Total Computational Complexity (sec)
Direct Authentication Protocol	3.013
Indirect Authentication Protocol	8.781

From Table.4, it can be seen that the direct authentication protocol is more efficient than the indirect authentication protocol.

In order to visualize the performance of the protocols [38] and [39] five different experiments were conducted. Each experiment is comprised of ten rounds of attacks. The robustness of the protocols at every experiment is plotted below.

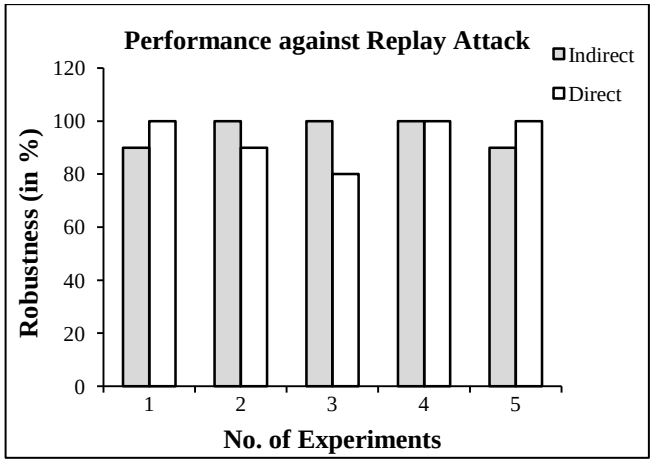

Fig.4(i)

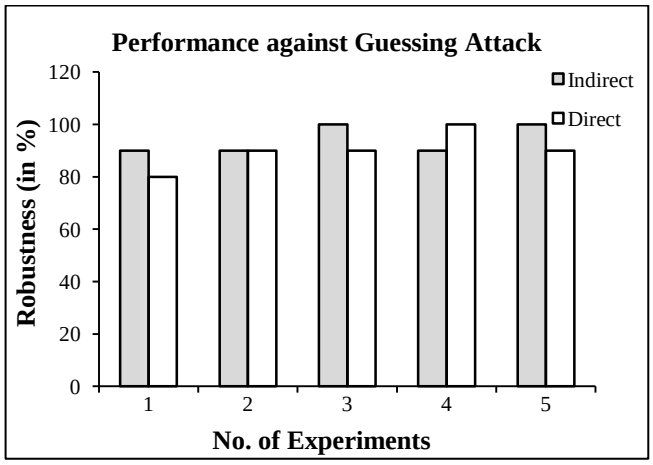

Fig.4(ii)

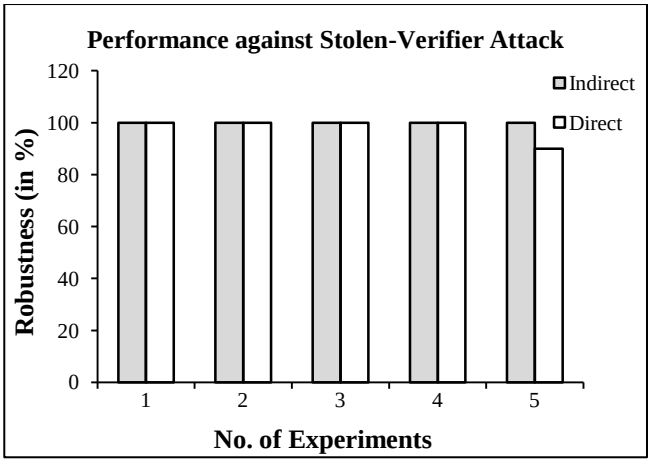

Fig.4(iii)

Fig.4. Effectiveness experiment for the proposed direct and indirect authentication protocol against (i) Replay attack, (ii) Guessing attack and (iii) Stolen-Verifier attack

Among the five experiments conducted for replay attack, indirect authentication protocol shows 100% robustness in three experiments and 90% robustness in two experiments whereas the direct authentication protocol shows 100% robustness in three experiments, 90% and 80% robustness in the other two experiments. Working against the guessing attack, indirect authentication protocol is 100% robust in two experiments and 90% robust in three experiments whereas direct authentication protocol is 100%, 90% and 80% robust in one, three and one experiment respectively. Indirect authentication protocol is 100% robust in all the experiments against Stolen-Verifier attack whereas the direct authentication protocol achieved 100% robust in four experiments and 90% in an experiment.

The overall performance can be visualized by taking the mean robustness for all experiments. The overall performance is illustrated in Fig.3.

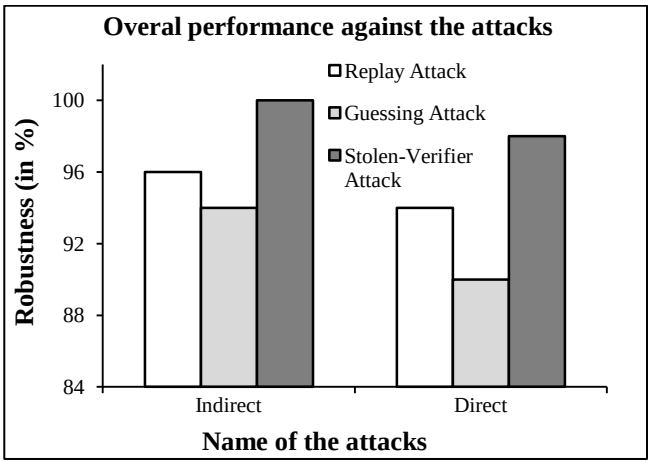

Fig.5. Mean robustness of the proposed protocols against the attacks

From Fig.5, it can be seen that the Indirect Authentication protocol is 96%, 94% and 100% robust against replay, guessing and Stolen-verifier attacks respectively whereas the direct authentication protocol is 94%, 90% and 98% robust against replay, guessing and Stolen-verifier attacks respectively. In an average, indirect authentication protocol is 2.6% more robust than direct authentication protocol.

5. CONCLUSION

In the previous two works, we have proposed two authentication protocols for mobile networks based on ECC. The protocols had taken the advantage of elliptic curve properties and hence it had been developed to provide the secure environment for mobile networks. A wide empirical analysis as well as the experimental analysis had been made over the proposed two protocols. One of the proposed two protocols is of direct type and the other one of indirect authentication type. To validate the efficiency of the protocols, we have utilized the performance measures such as computational overhead, communication overhead, storage overhead and computational complexity. To evaluate the effectiveness of the protocols, the protocol is subjected to assumed environment with replay attack, guessing attack and stolen-verifier attack. In analyzing different views, the indirect authentication protocol based on ECC seems to be effective however the direct authentication protocol is efficient. Hence, depends on the application and the environment, the protocol can be utilized to make a secure environment in mobile networks.

REFERENCES

[1] Terje Gjøsaeter, Kjetil Haslum and Trond Stølen Gustavsen, "Implementing Elliptic Curve Cryptosystems Using Hesse Curves over Prime Fields", *In Proceedings of the 8th Nordic Workshop on Secure IT Systems*, pp. 109-116, 2003.

[2] David Galindo, Sebastia Martin, Paz Morillo and Jorge L. Villar, "An Efficient Semantically Secure Elliptic Curve Cryptosystem Based On KMOV", *In Proceedings of WCC*, pp. 213-221, 2003.

[3] Lejla Batina, Siddika Berna Ors, Bart Preneel and Joos Vandewalle, "Hardware Architectures for Public Key Cryptography", *The VLSI Journal Integration*, Vol. 34, No. 1-2, pp. 1–64, May 2003.

[4] Ray C.C. Cheung, Wayne Luk and Peter Y.K. Cheung, "Reconfigurable Elliptic Curve Cryptosystems on a Chip", *In Proceedings of the Design, Automation and Test in Europe Conference and Exhibition*, Vol. 1, pp. 24-29, 2005.

[5] Ozturk, Sunar and Sava, "Low-Power Elliptic Curve Cryptography Using Scaled Modular Arithmetic", *In Proceedings of 6th International Workshop on Cryptographic Hardware in Embedded Systems*, Vol. 3156, pp. 92-106, 2004

[6] Nils Gura, Sheueling Chang Shantz, Hans Eberle, Sumit Gupta, Vipul Gupta, Daniel Finchelstein, Edouard Goupy and Douglas Stebila, "An End-to-End Systems Approach to Elliptic Curve Cryptography", *In Proceedings of CHES '02 Revised Papers from the 4th International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 349-365, 2002.

[7] Nghi Nguyen, Kris Gaj, David Caliga and Tarek El-Ghazawi, "Implementation of Elliptic Curve Cryptosystems on a Reconfigurable Computer", *In Proceedings of the IEEE International Conference on Field-Programmable Technology*, 2003.

[8] Elliptic curve from - http://mathworld.wolfram.com/EllipticCurve.html

[9] Vivek Kapoor, Vivek Sonny Abraham and Ramesh Singh, "Elliptic Curve Cryptography", *ACM Ubiquity*, Vol. 9, No. 7, pp. 1-8, 2008.

[10] G.V.S. Raju and Rehan Akbani, "Elliptic Curve Cryptosystem and its Applications", *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 1-4, 2003.

[11] O. Srinivasa Rao and S. Pallam Setty, "Efficient Mapping Methods for Elliptic Curve Cryptosystems", *International Journal of Engineering Science and Technology*, Vol. 2, No. 8, pp. 3651-3656, 2010.

[12] M. Aydos, B. Sunar and C.K. Koc, "An Elliptic Curve Cryptography Based Authentication and Key Agreement Protocol For Wireless Communication" , *In Proceedings of the Second International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications Symposium on Information Theory*, pp. 1-12, 1998.

[13] Wendy Chou, "Elliptic Curve Cryptography and Its Applications to Mobile Devices", *Technical Report, University of Maryland*, pp. 1-23, 2003.

[14] Istvan Zsolt Berta and Zoltan Adam Mann, "Implementing Elliptic Curve Cryptography On PC and Smart Card", *PeriodicaPolytechnica Series Electrical Engineering*, Vol. 46, No. 1–2, pp. 47-73, 2002.

[15] Kefa Rabah, "Elliptic Curve Cryptography over Binary Finite Field GF(2^m)", *Journal of Information Technology*, Vol. 5, No. 1, pp. 204-229, 2006.

[16] Kefa Rabah, "Implementation of Elliptic Curve Diffic-Hellman and EC Encryption Schemes", *Journal of Information Technology*, Vol. 4, No. 2, pp. 132-139, 2005.

[17] Miguel Morales-Sandoval and Claudia Feregrino-Uribe, "On the Hardware Design of an Elliptic Curve Cryptosystem", *In Proceedings of the Fifth Mexican International Conference on Computer Science*, pp. 64 - 70, 2004.

[18] Natarajan Vijayarangan, "A System and Design of Extensible Authentication Protocols Based on ECC and SKE Mechanisms for Mobile and Wireless Communications", *In Proceedings of the 9th WSEAS International Conference on Advances in E-Activities, Information Security and Privacy*, pp. 53-57, 2010.

[19] Ray C. C. Cheung, Nicolas Jean-baptiste Telle, Wayne Luk and Peter Y. K. Cheung, "Customizable Elliptic Curve Cryptosystems", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 13, No. 9, pp. 1048-1059, 2005.

[20] H.K. Pathak and Manju Sanghi, "Speeding up Computation of Scalar Multiplication in Elliptic Curve Cryptosystem", *International Journal on Computer Science and Engineering*, Vol. 02, No. 04, pp. 1024-1028, 2010.

[21] Rahila Bilal and M. Rajaram, "High Speed and Low Space Complexity FPGA Based ECC Processor", *International Journal of Computer Applications*, Vol. 8, No.3, pp. 5-10, 2010.

[22] Adnan Abdul-Aziz Gutub, "Remodeling of Elliptic Curve Cryptography Scalar Multiplication Architecture using Parallel Jacobian Coordinate System ", *International Journal of Computer Science and Security*, Vol. 4, No. 4, pp.373-435, 2010.

[23] Rahila Bilal and M. Rajaram, "Design and Implementation of High Performance ECC Coprocessor ", *International Journal of Engineering Science and Technology*, Vol. 2, No. 11, pp. 6759-6770, 2010.

[24] J. Portilla, A. Otero, E. de la Torre, T. Riesgo, O. Stecklina, S. Peter and P. Langendorfer, "Adaptable Security in Wireless Sensor Networks by Using Reconfigurable ECC Hardware Coprocessors", *International Journal of Distributed Sensor Networks*, Vol. 2010, pp. 1-12, 2010.

[25] K. Kumar, J. Nafeesa Begum and V. Sumathy, "A Novel Approach Towards Cost Effective Region-Based Group Key Agreement Protocol For Peer to Peer Information Sharing In Mobile Ad Hoc Networks", *International Journal of peer-to-peer networks*, Vol.1, No.1, pp. 10-28, 2010.

[26] R Rajaram Ramasamy, M. Amutha Prabakar, M. Indra Devi and M. Suguna, "Knapsack Based ECC Encryption and Decryption", *International Journal of Network Security*, Vol. 9, No. 3, pp. 218–226, 2009.

[27] S. Prasanna Ganesan, "An Efficient Protocol for Resource Constrained Platforms Using ECC", *International Journal on Computer Science and Engineering*, Vol. 2, No. 1, pp. 89-91, 2009.

[28] Wenbo Mao, "An identity-based non-interactive authentication framework for computational grids", *Hewlett-Packard Laboratories, Technical Report HPL*, pp. 1-12, 2004.

[29] Lei Yang, Jinsong Han, Yong Qi and Yunhao Liu, "Identification-Free Batch Authentication for RFID Tags", *In Proceedings of the IEEE International Conference on Network Protocols*, 2010.

[30] Pedro Peris-Lopez, Julio Cesar Hernandez-Castro, Juan M. Estevez-Tapiador, and Arturo Ribagorda, "EMAP: An Efficient Mutual-Authentication Protocol for Low-cost RFID Tags", *In Proceedings of the OTM Federated Conferences and Workshop*, 2006.

[31] S. Rathi and K. Thanushkodi, "Performance Analysis of Mobile IP Registration Protocols", *WSEAS Transactions on Computers*, Vol. 8, No. 3, pp. 538-548, 2009.

[32] Riham Abdellatif, Heba K. Aslan and Salwa H. Elramly, "New Real Time Multicast Authentication Protocol", *International Journal of Network Security*, Vol.12, No.1, pp.13-20, 2011.

[33] Kyusuk Han, Kwangjo Kim and Taeshik Shon, "Untraceable Mobile Node Authentication in WSN", *Journal of Open Access sensors*, Vol. 10, pp.4410-4429, 2010.

[34] Sencun Zhu, Sanjeev Setia and Sushil Jajodia, "LEAP+: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks", *ACM Transactions on Sensor Networks*, Vol. 2, No. 4, pp. 500–528, 2006.

[35] Eslam Gamal Ahmed, Eman Shaaban and Mohamed Hashem, "Lightweight Mutual Authentication Protocol for

Low Cost RFID Tags ", *International Journal of Network Security & Its Applications*, Vol. 2, No. 2, pp. 27-37, 2010.

[36] Adrian Perrig, Robert Szewczyk, Victor Wen, David Culler and J. D. Tygar, "SPINS: Security Protocols for Sensor Networks", *Journal Wireless Networks*, Vol. 8, No. 5, pp. 521-534, 2002.

[37] Rafael Martinez-Pelaez, Francisco Rico-Novella, Cristina Satizabal and Jhon J. Padilla, "Performance Analysis of Mobile Payment Protocols over the Bluetooth Wireless Network", *6th COLLECteR Iberoamerica*,2008.

[38] P.G.Rajeswari and Dr.K.Thilagavathi, "An Efficient Authentication Protocol Based on Elliptic Curve Cryptography for Mobile Networks", *International Journal of Computer Science and Network Security*, Vol. 9, No.2, pp. 176-185, 2009.

[39] P.G.Rajeswari and Dr.K.Thilagavathi, A Novel Protocol for Indirect Authentication in Mobile Networks based on Elliptic Curve Cryptography", *Journal of Theoretical and Applied Information Technology*, Vol.6, No.1, p.p. 56 – 66, 2009.