A NOVEL PASSWORDLESS AUTHENTICATION SCHEME FOR SMART PHONES USING ELLIPTIC CURVE CRYPTOGRAPHY

Dr. Sheetal Kalra

Department of Computer Science & Engineering
Guru Nanak Dev University, Regional Campus
Jalandhar-144008, India

ABSTRACT

Today, a large number of people access internet through their smart phones to login to their bank accounts, social networking accounts and various other blogs. In such a scenario, user authentication has emerged as a major security issue in mobile internet. To date, password based authentication schemes have been extensively used to provide authentication and security. The password based authentication has always been cumbersome for the users because human memory is transient and remembering a large number of long and complicated passwords is impossible. Also, it is vulnerable to various kinds of attacks like brute force, rainbow table, dictionary, sniffing, shoulder surfing and so on. As the main contribution of this paper, a new passwordless authentication scheme for smart phones is presented which not only resolves all the weaknesses of password based schemes but also provide robust security. The proposed scheme relieves users from memorizing and storing long and complicated passwords. The proposed scheme uses ECDSA which is based on Elliptic Curve Cryptography (ECC). ECC has remarkable strength and efficiency advantages in terms of bandwidth, key sizes and computational overheads over other public key cryptosystems. It is therefore suitable for resource constraint devices like smart phone. Furthermore, the proposed scheme incorporate CAPTCHA which play a very important role in protecting the web resources from spamming and other malicious activities. To the best of our knowledge, until now no passwordless user authentication protocol based on ECC has been proposed for smart phones. Finally, the security and functionality analysis shows that compared with existing password based authentication schemes, the proposed scheme is more secure and efficient.

KEYWORDS

User Authentication, CAPTCHA, Smart Phone, Elliptic curve cryptography, Elliptic Curve Digital Signature Algorithm

1. INTRODUCTION

We are living in the era of mobile computing where the recent years have witnessed an explosive growth of mobile devices like smart phones, laptops, tablets and so on. The introduction of new capabilities and functionalities for mobile devices has opened new avenues in this era. Of these, smart phones have become an essential part of human life today. According to eMarketer (a market research company), the number of smart phone users worldwide will surpass 2 billion in 2016 [29]. With time, smart phones have tremendously improved in terms of their processing

power, memory and other resources, making them capable to efficiently handle large processing computations anytime, anywhere. Today, a large number of people access internet through their smart phones to login to their bank accounts, social networking accounts and various other blogs. In such a scenario, user authentication has emerged as a major security issue in mobile internet. The goal of authentication is to ascertain the user identity by proving to the system that the user is who they claim to be. To date, the popular trend among service providers is to register the user who wants to avail their services. After successful registration, the user is issued a username/userID which is unique for each user. Most of the service providers give freedom to users to choose their own password. In such a scenario, to avail the service, the user is required to login to his/her account and enter username/userID and password. Login using userID and password [1][2] is still one of the most popular and convenient method of user authentication over insecure networks. Various other authentication alternatives like biometrics, smartcards and tokens are also available but they have their own disadvantages. Smartcard based schemes [3-5] need PINs and passwords, and are also vulnerable to smartcard lost problem. Biometric based scheme [6][7] require extra hardware for implementation and also raises privacy concerns. Passwords also have various drawbacks. The foremost problem is to memorize long and random passwords. Forgetting the passwords is very common among the users which arise due to fundamental limitation of human long-term memory. As a result, users tend to choose weak passwords like name of pets, date of birth and dictionary words [8][9]. This exposes the system to various types of security attacks. The most prominent of them are as follows:

1. Brute force attack: It is also known as exhaustive search. In this, the attacker generates every possible combination of user password and tries to authenticate itself as the actual user. If given enough time and provided the user does not change his/her password, the attacker can successfully launch this attack.
2. Dictionary attack: It is based on the fact that users tend to select memorable words for the password such as names of towns, pets, date of birth etc, thus reducing the number of possible combinations to only meaningful words. The attacker can compile the list of such meaningful words into a dictionary and then launch a search against the system by trying the same user account and words from dictionary file as the password.
3. Rainbow table attack: In this, the attacker makes use of rainbow table which contains precomputed values. Thus the attacker does not have to calculate all the combinations of possible passwords, which saves time and system resources.
4. Sniffing attack: In this, the attackers try to obtain the user credentials like userID and password over insecure networks. They can later use these credentials and impersonate as legal clients.
5. Imposter servers: In this, the attacker impersonates as real server and can persuade the user in sharing its confidential data including its userID and password
6. Man in middle attack: In this, the attacker sits somewhere between the user and the server on the network and intercepts the messages exchanged between the user and server. The attacker may even alter the messages and then send it to them.
7. Shoulder surfing: In this, the attacker can look over the user's shoulder while he is typing his password to authenticate to the server.
8. Offline attack: In this, the attacker tries to obtain offline access to the database where the user credentials (password file) are stored. This is possible if the attacker has administrative permissions to access the database or has physical access to the authentication server.

9. Keyboard loggers, Trojans and viruses: In order to access the user password, the attacker may install a key-logging resident program that can collect all the key strokes that the user has made. A Trojan and virus can also be used by attacker to obtain confidential user information.

Apart from these attacks, remembering long and complicated passwords strain human memory. Thus human beings tend to choose short and easily memorable passwords which they re-use [11] on every website they come across. Such weak passwords can be easily compromised by adversaries leading to serious security breach. Various strong password generators (like 1Password) can be used to generate secure, long and complicated passwords; yet, memorizing such complicated passwords become a major issue. Even storing these passwords in the system is not safe as they can be easily intercepted by an insider or compromised through impersonation attack.

What if the user can login to any website without the need of password? Such passwordless systems will resolve all the above mentioned problems which are due to passwords. Passwordless systems are user-friendly because they relieve users from remembering and storing complex and lengthy passwords. However, while designing such systems, an utmost care should be taken to ensure the security and privacy of the system and user.

Human Interaction Proof (HIP) [12][13] system can successfully ensure the security and privacy of online web resources by distinguishing between human users and computers. One form of HIP is CAPTCHA (Completely Automated Public Turing test to tell Computers and Human Apart). CAPTCHAs [14] make use of hard AI problem to defend the system against malicious internet bot programs. Various commercial websites like Microsoft, Google and Yahoo have employed CAPTCHAs to prevent spams and other automated malicious activities. CAPTCHAs can either be OCR (Optical Character Recognition) based or non-OCR based. OCR based CAPTCHAs include textual CAPTCHAs while non- OCR based include audio, logical, animated and video CAPTCHAs. Logical CAPTCHAs include questions [15], puzzles [16] which are easily solvable by human beings but are a big challenge for computers. To date, a number of improved CAPTCHA based schemes [17-22] have been proposed which can be conveniently implemented on systems with constrained resources.

In this paper, we introduce a new and secure passwordless authentication scheme for smart phones. The password based authentication has always been cumbersome for the users because human memory is transient and remembering a large number of long and complicated passwords is impossible. To overcome this shortcoming, the proposed scheme uses passwordless authentication which is based on CAPTCHA and ECDSA.

The remainder of this paper is organized as follows: Section 2 gives the overview of mathematical background of Elliptic curve, its related mathematical problems and an outline of Elliptic Curve Digital Signature Algorithm. Section 3 reviews the scheme in [10] and its security weaknesses are analyzed in Section 4. Section 5 presents our advance smart phone based passwordless authentication scheme on ECC. Section 6 analyses the security of the proposed scheme and presents the functionality features of our scheme in Section 7. Section 8 concludes the paper.

2. MATHEMATICAL BACKGROUND

The robustness of any cryptographic security protocol depends on the hardness in solving the underlying mathematical problem. The security of ECC based protocols depend on the difficulty of solving Elliptic Curve Discrete Logarithm Problem (ECDLP), Elliptic Curve Computational Diffie–Hellman Problem (ECCDHP) and Elliptic Curve Decisional Diffie–Hellman Problem (ECDDHP).

2.1. THEORY OF ELLIPTIC CURVE

There are two families of elliptic curves [23] which are used in cryptographic applications: prime curve over Z_p and binary curves over $GF(2^m)$. For Elliptic Curve Digital Signature Algorithm (ECDSA), prime curves are used. The equation of a non-singular elliptic curve $E_p(a,b)$ over a finite field Z_p can be written as:

$$y^2 \bmod p \equiv x^3+ax+b(\bmod p)$$

where a and b are two integer elements and p is a large prime number. Furthermore, for the above equation to be non-singular, the condition $4a^3+27b^2 \neq 0$ must be satisfied. G is a base point of the elliptic curve with a prime order n and O is the point of elliptic curve at infinity, where n multiplies G is equal to O ($n.G=O$). An additive cyclic group $E = \{(x, y) \in E_p(a, b)\} \cup \{O\}$ is formed by any point $P(x, y) \in E_p(a, b)$, $x, y \in Z_p$, where O serves as additive identity element of the group. The addition rules for set of points over $E_p(a, b)$ are as follows [24]:

1. Identity: $P + O = O + P = P$, for all $P \in E_p(a, b)$
2. Negative: If $P = (x_1, y_1)$ where $P \in E_p(a, b)$ and the negative of P is denoted as $-P = (x_1, -y_1)$, then $P + (-P) = O$
3. Point addition: If $P = (x_1, y_1)$ and $Q = (x_2, y_2)$ where $P,Q \in E_p(a, b)$ and $P \neq -Q$, then $R = P + Q = (x_3, y_3)$ where

$$x_3 = (\lambda^2 - x_1 - x_2) \bmod p$$
$$y_3 = (\lambda (x_1 - x_3) - y_3) \bmod p$$
$$\lambda = \left(\frac{y2-y1}{x2-x1}\right) \bmod p$$

4. Point multiplication: If $P = (x_1, y_1)$ and $P \neq -P$ where $P \in E_p(a, b)$ then $2P = P + P = (x_3, y_3)$ where

$$x_3 = (\lambda^2 - 2x_1) \bmod p$$
$$y_3 = (\lambda (x_1 - x_3) - y_1) \bmod p$$
$$\lambda = \left(\frac{3x_1^2+a}{2y_1}\right) \bmod p$$

The point multiplication is computed by repeated addition as:

$$k.P = P+P+\ldots\ldots+P \text{ (k times)}$$

2.2. MATHEMATICAL PROBLEMS

The security of ECC [25] depends on hardness in solving the following problems:

1. Elliptic Curve Discrete Logarithm Problem (ECDLP): Given the equation $P = kG$ where $P, G \in E_p(a,b)$ and $k < p$, it is relatively easy to compute P when the values of k and G are known, but it is hard to evaluate k given the values of P and G.
2. Elliptic Curve Computational Diffie–Hellman Problem (ECCDHP): Given G and two point xG, yG, computation of xyG is hard, where $x, y \in Z_p^*$ and are randomly chosen and are smaller than n. Like ECDLP, the solution to ECCDHP is also computationally hard.
3. Elliptic Curve Decisional Diffie–Hellman Problem (ECDDHP): Given G and three point xG, yG, zG, it is hard to decide whether $zG = xyG$ or not, where $x, y, z \in Z_p^*$ and are chosen randomly and are smaller than n. Like ECCDHP, the solution to ECDDHP is also computationally hard.

2.3. ELLIPTIC CURVE DIGITAL SIGNATURE ALGORITHM

The Elliptic Curve Digital Signature Algorithm (ECDSA) is a variant of Digital Signature Algorithm (DSA) which is based on Elliptic Curve Cryptography (ECC). ECDSA was first proposed by Scott Vanstone in 1992 [26] and is today accepted as an IEEE, ANSI, ISO and NIST standard. ECDSA has received world wide acceptance by the security experts and the research community. In comparison to other systems, ECC uses smaller parameters to provide equivalent level of security which result in higher processing speed and smaller keys. ECDSA key pair includes a random multiple of the base point called the public key and an integer used to generate the multiple called the private key. ECDSA provide entity authentication, data integrity and non-repudiation services.

3. REVIEW OF THE SCHEME IN [10]

In this section, we briefly review the scheme in [10]. The operations of the scheme can be divided into two phases: Registration phase and Login and authentication phase. Table 1 lists some notations that will be used throughout the paper.

3.1. REGISTRATION PHASE

During registration phase, the user registers to the remote server with his/her username and IMEI. In case the user changes his/her mobile phone, he/she must contact the service provider in order to replace old IMEI with new IMEI.

The IMEI (International Mobile Equipment Identity) number is a 15-digit number which is used to recognize the GSM/DCS/PCS mobile phones in the network service. The IMEI number is unique for every mobile phone. As of 2004, its format is AABBBBBB-CCCCCC-D where first 8 digits (AABBBBBB) are the verification code of the respective country, the next 6 digits (CCCCCC) are the serial number of the mobile phone and the last one digit (D) is a control related number.

3.2. LOGIN AND PASSWORD AUTHENTICATION PHASE

When the user wants to avail the services provided by the remote server, he/she has to login to the system. The login and authentication phase of scheme in [10] is depicted in Fig. 1 and consists of the following steps:

Step 1. To login to the system, the user sends a login request to the server.
Step 2. On receiving the request, the server generates a random number and sends a code to login program so the program can get the image of the number.
Step 3. Based on the received code, a multi-digit number (for example six digits) is generated. The program then draws these multi-digit numbers in a crooked shape. For each digit, a random font and a random degree of skew is chosen. Finally these images are placed next to each other to form a six digit number.
Step 4. The login program shows this image to the user and asks them to recognize and type the number shown in the image (CAPTCHA). The user enters his/her username and the number shown.
Step 5. After entering the username and the number shown, the login program gets the IMEI code from the mobile phone. Then the login program uses MD5 algorithm to generate a code from combined IMEI code and the number entered by the user.
Step 6. The login program sends this MD5 code along with the username and the image code to server.
Step 7. The server extracts the user's IMEI from the database by username. It then finds the correct number according to the image code which was sent to the user. The MD5 algorithm is then applied to the extracted IMEI code and the correct number to obtain MD5 code. If this MD5 code is equal to the user's response, the user is authenticated and he/she is granted access to enter the site otherwise the access is denied.

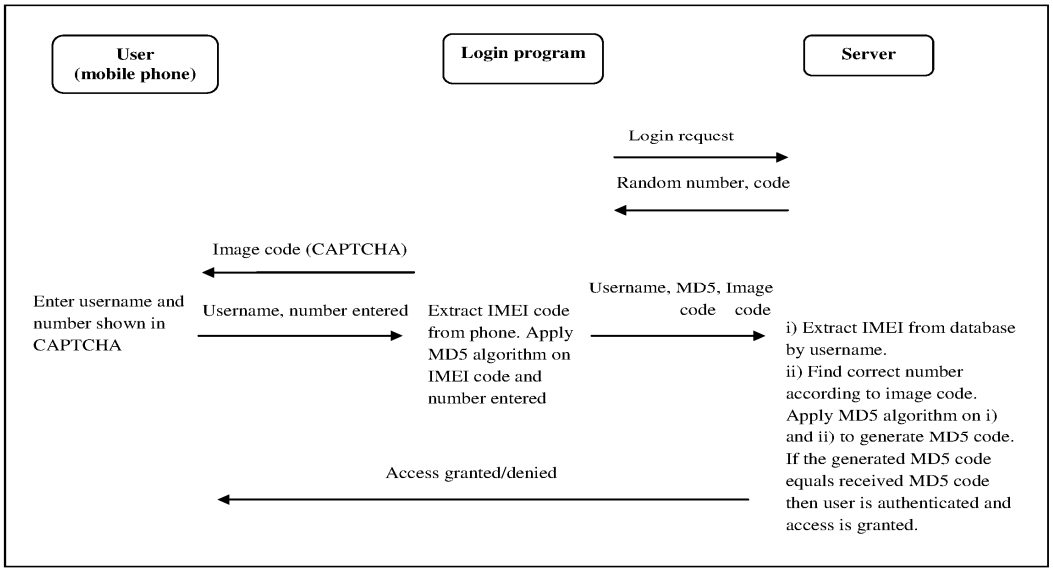

Fig. 1. Login and authentication phase of the scheme in [10]

4. SECURITY WEAKNESSES

The authors of [10] proposed a passwordless user authentication method for mobile phones based on IMEI code and CAPTCHA. Though the scheme is simple, it suffers from various security flaws as explained below:

4.1. MOBILE PHONE LOST/STOLEN PROBLEM

If the user accidently loses his phone or if it is stolen by some adversary, the user's secret information can be easily compromised by the adversary. Let us assume that the username is somehow revealed to adversary. The adversary can impersonate as a legitimate user and send a login request to the server. It can then enter the username and the CAPTCHA shown. The login program gets the IMEI code from the lost/stolen mobile phone and do further computation. Thus the security of the system can be breached very easily.

4.2. MAN-IN-MIDDLE ATTACK

The scheme in [10] is vulnerable to man-in-middle attack. The attacker can sit between the user and server in place of login program and intercept the messages exchanged between the user and server. The attacker can even alter the messages and then send it to them.

4.3. WEAKNESSES OF MD5 ALGORITHM

The scheme in [10] uses MD5 (message-digest) algorithm which is commonly used as a means to verify integrity of data. Over years, many serious flaws [30-32] have been found by researchers regarding MD5 algorithm. Some of the security attacks which can be launched on MD5 are collision attack, chosen-prefix collision attack, preimage attack and rainbow table attack. The security of MD5 hash function is severely compromised and is thus unsuitable for further use.

4.4. INSIDER ATTACK

The scheme in [10] stores username and the corresponding IMEI on the remote server. A privileged insider can easily steal these user credentials from the server. He can further steal the mobile phone of user or illegally access it, use these credentials and thus gain access to the server as a legal user. Thus the security of the system can be easily compromised.

4.5. SERVER SPOOFING ATTACK

Also known as server impersonation attack, this attack can be easily launched on scheme [10]. The adversary may set up a fake server and when the user sends a login request, it may pretend itself as a legitimate server. The adversary may then generate a random number and send a code to login program so the program can get the image of the number. It can carry out further steps in a similar fashion. Thus, without knowing any user credentials, it can successfully launch server spoofing attack.

4.6. NO SECURITY AND USER ANONYMITY

The communication between client and remote server in scheme [10] is carried out in plaintext. No encryption or hash function is employed, thereby making it an easy target for attackers and exposing the identity of the user to the third parties.

5. PROPOSED SCHEME

In this section, we propose a new passwordless authentication scheme for smart phones which not only resolve all the weaknesses of password based schemes but also provide robust security. The proposed scheme relieves users from memorizing and storing long and complicated passwords. The proposed scheme uses ECDSA which is based on Elliptic Curve Cryptography (ECC). ECC is one of the strongest public-key cryptographic systems known today. Compared with RSA, Rabin and Elgamal cryptographic systems, ECC has remarkable strength and efficiency advantages in terms of bandwidth, key sizes and computational overheads. ECC can therefore be efficiently implemented in resource constraint devices like smart phone. Furthermore, the proposed scheme incorporate CAPTCHA which play an important role in protecting the web resources from spamming and other malicious activities. Table 1 denotes the parameters used in the proposed scheme and Fig.2 gives an overview of the proposed passwordless authentication scheme.

Table 1
Notations

Notations used	Description
IMEI	International Mobile Equipment Identity
CAPTCHA	Completely Automated Public Turing test to tell Computers and Human Apart
Q	A prime number
a, b	Integers that specify the elliptic curve equation $y^2=x^3 + ax + b$ defined over Z_q
G	A base point on the elliptic curve equation represented by $G = (x_g, y_g)$ of order n such that $n .G = O$, where n is a large prime number
N	Order of point G, i.e. n is the smallest positive integer such that $n .G = O$. This is also the number of points on the curve.
D	Private key of user; $d \epsilon [1, n-1]$
Q	Public key of user; $Q = dG \epsilon E_q(a, b)$
$H( )$	A collision resistant one-way secure hash function
$\rightarrow$	Public channel
$\Rightarrow$	Secure channel

The proposed scheme consists of the following four phases: Registration phase, Precomputation phase, Login and Authentication phase and User eviction phase.

5.1. REGISTRATION PHASE

Before the system begins, the server selects a large prime number q and two integers a and b, where $q > 2^{160}$ and $4a^3 + 27b^2 \neq 0$. The server then selects an elliptic curve equation $y^2=x^3 + ax + b$ defined over Z_q. Let $G = (x_g, y_g)$ be the base point in $E_q(a, b)$ whose order is n. The order n of a point G on an elliptic curve is the smallest positive integer n such that $n .G = O$ where O is a point of elliptic curve at infinity.

Step 1. The user registers to the server with his/her unique IMEI number over a secure channel.
Step 2. The server verifies the validity of the user by checking if the sent IMEI number is valid or not and belongs to the corresponding user. The server further checks its database to make sure that the user is not already registered. After successful verification, the server stores each legal user's unique IMEI number in its database. The server further shares the global parameters q, a, b, G, n with the user. Table 2 depicts the verifier table which stores user's unique IMEI number, its public key and the status bit. The status bit indicates the status of the user i.e. when the client is logged into the server, the status bit is set to one, otherwise it is set to zero.

Table 2
The verifier table

IMEI	Public Key	Status-bit
Unique IMEI of A	Q_A	0/1
Unique IMEI of B	Q_B	0/1
Unique IMEI of C	Q_C	0/1
…	…	…

Step 3. After the user receives the global variables, he/she activates the smart phone.

5.2. PRECOMPUTATION PHASE

After activating the smart phone, the user generates a pair of keys, one private and one public. The smart phone chooses a random d from $[1, n-1]$ which forms the private key of the user and is stored somewhere safe in the smart phone. It also computes the public key $Q = d.G$ which is a point in $E_q(a, b)$. The digest value of public key along with user's IMEI number is communicated to the server. The server stores the user's public key Q corresponding to its IMEI number in the verifier table.

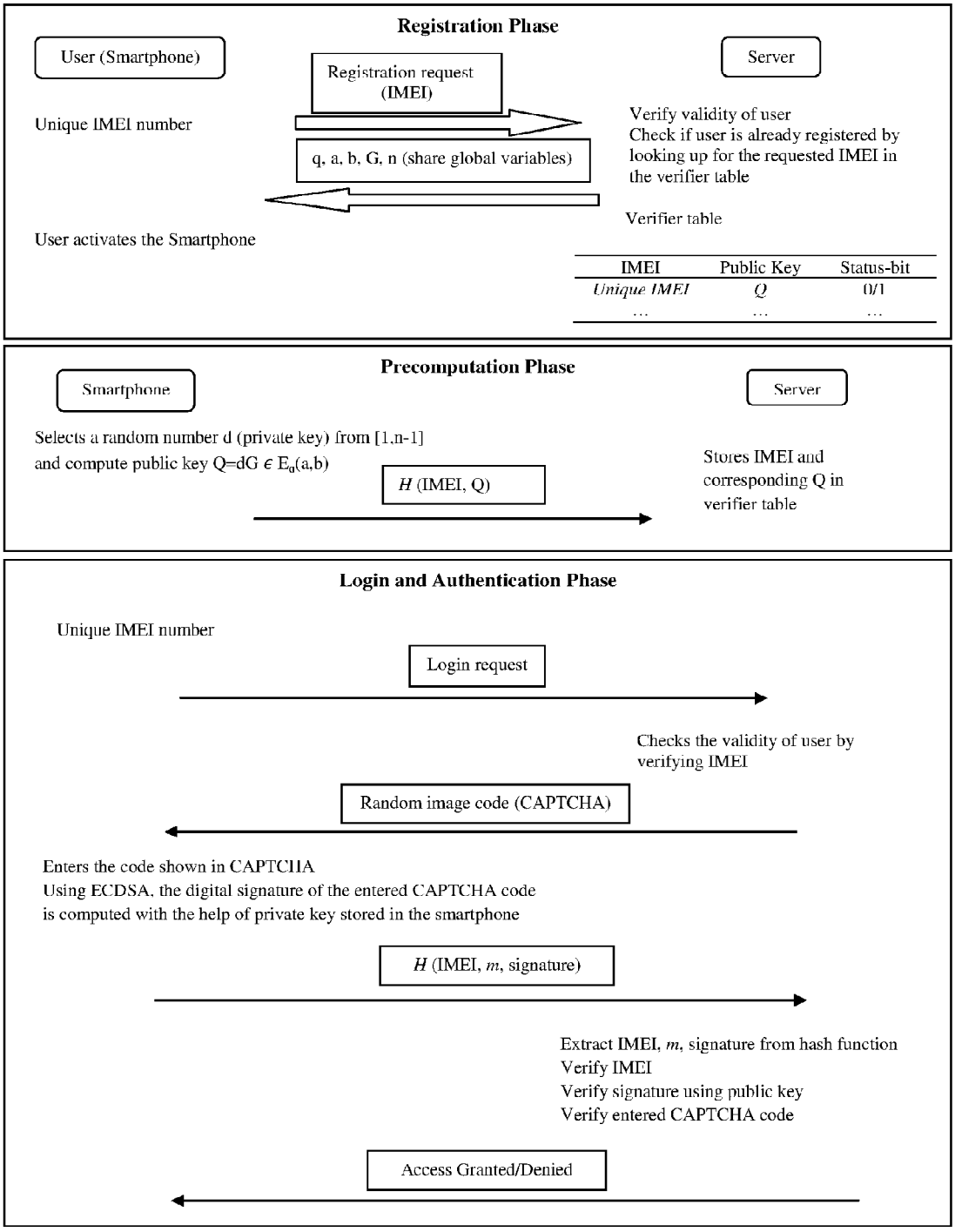

Fig. 2. Proposed passwordless authentication scheme

5.3. LOGIN AND AUTHENTICATION PHASE

The login and authentication phase has the following steps:

Step 1. When the user wishes to access the resources of the remote server, he/she sends a login request to the server through his/her smart phone.
Step 2. On receiving the login request, the server first checks the validity of the user by verifying its unique IMEI number. It further checks if the user is registered by checking its database. If the user is registered, the server sends a random image code i.e. CAPTCHA to the requesting user.
Step 3. The user simply enters the code which is shown in the CAPTCHA. Using ECDSA, the digital signature of the entered CAPTCHA code m is computed with the help of private key stored in the smart phone as explained below:

1. Select a random number k, $k \epsilon [1, n-1]$
2. Compute point $p = (x, y) = kG$
3. Compute $r = x \bmod n$. If $r = 0$ then goto step 1
4. Compute $t = k^{-1} \pmod n$
5. Compute $e = H(m)$, where H is the hash function
6. Compute $s = k^{-1} (e + dr) \bmod n$. If $s = 0$ then goto step 1.
7. The user's signature for entered CAPTCHA code m is the pair (r, s)

The user then transmits the digest value of IMEI number, entered CAPTCHA code m and the signature to the remote server.

Step 4. The remote server extracts the IMEI number, the entered CAPTCHA code m and the signature from the hash function. The server first verifies the validity of IMEI code and then verifies the signature using the following steps:

1. Verify that r and s are integers in the range $[1, n-1]$
2. Using SHA 1, compute $e = H(m)$
3. Compute $z = s^{-1} \bmod n$
4. Compute $u_1 = ez$ and $u_2 = rz$
5. Compute the point $X = (x_1, y_1) = u_1G + u_2Q$
6. If $X = O$, reject the signature else compute $v = x_1 \bmod n$
7. Accept the signature if and only if $v = r$

The server can further verify the entered CAPTCHA code with the sent CAPTCHA. If all the above steps are satisfied, the server grants the user's login request otherwise the request is rejected.

5.4. USER EVICTION PHASE

Step 1. If a user A is evicted by server, the server will delete A's IMEI number and the corresponding public key Q and status bit from verifier table.
Step 2. When this evicted user tries to access the remote server, the server can detect this user since the IMEI number cannot be found in the verifier table.

6. SECURITY ANALYSIS OF THE PROPOSED SCHEME

In this section, the security analysis of the proposed scheme is given for the validation of our claim. The proposed scheme not only alleviates the hassle of long and complicated passwords but also resolves various password related security issues. Furthermore, it is based on ECC which

provides highest strength-per-bit with substantially smaller key sizes making it ideal for resource constraint smart phones. Table 3 gives the comparison of our scheme with related schemes.

The security attributes provided by our scheme are as discussed below:

6.1. BRUTE FORCE ATTACK

In systems where users authenticate using userID and password, the adversaries can easily guess their userID and password, using various methods. One such method is the brute force attack, also known as exhaustive search. In this, the attacker generates every possible combination of user password and tries to authenticate itself as the actual user. If given enough time and provided the user does not change his/her password, the attacker can successfully launch this attack. The proposed scheme does not require any password to authenticate the user. Thus, brute force attack is not possible in our scheme.

6.2. DICTIONARY ATTACK

Another popular attack on password based authentication schemes is the dictionary attack. It is based on the fact that users tend to select memorable words for the password such as names of towns, pets, date of birth etc, thus reducing the number of possible combinations to only meaningful words. The attacker can compile the list of such meaningful words into a dictionary and then launch a search against the system by trying the same user account and words from dictionary file as the password. Dictionary attack is not possible in our scheme as no passwords are used to authenticate the user to the server.

6.3. RAINBOW TABLE ATTACK

In rainbow table attack, the attacker makes use of rainbow table which contains precomputed values. Thus the attacker does not have to calculate all the combinations of possible passwords, which saves time and system resources. Rainbow table attacks are not possible in the proposed scheme as no password is required for authentication to the system. Furthermore, it uses EDSA and CAPTCHA which are based on random factors, thus no such attack is possible.

6.4. SNIFFING ATTACK

In sniffing attack, the attackers try to obtain the user credentials like userID and password over insecure networks. They can later use these credentials and impersonate as legal clients. In the proposed scheme, no passwords or private keys are exchanged between the user and server. Furthermore, before communication, the hashed code of the message to be sent is computed using Secure Hash Algorithm (SHA) which is one way and collision resistant. Thus the proposed scheme can successfully resist the sniffing attacks.

6.5. SERVER SPOOFING ATTACK

In server spoofing attack, the attacker tries to masquerade as a server to know user's secret credentials. The proposed scheme is based on ECDSA in which each message that the server sends to the user is accompanied by the digital signature of the server. The user can easily verify

the authenticity of the server by verifying the signature using its private key. Thus server spoofing attack is not possible in the proposed scheme.

6.6. MAN-IN-THE-MIDDLE ATTACK

In man-in-the-middle attack, the attacker sits somewhere between the user and the server on the network and intercepts the messages exchanged between the user and server. The attacker may even alter the messages and then send it to them. The proposed scheme is based on ECDSA where both the user and server attach their digital signature along with the message. They can easily verify the authenticity of each other by verifying the digital signature using their respective private keys. Such digital signatures are impossible to be forged by the attackers due to intractability of ECDLP and if they do, the receiving entity can easily figure out that it is forged.

6.7. SHOULDER SURFING ATTACK

In shoulder surfing attack, the attacker can look over the user's shoulder while he is typing his password to authenticate to the server. In the proposed scheme, no password is required for authentication. It just requires typing the CAPTCHA which is random for every user. Thus shoulder surfing attack is not possible in our scheme.

6.8. MANY LOGGED-IN USERS' ATTACK

The proposed scheme can resist many logged-in users' attack. In this attack, more than one adversary tries to guess the userID and password of the user and login as legal user. In the proposed scheme, the server maintains a status-bit in the verifier table. Thus, if the user login from his/her smart phone, the status-bit is set to one. In the meantime, if the attackers somehow try to break in into the system, their login requests will be rejected because the status-bit indicates still someone is logged-in.

Table 3
Security comparison of our scheme with related schemes

Security Characteristics	Our scheme	The scheme in [10]	The scheme in [3]
Brute force attack	No	No	Yes
Dictionary attack	No	No	Yes
Rainbow table attack	No	Yes	Yes
Sniffing attack	No	Yes	No
Server spoofing attack	No	Yes	Yes
Man-in-middle attack	No	Yes	No
Shoulder surfing attack	No	No	Yes
Many logged-in users attack	No	Yes	No

The proposed scheme can thus successfully withstand all the security attacks. The robustness of the system depends on the hardness in solving ECDLP, ECCDHP, ECDDHP problems. Furthermore, every smart phone possesses a unique IMEI number. The system works only after it verifies the device's IMEI number and that the requesting user using that smart phone is authenticated. Thus the proposed scheme provides security from any type of system intrusion.

7. FUNCTIONALITY FEATURES OF THE PROPOSED SCHEME

In this section, we summarize the functionality features of the proposed scheme. Table 4 gives the functionality comparison of our scheme with related schemes.

7.1. NO PASSWORD REQUIREMENT

The prime attraction of the proposed scheme is that it do not require any password for logging into the system. It thus saves the user from the hassle of memorizing long and complicated passwords. It also prevents various types of security attacks that are possible in a system due to the use of passwords.

7.2. MUTUAL AUTHENTICATION

The proposed scheme provides mutual authentication between the user and the server. Both user and server generate their own public and private key pair. They use these keys for digital signature and authentication. Thus, they can successfully authenticate each other by verifying the digital signature using their private key.

7.3. PREVENTION OF CLOCK SYNCHRONIZATION

The clock synchronization problem arises due to the use of time stamps used in login systems to prevent replay attacks. The proposed scheme is based on the use of random numbers and random CAPTCHA rather than time stamps to prevent replay attack and thus no clock synchronization problem exist in the proposed scheme.

7.4. NO EXTRA HARDWARE DEVICES

The proposed scheme does not require the use smartcards or any other external hardware device. For the implementation of the proposed system, the user just requires a smart phone with Wi-Fi facility. Today smart phones are accessible to all the users all over the world, that too at very reasonable prices. Thus the proposed scheme can be easily implemented anytime and anywhere without any extra hardware requirement.

7.5. USER-FRIENDLY

The proposed scheme is based on the concept of passwordless authentication. The users thus don't have to worry about password selection and memorizing. The password based authentication has always been cumbersome for the users because human memory is transient and remembering a large number of long and complicated passwords is impossible. To overcome this shortcoming, the proposed scheme uses passwordless authentication which is based on CAPTCHA and ECDSA. The proposed scheme can be easily used by users of all ages, even children.

7.6. USER ANONYMITY

During the communication between user and remote server over an insecure network, the attacker or third parties may know the identity of the user by intercepting the messages exchanged between them. Thus providing user anonymity is very important. In the proposed scheme, no personal information about the user is involved in the messages exchanged between user and server. Furthermore, the digest value of the messages to be transmitted is computed using secure one way hash functions and then this digest value is transmitted. Thus the proposed scheme ensures user anonymity.

7.7. ANTI-SPAM

Providing secure communication over network systems is a challenging task. The attacker launches these attacks through various automated trials, via internet bot programs which do not require any user interaction. Often spam is used by attackers to obtain credit card information, passwords and other security sensitive data. The proposed scheme incorporate CAPTCHA which play an important role in protecting the web resources from spamming and other malicious activities.

Table 4
Functionality comparison of our scheme with related schemes

Functionality comparison	Our scheme	The scheme in [10]	The scheme in [3]
Password requirement	No	No	Yes
Mutual authentication	Yes	No	Yes
Prevention of clock synchronization	Yes	Yes	Yes
Extra hardware devices	No	No	Yes
User friendly	Yes	Yes	No
User anonymity	Yes	No	Yes
Anti spam	Yes	Yes	No

8. CONCLUSIONS

In this paper, we have proposed a new passwordless authentication scheme for smart phones that exploits the advantages of CAPTCHA and ECDSA. ECDSA provide entity authentication, data integrity and non-repudiation services. CAPTCHAs defend the system against harmful internet bot programs and other malicious activities. In contrast to traditional password based authentication schemes, the users don't have to worry about password selection and memorizing. The users can easily login from their smart phones and access the web resources. The security and functionality analysis of the proposed scheme is given for the validation of our claim. The proposed scheme thus provides robust security and efficiency from any type of system intrusion.

REFERENCES

[1] C. C. Chang, C. Y. Lee, Y. C. Chiu, "Enhanced authentication scheme with anonymity for roaming service in global mobility networks," Computer Communications, vol. 32, no. 4, pp. 611-618, 2009.

[2] M. Peyravian, N. Zunic, "Methods for protecting password transmission," Computers & Security, vol. 19, pp. 466-469, 2006.

[3] C.T. Li, "A new password authentication and user anonymity scheme based on elliptic curve cryptography and smart card," IET Information Security, vol. 7, No. 1, pp. 3-10, 2012.

[4] S. Kalra, S. Sood, "Secure authentication scheme for Iot and cloud servers," Pervasive and Mobile Computing, In press 2015.

[5] H. L. Yeh, T. H. Chen and W.K. Shih, "Robust smart card secured authentication scheme on SIP using Elliptic Curve Cryptography," Computer Standards & Interfaces, vol. 36, no. 2, pp. 397-402, 2014.

[6] Yoon E., Yoo K, "Robust biometric-based three-party authenticated key establishment protocols," Int. J. Comput. Math., vol. **88**, Issue 5, pp. 1144–1157, 2011.

[7] D. He, D. Wang, "Robust Biometric-Based Authentication Scheme for Multiserver Environment," IEEE Systems Journal, vol. PP, Issue 99, pp 1-8, February 2014

[8] Sasse, M.A., Brostoff, S., Weirich, D., 2001. Transforming the 'weakest link'—a human/computer interaction approach to usable and effective security. BT Technical Journal 19, 122–131.

[9] Brown, A.S., Bracken, E., Zoccoli, S., Douglas, K., 2004. Generating and remembering passwords. Applied Cognitive Psychology 18, 641–651.

[10] M. Shirali-Shahreza and S. Shirali-Shahreza, "Passwordless Login System for Mobile Phones Using CAPTCHA," Proc. 49th International Symposium ELMAR-2007, Zadar, Croatia, September 2007.

[11] Ives, B., Walsh, K.R., Schneider, H., 2004. The domino effect of password reuse. Communications of the ACM 47 (4), 76–78.

[12] K. Chellapilla et al., "Building Segmentation Based Human-Friendly Human Interaction Proofs (HIPs)," Proc. 2nd Int'l Workshop Human Interaction Proofs (HIP 05), LNCS 3517, Springer, 2005, pp. 1-26.

[13] T. Converse, "CAPTCHA Generation as a Web Service," Proc. 2nd Int'l Workshop Human Interactive Proofs (HIP 05), LNCS 3517, Springer, 2005, pp. 82-96.

[14] L. von Ahn, M. Blum, and J. Langford, "Telling Humans and Computer Apart Automatically: How Lazy Cryptographers Do AI," Comm. ACM, vol. 47, no. 2, 2004, pp. 57-60.

[15] M. Shirali-Shahreza and S. Shirali-Shahreza, "Question-Based CAPTCHA", Proceedings of the International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007), Sivakasi, India, December 13-15, 2007, Vol. 4, pp 54-58.

[16] F. Ali Bin Hamid Ali, F. Bt Karim, "Development of CAPTCHA system based on puzzle," Proc. International Conference on Computer, Communications, and Control Technology (I4CT), Langkawi, Sept. 2014, pp. 426-428.

[17] A.E. Ahmad, J. Yan, and L. Marshall, "The Robustness of a New CAPTCHA," Proc. 2010 European Workshop System Security (EuroSec 10), ACM Press, 2010, pp. 36–41.

[18] J. Yan and A.E. Ahmad, "Usability of CAPTCHAs or Usability Issues in CAPTCHA Design," Proc. 4th Symp. Usable Privacy and Security (SOUPS 08), ACM Press, 2008, pp. 44–52.

[19] H.S. Baird, M.A. Moll, and S.Y. Wang, "A Highly Legible CAPTCHA that Resists Segmentation Attacks," Proc. 2nd Int'l Workshop Human Interaction Proofs, LNCS 3517, Springer, 2005, pp. 27–41.

[20] Jung-San Lee, Ming-Huang Hsieh, "Preserving user-participation for insecure network communications with CAPTCHA and visual secret sharing technique," IET Networks, vol. 2, no. 2, pp. 81-91, 2013.

[21] H. D. Truong, C. F. Turner, C. C. Zou, "iCAPTCHA: The Next Generation of CAPTCHA Designed to Defend against 3rd Party Human Attacks," Proc. IEEE International Conference on Communications (ICC), Kyoto, June 2011, pp. 1-6.

[22] A. Chandavale, A. Sapkal, "An Improved Adaptive Noise Reduction for Secured CAPTCHA," Proc. 4th International Conference on Emerging Trends in Engineering and Technology, Port Louis, Nov. 2011, pp. 12-17.

[23] Enge, Elliptic Curves and Their Applications to Cryptography — An Introduction, Kluwer Academic Publishers, 1999

[24] J. Silverman, The Arithmetic of Elliptic Curves, Springer-Verlag, 1986.

[25] A. Menezes, Elliptic Curve Public Key Cryptosystems, Kluwer Academic Publishers, Boston, 1993.

[26] S. Vanstone, "Responses to NIST's Proposal", Communications of the ACM, 35, July 1992, 50-52.

[27] A. Gupta, A. Jain, A. Raj, A. Jain, "Sequenced Tagged Captcha: Generation and its Analysis," Proc. International Advance Computing Conference, India, March 2009, pp. 1286-1291.

[28] W. Stallings, Cryptography and Network Security: Principles and Practices. Prentice Hall (2004).

[29] http://www.emarketer.com/Article/2-Billion-Consumers-Worldwide-Smartphones-by-2016/1011694

[30] T. Xie, F. Liu and D. Feng, "Fast collision Attack on MD5," Cryptology ePrint Archive, Report 2013/170 (2013), http://eprint.iacr.org/

[31] X. Wang and H. Yu, "How to Break MD5 and Other Hash Function," Advances in Cryptology-Lecture Notes in Computer Science, vol. 3494, pp.19-35, 2005.

[32] M. Stevens, "Single-block collision attack on MD5," Cryptology ePrint Archive, Report 2012/040 (2012), http://eprint.iacr.org/

[33] Miller, V.S., Use of elliptic curves in cryptography, In: Proceedings of Advances in cryptology-CRYPTO'85 (1985) 417–26

[34] Koblitz N, Elliptic curve cryptosystem. Journal of Mathematics Computation 48 (177) (1987) 203–209.