SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TNC TRANSACTIONS ON NETWORKS AND COMMUNICATIONS

VOLUME 4, ISSUE 3
ISSN: 2054 -7420

Pseudonym Creation and Parameter Based Key Generation for a Robust Authentication using *ECC*

[1]R Bhakthavathsalam, [2]Kavyashree U B, [3]Maithri R Kumar, [4]Nikhila Dwarakanath, [5]Saripalli Pooja and [6]Gowranga K H

[1, 6] Supercomputer Education and Research Centre, Indian Institute of Science, Bangalore, India;
[2, 3, 4, 5] Dayananda Sagar College of Engineering, Bangalore, India;
bhaktha@serc.iisc.ernet.in; kavyashree.ub9294@gmail.com; maithri.rk@gmail.com; nikhiladknath@gmail.com; poojasaripalli@gmail.com; gowranga@cds.iisc.ac.in

ABSTRACT

Security is an important aspect in every field today and it cannot be neglected whatsoever. Network security has become one of the most important concerns of this generation due to its wide range of applications. With the emerging techniques in elliptic curve cryptography (ECC), it has now become the mainframe of many cryptosystems. Due to the generation of small key sizes in ECC, it poses as a favourable cryptosystem that can be used to minimize memory consumption. The previous schemes that were developed were unable to achieve user anonymity and were also vulnerable to stolen-verifier attacks, offline password guessing and insider attacks. The proposed scheme aims at improving the security provided to a user during authentication phase involved in any transaction. A step-by-step process is outlined in order to enhance security. It intends to provide a system involving Elliptic curve cryptography along with One Time Passwords to implement a new authentication scheme that overcomes certain threats to network security. The main features of this scheme includes generation of a common key with scalar multiplication, provision of user anonymity in order to prevent man-in-the-middle attacks using the Elliptic curve base point and a random number in order to prevent session hijacking attack or replay attack. This authentication paradigm finds its applications in ATM systems, RFID tags, smart card applications, mobile applications with SIM cards and so on. The proposed scheme works favourably with devices that have less resources (like memory).

Keywords: Elliptic curve cryptography (*ECC*); User anonymity; One Time Password (*OTP*); Network security; Scalar multiplication; Authentication scheme; Decryption; Encryption; Key agreement; Key generation.

1 Introduction

Information security [1] is the application of measures to ensure the safety and privacy of data by managing its storage and distribution. Information security has both technical and social implications. The first simply deals with the 'how' and 'how much' question of applying secure measures at a reasonable cost. The second grapples with issues of individual freedom, public concerns, legal standards and how the need for privacy intersects them. This discussion covers a range of options open to business managers, system planners and programmers that will contribute to your ultimate security strategy. The eventual

DOI: 10.14738/tnc.43.2070
Publication Date: 10th July 2016
URL: http://dx.doi.org/10.14738/tnc.43.2070

choice rests with the system designer and issues. Modern security products are now designed to balance the needs of business on the Internet while protecting against today's sophisticated threats. Modern information security practices have evolved into a blended approach to managing access to information.

In implementing a security system, all data networks deal with the following main elements:

- Hardware which includes servers, redundant mass storage devices, communication channels and lines, hardware tokens (smart cards) and remotely located devices (e.g., thin clients or Internet appliances) serves as interfaces between users and computers
- Software, includes operating systems, database management systems, communication and security application programs
- Data, including databases containing customer - related information.
- Personnel, are originators or/and users of data and they may be professional personnel, clerical staff, administrative personnel, and computer staff.

Cryptography [1] is the method of converting data from a human readable form to a modified form, and then back to its original readable form, to make unauthorized access difficult. Cryptography is used in the following ways:

- provides data security by ensuring adequate encryption of the data.
- It provides a means to check if data has been manipulated or not, thus ensuring data integrity.
- provides data uniqueness by confirming that data is "original", and not a "duplicate". The "original" data is sent with a unique identifier. This unique identifier is then checked by the receiver of the data.

1.1 Elliptic Curve Cryptography

Elliptic curve cryptography (ECC) [2] is a public key encryption technique based on the concept of *elliptic curves* that has gained popularity recently in creating cryptographic keys. Elliptic curve cryptography uses the properties of a basic elliptic curve equation for the key generation over the generation of the product of very large prime numbers. It can be used in conjunction with most public key encryption methods, such as RSA, and Diffie-Hellman. According to some researchers, ECC can yield a level of security with a 160-bit key that other systems require a 1,024-bit key to achieve. Fig 2 shows a comparison between the key sizes generated by RSA and ECC. The equation of an elliptic curve is: $y^2=x^3+ax+b$ *(1)*, where a and b must satisfy the condition $4a^3+27b^2\neq 0$ *(2)*. There are two basic group operations known as Prime Field Arithmetic operation and Scalar Multiplication performed on elliptic curves as explained below:

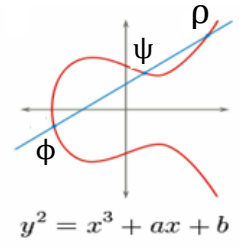

Figure 1: Elliptic curve [6]

ECC(in bits)	RSA (in bits)
106	512
112	768
132	1024
160	2048
283	7680

Figure 2: Comparison of key sizes in RSA and ECC [7]

1.2 Prime Field Arithmetic

The prime field GF(p) [3] is the finite field whose elements are all the integers between 0 and p-1 inclusive. For every a € GF(p) the following operations are defined as:

- Addition i.e., If a, b € GF(p) then a + b = r, where r € GF(p) and equals (a + b) mod p.
- Multiplication i.e., If a, b € GF(p) then a * b = r, where r € GF(p) and equals (a * b) mod p.
- Inversion i.e., If a is a non-zero element in GF(p), then a'1 = c € GF(p) such that a*c = 1 mod p.

Points whose coordinates (x, y) satisfy equation along with the point of infinity P∞ (which is at the top of the y-axis), make up the set of rational points on curve E W. Trappe and L. Washington (2002). The negative of a point P is its reflection in the x-axis, i.e., -P = (x, -y). For elliptic curves over prime fields GF(p) with p > 3, the parameters a and b of eq. 2 should satisfy the condition 4a3+27b2≠ 0 (mod p). D Hankerson et al. (2004). The condition is required to ensure that the curve is smooth, and there are no points at which the curve has two or more distinct tangent lines. The allowable point coordinates become integer modulo p. The standard representation of points on an elliptic curve E over prime fields is the Affine coordinates. The graphical representation of point addition is computed using Eq.3, Eq. 4 and Eq. 5 and the graphical representation of point doubling is computed using Eq.6, Eq.7 and Eq.8.

$$\lambda = (Y_2 - Y_1) * (X_2 - X_1) - 1 \quad (3)$$

$$X_3 = \lambda_2 - X_1 - X_2 \quad (4)$$

$$Y_3 = \lambda(X_1 - X_3) - Y_1 \quad (5)$$

$$\lambda = (3X_1^2 + a) * (2Y_1) - 1 \quad (6)$$

$$X_3 = \lambda_2 - 2X_1 \quad (7)$$

$$Y_3 = \lambda(X_1 - X_3) - Y_1 \quad (8)$$

Point multiplication, or scalar multiplication, is higher in the hierarchy of ECC operations than the point addition and doubling. It is computed by performing a series of point additions and point doublings. A point ‘P’ is multiplied by a scalar k to yield the point ‘kP’. For example, to compute ‘5P’ one would double ‘P’ twice to obtain ‘4P’ then add ‘P’ to the intermediate result to obtain ‘5P’. Various algorithms exist for point multiplication. When the point P is unknown the unknown point multiplication is used, whereas when the point is fixed, the known point multiplication is used.

The known point multiplication is faster than the unknown point multiplication because certain fixed parameters are known. When two points are added or doubled in Affine Coordinates I. Blake et al. (1999), we need to perform modular inversion to compute p, e.g., (X2-X1)-1 or (2Y1)-1, in eq.3 and eq.6. Modular inversion is a computationally intensive operation, and it is required each time a point is added or doubled. There are alternative point representations that favour modular multiplications over inversions. Some modular operations are typically required in a larger framework such as the signature schemes.

The performance of field multiplication is fundamental to mechanisms based on elliptic curves. Constraints on hardware integer multipliers and the cost of carry propagation can result in significant bottlenecks in direct implementations of algorithms.

2 Related Works

2.1 Islam Biswas Scheme:

Islam and Biswas [4] proposed an advanced password authentication scheme based on ECC. The authors claimed that their scheme provides mutual authentication and is free from all known cryptographic attacks, such as replay attack, offline password guessing attack, insider attack and so on. Although their scheme is superior to the previous solutions for implementation on mobile devices, we find their scheme cannot achieve the claimed security: their scheme is vulnerable to the offline password guessing attack, the stolen verifier attack.

Islam-Biswas's scheme consists of four phases: the registration phase, the authentication phase, the session key distribution phase and the password change phase. In Islam-Biswas's scheme, the user's identity ID is transmitted in plain, which may leak the identity of the logging user once the login messages were eavesdropped. In a word, neither initiator anonymity nor initiator un-traceability can be preserved in their scheme.

2.2 Chun Ta Li's Scheme:

The Chun Ta Li's scheme [5] for user authentication was an enhancement from the other schemes that were already provided. It is demonstrated that the so-called secure, anonymous user authentication scheme introduced by He et al. is vulnerable to eavesdropping attack and is not practical for real-life implementation. Li showed that user anonymity of their scheme is not achieved, the user has to bear in mind a long identity (128 bit) during the login phase, and there is no provision for fairness in the key agreement.

To remedy these security weaknesses, Li further proposed a novel authentication scheme which is immune to various known types of attack and is more secure and practical for mobile wireless networking. However, it is found that Li's authentication scheme has a serious security problem in that all registered users' sensitive passwords can be easily derived by the privileged-insider of remote server. The attack exists because the data written on smart card is enough to compute the password.

3 System Model

A user initially registers himself/herself with the server by providing his PIN (Personal Identification Number) and password via the client system. A secure communication channel is established between the client and the server before the information is sent in an encrypted format. The server stores the password verifier and pseudonym of the user's PIN in the verifier table after performing decryption. Upon further accesses, the authentication phase takes place and only if the right password and OTP are entered, the user is granted permission to login. The user is given a maximum of three trials. The OTP is sent to the user's registered email address. The established channel is used in every phase to communicate in a secure manner.

The user, client and server pose as active participants in every phase. A change of password upon request is provided to the user. An additional option of account deletion is provided in case the user wishes to deactivate and delete his/her account. On choosing account deletion, both the client and server systems authenticate the user details and perform the deletion operation and simultaneously update the same in their respective systems. Else, a status bit of 0 is sent indicating that the user does not wish to delete his account.

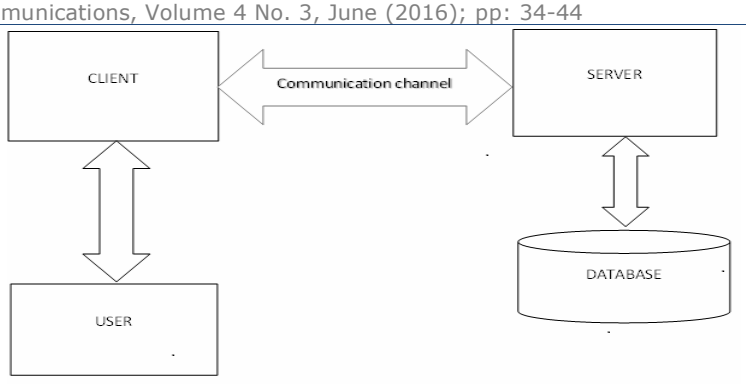

Figure 3: Block Diagram for System Design Model

4 A Secure Authentication Algorithm

The algorithms [5] aim at providing layers of security to the system to prevent security attacks like replay attack, session hijacking and man-in-the-middle attack. Security provided by this scheme is robust and efficient. The proposed scheme consists of four phases namely registration phase, authentication phase, password change phase and user eviction phase. Now each of these phases is described. The following are the terminologies that are going to be used throughout the paper:

- ID : Client identity
- V : Password verifier
- R : Random Number generated
- V_s : Key of the server
- G : Generator of elliptic curve
- H(.) : Hashing function
- $E_k(.)/D_k(.)$: Encryption/decryption function.
- W_c : $r_c \cdot r_c \cdot pw \cdot G$
- W_s : $r_s \cdot V_s = r_s \cdot d_s \cdot G$
- IND : It is a pseudonym for ID, IND=ID. $r_c.G$
- $K(K_x,K_y)$: Key for encryption $K=d_s.V=pw.r_c.d_s.G$
- SK : $(r_c \cdot r_c \cdot pw) \cdot W_s = r_c \cdot r_c \cdot pw \cdot r_s \cdot d_s \cdot G$

4.1 Registration Phase:

When a client [5] wants to become a new legal client, the client must register to the server with his/her PIN ID, password pw and Email ID. Then server stores each legal client's identity IND (pseudonym of user's identity, as computed on client side), password-verifier V, a status–bit, the Email ID and the flag bit, where the status–bit indicates the login status of the client to prevent many logged-in users' attack. If the client is logged in to the server, the status–bit is set to 1; otherwise it is set to 0. The flag bit is set to 0. Upon each unsuccessful attempt, the flag is incremented; if flag equals 3, access is denied.

4.2 Authentication Phase:

When a client [5] wants to access the Server, the client has to enter the PIN ID and password. This phase mainly involves validating the user and also provides an option for sending an OTP for a more secure authentication. The client then computes IND. The following steps are then performed:

Step 1:User→Server: IND, E_{Kx} (IND, R, W_c, V')

The client retrieves r_c, generates a new random number r_c', computes

$R = r_c.V_s = r_c.d_s.G$, $W_c = r_c.r_c.pw.G$, $V' = pw.r_c'.G$ and $E_{Kx}(IND, R, W_c, V')$ and sends ID and $E_{Kx}(IND, R, W_c, V')$ to Server, where V_s is the public key of Server and encryption key K_x is the x coordinate of $K = pw.r_c$. $V_s = pw.r_c.d_s.G = (K_x, K_y)$.

Step 2: Server→User: ($W_c + W_s$), H(W_s, V',SK)

Server computes the decryption key K_x by computing $K = d_s \cdot V = pw \cdot r_a \cdot d_s \cdot G = (K_x, K_y)$ and decrypts $E_{Kx}(IND, R, W_c, V')$ to reveal (IND, R, W_c, V'). Then Server compares decrypted ID with received IND . If all the conditions are satisfied, Server sends ($W_c + W_s$) and H(W_s, V')to client, where $W_s = r_s \cdot V_s = r_s \cdot d_s \cdot G$ and r_S is a random number which is generated by Server. Server generates the session key

$SK = (r_s.d_s).W_c = r_c.d_s.pw.r_c.r_c.G$.

Step 3: User→Server: IND ', H (W_c, W_s, V',SK)

Client retrieves W_s by subtracting W from ($W_c + W_s$) and checks whether the hashed result of (W_s, V',SK) is equal to the received H (W_s, V',SK). If so, the client computes H (W_c, W_s, V',SK) and sends it to Server.Here, the client computes the final session key $SK = (r_c.r_c.pw).W_s = r_c.r_c.pw.r_s.d_s.G$. The client also generates the new pseudonym IND'= $ID.r_c'.G$ and sends it to the Server.

Step 4: Server→User: Access Granted/Denied

Server uses its own W_s and (W_c, V') which is received from client in Step 1 to compute H (W_c, W_s, V',SK) and checks whether the hashed result of (W_c, W_s, V',SK) is equal to the received H (W_c, W_s, V',SK). If it holds, then Server generates an *OTP* and sends it to the Email ID the user registered with. Only is the user enters the right *OTP* within the given amount of time ,Server grants login request and replaces old password-verifier $V = pw \cdot r_c \cdot G$ with new password-verifier $V' = pw \cdot r_c' \cdot G$, and the old pseudonym $IND=ID.r_c.G$ by the new pseudonym $IND'=ID.r_c'.G$,otherwise Server denies login request. Finally, the user replaces r_c with r_c' if all of the conditions are satisfied. After finishing the authentication phase, the verifier table is updated.

4.3 Password change Phase:

When the client [5] wants to change his/her original password pw to a new password pw', client must notify the server to update the old password-verifier $V = pw \cdot r_c \cdot G$ to a new password-verifier $V' = pw' \cdot r_c' \cdot G$. If the authentication token H (W_c, W'_s) is valid,Server subtracts W_c from $W_c + V'$ to reveal new password-verifier V ' and replaces V with V ' if the computed value of the hash function of (W_s, V') is the same as the received value of the hashed result of (W_s', V'). The steps involved in the password change phase are as follows:

- *Step 1: User→ Server: IND, E_{Kx} (IND, R, W_c, V')*
- *Step 2: Server→User: ($W_c + W_s$), H (W_s, V')*
- *Step 3: User→Server: IND', H (W_c, W_s, V',SK), H($W_s + W_c$ + V')*
- *Step 4: Server→User: Password Change Granted/Denied*

4.4 User Account Deletion Phase:

After the user [5] has been successfully authenticated, the user can choose to delete the registered account. Upon deletion of account, the user's details are deleted on the client side. The client then communicates with the server to do the same. The server deletes the user's details from the verifier table. The same user cannot further obtain access after account deletion.

5 Implementation Analysis

The following are the steps involved during the execution of the algorithm:

- The user enters his/her option for registration, authentication or password change.
- During registration, the end-result is a successful registration of the user, with the details of the user being stored on the server verifier table.
- Upon authentication, the end-result is either access granted or denied during a transaction.
- If the user wishes to update the old password to a new password, a password change option is available.
- Finally, an option of deletion of the user's account is available, which deletes the user's information stored from the server and client records.

6 Testing and Results

Firstly, the user has to select the phase that he wishes to execute.

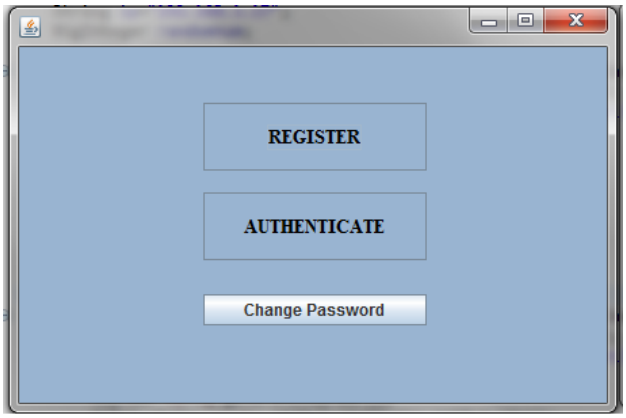

Figure 4: User's choice

Case 1: Registration phase

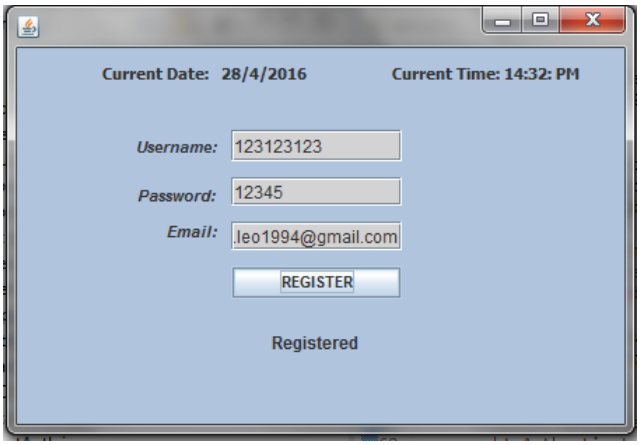

Figure 5: User Registration

At the client side, the user registers himself with his user PIN, password and email. The values received at server side are the base point (G) on EC curve and public key (Us) of server.

```
Gx value:484395612939064517590525852527979142027629495260417479958440807170824046352 86
Gy value:361342509567497957985851279195878819566111066729850150718771982535684144051 09
Usx value:365961645058617195368911348759382860352692343970484076656479883846918415605 46
Usy value:932051044005639701052497039985074059127340266438602194038542509942003553964 40
Registered
```

Figure 6: User Registration process

Figure 6 represents successful registration. Upon successful registration, public key of the server(Us) and generator point(G) are sent to the client.

Server Side:

```
in servereg
this is first
Recieved values:
Id: 65610f06430c44ffd47f9758dcc039aaf237a48a761e087fd19cb89b21189bdc
Uax: 1143892465395112978894345551477716801784832459418490166919214192128381683550 50
Uay: 9126371763060230112600300692336311512772837620982687534992463418212355009704 8
email id: motz.leo1994@gmail.com
sent Gx
sent Gy
sent Usx
sent Usy
Successfully Registered
```

Figure 7: Server side successful registration

Case 2: Authentication phase

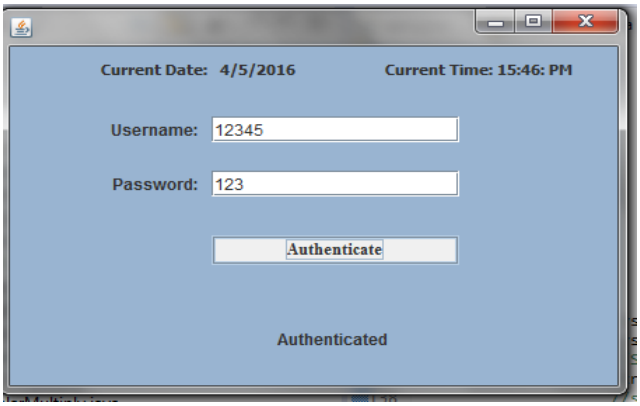

Figure 8: User authentication

During this phase, user enters the registered user PIN and password. Results of necessary computations are exchanged between client and server. For verification purposes an OTP is sent to the email id of the user. The user is required to enter this at the client side which is sent to the server.

Figure 9: OTP entry

On successful authentication, the user can choose to delete his account.

```
User account authenticated.Proceeding...
DELETE ACCOUNT?
1
Account deleted
Authentication phase ended
```

Figure 10: Successful authentication

Server side:

```
Access Granted sending otp
OTP   :05042660
message sent successfully....
database updated in changeflag:  1
Access granted
Ua before db update:  1127955191510212139177543778551007140132028853924260988584933697859679300307951115869484079038832488
status of changedb:1
Ua after change in db:  9819779657862166011228059769126034323213863814237658041253553110924533536855711112250714421052906 7
INDa'     :b2df22f56a709a011f1db8f19f63a837ef4864616ae97dbb187d01f1d54ea40d
delete account (1) yes (0) no: 0
```

Figure 11: Result for successful authentication

```
Access Granted sending otp
OTP   :53191471
message sent successfully....
database updated in changeflag:  1
Access granted
Ua before db update:  5106511898196698625686762143616852853288040894148462896532910198372097186804624000287643729648234004
status of changedb:1
Ua after change in db:  6478655362570301522775629407147174772880958658346289125978740351309636758042426330290143260201387 3
INDa'     :0e2a184e0ed998c60bee3a5814f6e256f9a1cefbc6e3f146aeba58b19b49cfe7
delete account (1) yes (0) no: 1
database updated:  1
```

Figure 12: Result for account deletion

The user has a choice of deleting his account only after authentication, when he no more requires it. Fig 11 and Fig 12 show the user's choice to delete account. Fig 12 depicts deleting account, whereas Fig 11 depicts only authentication. In authentication phase, 1st the check for correct password is done, only then an OTP is mailed to the user. Upon the right entry, he is given an option of deleting account. If he chooses 1, then account is deleted.

Case 3: Password change phase

During this phase, the user who wishes to change his password, has to enter his old password and also enter the new password and notifies the server to perform the same. After several steps of verification, as described in the proposed scheme, the password is changed. This step, too, includes the user entering OTP. The server then authenticates the user, only if the user has entered the correct password, he is allowed to change it. The server then saves the new password verifier in place of the old verifier against the username specified. Upon successful change, an appropriate status is reflected.

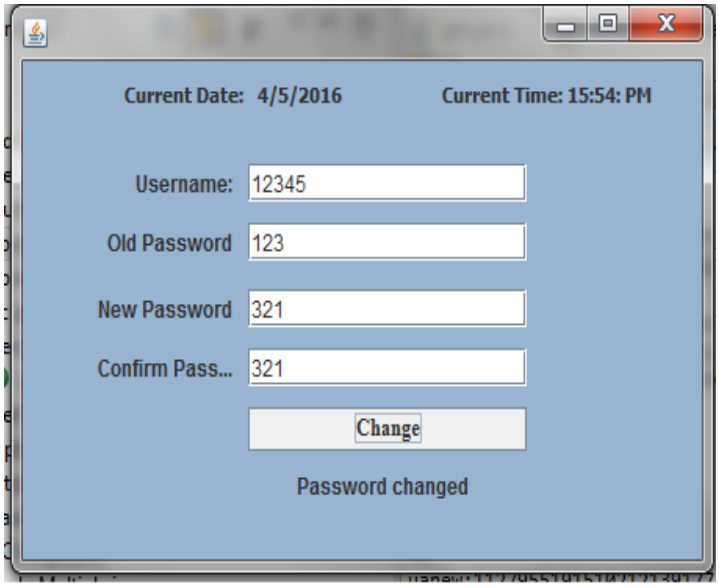

Figure 13: User password change

Server side:

```
Access Granted sending otp
OTP    :60192597
message sent successfully....
database updated in changeflag:  1
AcCess granted
Ua before db update:  98197796578621660112280597691260343232138638142376580412535531109245335368557111122507144210529067 17
status of changedb:1
Ua_ after change in db:  5106511898196698625686762143616852853288040894148462896532910198372097186804624000287643729648234
Idx     :e52cc69f392cbc9136aaccce931b66e3bb593b9ba7d474f4d6d0faad1022c2d2
Password changed sucessfully
```

Figure 14: Result for password change

When the user wants to change his password, he is first authenticated as mentioned previously and then granted the permission for change his password.

7 Conclusion

An advanced password authentication and updated scheme is implemented that empowers login clients and remote servers with a secure and privacy-preserving authentication. Even though the password-verifier is compromised by an adversary, the system does not worry that the stolen password-verifier will be used by others. Moreover, the proposed scheme allows a registered client to login to the server anonymously, when a pseudonym is used during the authentication procedure. Therefore, the proposed ECC-based scheme achieves mutual authentication, lower computation cost [8] and privacy protection. This scheme is an efficient and robust alternative for other popular schemes that use RSA cryptographic algorithm. It attempts to overcome a variety of drawbacks present in previous schemes.

Since the mechanism of parameter-based key generation is different on both client and server, and uses different number of parameters for the creation of keys, this scheme presents a unique way for the key generation protocol. Another major observation here is that the PIN that is provided by the user changes during every session, which makes it more secure to attacks by outsiders. The password provided by the user is not sent directly; instead, it is modified using elliptic curve parameters to form a password verifier

which changes every session. Additional features like account deletion and usage of *OTP* makes this scheme more user-friendly and efficient.

This scheme can be effectively implemented in the login portals, online payment application services (like Bitcoin) that are prevalent in a variety of applications in today's technology, which can be altered to suit each application's requirements.

To further enhance security, biometric features like fingerprint verification, retinal detection and verification can be added which makes it more robust, secure and efficient. QR code can be used at the time of authentication. This scheme can be extended on to smart cards, RFID tags, SIM cards and other mobile applications.

REFERENCES

[1] *The concept of Security*: www.smartcardbasics.com/smart-card- security.html

[2] *A brief introduction to Elliptic Curve Cryptography and the algorithms*: http://searchsecurity.techtarget.com/definition/elliptical-curve- cryptography

[3] *Prime Field Arithmetic:* http://andrea.corbellini.name/2015/05/23/elliptic-curve-cryptography-finite-fields-and-discrete-logarithms/

[4] *Islam Biswas Scheme for User Authentication:* https://eprint.iacr.org/2012/190.pdf

[5] Chun-Ta Li: *A new password authentication and user anonymity scheme based on elliptic curve cryptography and smart card*, IET Inf. Secur, 2013, Vol. 7, Iss 1.

[6] William Stallings: *Cryptography and Network Security,* 3rd edition, Published November 16th 2005 by Prentice Hall Hardcover, 680 pages.

[7] Bos, Joppe W, et al. *Elliptic curve cryptography in practice*; Financial Cryptography and Data Security, Springer Berlin Heidelberg, 2014, 157-175.

[8] Sukalyan Goswami et al. *Enhancement of GSM Security Using Elliptic Curve Cryptography Algorithm;* 2012 Third International Conference on Intelligent Systems Modelling and Simulation, 639-644.