An Adaptive Two-Factor Authentication Scheme Based on the Usage of Schnorr Signcryption Algorithm

Hussein Al Bazar*[1], Ahmed Abdel-Wahab[1,2], Marwan Alshar'e[3], Abdallah Abualkishik[4]
[1 2]Faculty of Computer Studies, Arab Open University (AOU), Riyadh, Kingdom of Saudi Arabia
[2] Faculty of Engineering, Al-Azhar University, Cairo, Egypt
[3 4] Faculty of Computing and Information Technology, Sohar University, Sohar, Oman
E-mail: halbazar@arabou.edu.sa, a.rakha@arabou.edu.sa, mshare@su.edu.om, AAbualkishik@su.edu.om
*Corresponding author

Keywords: *authentication, encryption, two-factor authentication, Schnorr signcryption*

Received: January 19,2023

With the current continuous increase in the usage of communication technologies and data transmission amount, the available web services and mobile applications are used to access and complete several online activities such as online education, e-commerce, and online financial transactions. The needs for a strong and secure access control authentication solution for users' data and the available online resources are crucial. Authentication can play a major role in protection by implementing a solution, which allows users to prove their identities and provide access permission to all legitimate users only. This paper proposes two-factor authentication scheme based on the usage of a modified copy of the Schnorr Signcryption algorithm. The proposed scheme achieves the target goal of this paper, which provides the users with an efficient solution that can be implemented in any authentication solution as the second factor, with no special requirement and can be used to authenticate the users and control recourses access permission. Moreover, a Java-based application is developed to examine the proposed scheme in a client-server architecture where it is found that the solution can successfully authenticate authorized users and reject any unauthorized access request with an adequate security protection level and acceptable performance.

Povzetek: V članku je predlagana rešitev z dvofaktorsko avtentikacijo na osnovi modificiranega algoritma Schnorr Signcryption, ki omogoča učinkovito avtentikacijo uporabnikov in nadzor dostopa do virov. Rezultati preizkusa kažejo zadostno varnost in sprejemljivo zmogljivost.

1 Introduction

Nowadays, the advancements in network connectivity, communication speed and the usage of mobile phone devices have increased to complete most daily activities amongst various types of services such as messaging, e-commerce, healthcare, and online financial transactions. Additionally, the important information, which can be transferred online has increased [16, 38]. With the subsequent increase in communication technology usage, the transmitted amount of critical messages between communicated parties and the number of IoT devices reached 20 billion with a data size of 40 Zettabytes. The security and protection of exchanged messages have become a vital requirement and more challenging [10, 35]. Moreover; the decrease in the cost of electronic devices, along with the increasing number of online users, the increase in various types of services, applications, and online financial transactions, activities are accessed by using the Internet with end-user devices such as computers, smartphones, and tablets assuring the need for secure, efficient, and reliable techniques for authentication and identification is important while keeping the process of exchanging information protected from any external sources of threats [11, 22, 31].

There is a need for a strong user authentication scheme, and access control solution that can protect users' online records and data, which are accessed through different online services e.g., online shopping and online banking [26]. Authentication is the process of proofing the claimed identity by receiving data from the user e.g., password, smartcard, and biometric data and then comparing this data with a stored database. The process can be accomplished by using one or more mechanisms to grant the user permission to access some of the services, communications, or data access [2, 16, 23]. Authentication provides an effective method for access control by implementing more than one single factor that is used to authenticate the connected user, which improves security systems and the privacy of data [5, 7]. Authentication is crucial in the context of requesting access permission for important data such as patient healthcare data, online banking data, access control, governmental applications, educational institutions, etc. [3, 24]. The usage of conventional single factor authentication methods e.g., users' password, PIN or biometrics is inadequate and insufficient to protect the data from unauthorized access. To achieve adequate security protection, a single factor can be accepted as an identity verification option, and this highlights many security concerns [3, 12, 39].

Authentication is an essential concept that is implemented to provide safeguarding protection against illegitimate access requests from a threat actor, which is attempting to gain access through to different sensitive data, applications or services [23, 37].

In general, authentication can be performed by using a single factor from available factors, which is related to users' identity as what you know, where users' password or identification numbers can be used within this factor, and from what you have where security token or users' card or one-time password can be used within this factor, or from what you are such as biometric data [2, 30]. Further, authentication is classified into three main types. The first factor comprises the single factor authentication, which is considered to be the simplest and less complex. However, it is considered to be the weakest type of authentication since it is less reliable, inadequate, and grants critical access permission such as financial transactions due to its defenselessness [2, 16, 23]. The second factor comprises the two-factor authentication, which is considered to be the more secure form of authentication in comparison with the single factor, and which relies on using two factors such as something you know and something you have [20, 40]. The third factor comprises the multifactor authentication, which is implemented by using a multifactor verification such as something you know, something you have and something you are. Additionally, it is considered to be more complex as it provides more overhead. Nonetheless, it is a more secure and reliable form of authentication than other types of the indicated examples [2, 16]. Thus, two-factor and multifactor authentication can be implemented to provide proof of users' identity and to ensure a further secure system[2, 3]. Moreover, according to the study conducted by [9], it is shown that the global two-factor authentication market is worth 4.8 USD billion by the end of 2020, which continuously increases every year as a result of the increment of online payment methods, such as PayPal, Stripe, and the increase in Cybercrime cases. It is expected by 2027 to reach 14.5 USD billion (see Figure 1).

The two-factor authentication can be implemented and used as a strong solution to extend the security level by implementing multiple elements for improving the security level such as what a user knows and what a user has. [4, 12, 27]. Therefore, two-factor mobile authentication has become a major need to achieve the required security and privacy level [2, 7]. In two-factor mobile authentication, smart devices are used as a second level of authentication, which requires human interaction with the smart device [40].

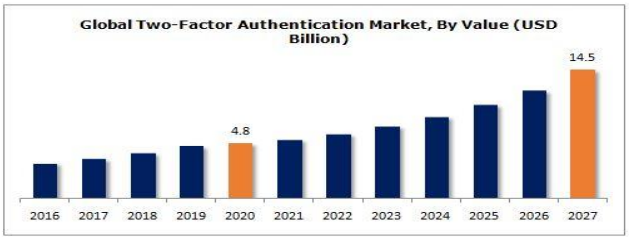

Figure1: Global two-factor authentication market.[9]

The main contribution of this research is to propose a Two-factor authentication scheme based on the usage of a modified copy of the Schnorr Signcryption algorithm, which can be used as a second factor. The proposed solution aims to provide an adequate level of data, confidentiality, and integrity protection from different unauthorized access attempts. The proposed scheme able to be used as a second factor within any single factor such as a smart card or conventional username and password, and can be implemented in several client-server architectures without any restrictions where the end users' device can either be a smartphone or computer.

The outlines of the paper are organized as follows: Section 2 presents the related research of different two-factor authentication solutions. In section 3, an overview of the Schnorr Signcryption algorithm is presented. Details of the proposed scheme are provided in Section 4. Implementation and obtained results discussion are provided in Section 5. The conclusions of this research are summarized in Section 6.

2 Related works

Different solutions for two-factor authentication have been proposed by using various types of technologies such as biometrics, IoT, One- time Password (OTP), Smart card-based solutions, Cryptography, etc. In the context of IoT and wireless communication, a secure wireless body area networks (WBANs) two-factor authentication scheme is proposed based on a hash-chain and forward secure authentication in healthcare IoT to protect patients' data, which is collected by using wireless sensors that are forwarded to local gateways, which are connected to doctors' and nurses' mobile devices [14]. The authentication schema is proposed for electronic healthcare services to allow patients to gain access to some medical services remotely. The proposed scheme was designed based on the usage of searchable encrypted keywords and Information Retrieval (IR), including the decryption data key, which is available for authorized users only e.g., doctors and patients [31]. An improved two-factor authentication scheme for the Wireless Medical Sensor Network (WMSN) is proposed to allow doctors to access real-time patients' data, which is collected from sensors that are attached to the patients' body and sent by using secure wireless communication. The proposed scheme is designed based on the implementation of an AES symmetric key system, one-way SHA1 hash function, and X-OR operation [36].

The two-factor authentication protocol is proposed to protect IoT devices from a range of attacks such as spoofing or impersonation [4]. The proposed protocol is designed by using physical forms of identification such as fingerprints, which are inherently considered a form of identification that is not cloned. Two characteristics of wireless signals, particularly, the received signal strength and link quality indicator are used in order to determine the IoT device's location and to lock the authentication of devices to a fixed predefined location [4]. Further, they are used to protect IoT devices from unauthorized control physical and cloning attacks. The lightweight two-factor

authentication scheme of IoT devices is proposed to provide a more contingent IoT devices authentication process. In the proposed schema, a physical unclonable function (PUF) has been used as the second authentication factor providing an unduplicated one-way function with a password or shared secret key, which is used as the first step in the authentication factor [15]. Moreover, the three-factor authentication protocol is suggested for wireless sensor networks that are used for providing session keys [35].

The two-factor authentication prototype method is proposed for transit applications. The first factor uses a PIN, which is stored on the smart card memory that is used during the login phase and verified by the server [22]. The second factor that is used in this method is based on the One-Time Password (OTP), which is generated by using the SHA-256 algorithm and is sent to users' device by using the ZigBee network. The generated OTP is entered on an Arduino-based kiosk machine for verification [22]. In a simple client-server, the Merkle tree-based One-Time Password (MOTP) algorithm is proposed to overcome the problem of leakage of the shared secret key between the client and the server to generate the OTP [39]. In this proposed algorithm, a combination of the T/Key OTP algorithm and Merkle tree is used to construct many OTPs with a lower processing time, along with a more efficient execution and verification performance [39].

The main aim is to provide a more secure user authentication solution for online transactions that are carried out by using mobile phone devices and to provide a more secure OTP generation solution. The two-factor authentication is based on quantum computing, which is used for generating quantum OTP (QOTP). By using quantum operations, the generated QOTP is integrated with user biometrics, which is shared with the server at the initialization phase for user authentication [30]. Additionally, two-factor authentication is proposed based on the usage of an alphanumeric password and a graphical password. In this proposed solution, the OTP is used for authentication, which is generated by using the cryptography algorithm, and which combines many parameters such as International Mobile Subscriber Identity (IMSI), PIN, Date of birth of user, and Username. The user will have an option either to receive a generated OTP from the server or to be generated by a user's mobile device and validate it by using the same used cryptographic algorithm [18].

Instead of using hardware token devices as computer-based software tokens, a two-factor authentication method is proposed by using a user's smartphone. In this method, using the IMSI number, ATM PIN, Timestamp, Date of birth of user, and username, an OTP can be generated by using the Cryptographic algorithm Secured Hash Algorithm (SHA1) at the server. This allows sending an SMS to the user's smartphone. In this context, the smartphone is used as a token to locally generate an OTP, which is valid for a short period of time and is validated by using the same secure cryptographic algorithm [1]. The two-factor authentication method is proposed by taking advantage of the multi-tasks, which can be carried out by using a mobile phone device. In this method, a mobile phone is used instead of using the hardware token or cards with the software application that is installed in it [3]. Furthermore, the OTP is locally generated by using a range of unique factors that are associated with a particular mobile phone. The generated password is sent to the server, which uses the same factors that are used with the client for generating the password. If both generated passwords show a correct match, the authentication process of the mobile phone is successfully completed. If an unsuccessful match has been found, an SMS request is sent from the client along to the server and an OTP is generated at the server, which is sent to the client to be used for a short period of time [3].

To reduce the interaction between users and their mobile phones, there is a need for a secure communication channel among the connected devices. Therefore a sound proof two-factor authentication is proposed based on the distance (proximity) between the computer device and the proximity of the user's mobile phone [17]. Based on this solution, the first authentication factor is based on the username and password, which is sent to the web server. In the second step, the local computer ambient noise with its integrated microphone will encrypt it, and finally, it will send the recorded audio to the mobile phone device. The mobile device will decrypt the recorded audio and compare it with its local recorded ambient noise to determine that both deceives are located in the same environment or not and to inform the server to accept or reject the login request based on the obtained compared results [17]. For a more transparent, simple and less tedious interaction environment between the user and the device, a transparent two-factor authentication (T2FA) method based on a PUF and voiceprint is proposed [40]. Based on this method, the second factor is divided into two authentications; the first authentication authenticates the user's mobile phone by using the PUF, and the second authentication is used to monitor the case when a mobile phone and the computer login are located in the same environment by implementing a background voiceprint comparison between the computer and user's mobile phone environments [40].

Similarly, the TouchIn two-factor mobile authentication system is proposed based on a single or multiple-finger use of the touch screen that is used for a random drawing and unlocking of the mobile device. In the proposed system, mobile users are required to draw a random geometric curve, which is considered to be known as a curve password. An authentication template is extracted by using multiple features such as x, y-coordinates, accelerated finger pressure, and hand geometry [32]. Furthermore; acoustics and vision-based two-factor authentication systems for smartphone users' method is proposed. In this method, Convolutional Neural Network (CNN) and visual facial landmark locations are implemented to extract unique acoustic and facial landmark features, where the Support Vector Machine (SVM) method is used for the final authentication processes [42].

A Secure elliptic curve-based scheme is proposed for data protection on a USB device. By using a USB interface through this scheme, users pass through a mutual

authentication process, and a session negotiation key is generated between the server and the user. At each time, users encrypt data on a USB device, which improves data integrity to be automatically removed whenever there is no more process to be conducted within that session [5]. A stolen verifier attack-resistant authentication scheme is proposed based on the use of a graphical password for e-services and can be implemented with smart cards or a USB token [26]. In this scheme, a graphical password is used instead of using a text password by uploading multiple images by the user three times on a frame with 3x3 sized images where a user selects one image from each frame with a dynamically generated image ID as a graphical password. Each image is hashed and a final hash result is stored on the server during the registration process. Finally, the three smart card parameters are personalized by the server and are sent to the user by a secure channel [26]. A multi-server two-factor-and-key-agreement authentication scheme is proposed using elliptic curve public key cryptography (ECC). In this proposed scheme, users and multiple servers are registered to a registration center where users can employ a unique ID and password, which are used to provide the authentication-and-key-agreement with multiple servers to ensure that they cannot be traced and completed anonymously for users with perfect forward security [38].

A smart card-based two-factor authentication scheme is proposed by using an applied pi calculus, and a Computational Diffie–Hellman (CDH) secure session key is used to provide protection from a password attack, and to prevent the adversary that guesses the correct password even if all information on the smart card is exposed [37]. Furthermore, an improvement of Kim's smart card-based two-factor authentication schema is proposed to combat a de-synchronization attack with no performance reduction by adding additional requirements in terms of storage, computational, communication, and by providing a secure privacy-preserving scheme for real-life application environments [33].

A systematic framework consisting of an independent criteria set is proposed to evaluate the most typical two-factor authentication [34]. The smart-card-based password two-factor authentication schema is produced by integrating a defensive tactic of honeywords along with the proposed fuzzy-verifiers method, which fulfils all the criteria listed in the proposed evaluation framework [34]. Moreover, a smart card-based password two-factor authentication protocol is proposed as an enhancement of the Yang et al scheme. In the proposed protocol, key-compromise impersonation resilience is added to the original Yang et al. scheme as one of the more important requirements, which provides protection against accessing the server's service and the server's long-term key that is compromised [25].

The cryptographic technique is developed from blockchain settings, which is used to integrate security policies on banking systems [6]. In this system, the idea is to conduct the usage of the decentralized trusted authority that is selected as a committee by the account holder and will share secret answers known as a smart contract. This contract can also be used as a supervision tool to invoke a two-factor authentication procedure for any exceptional case such as a key recovery procedure or extreme amount fund transactions [6]. Two-factor authentication and verification are proposed for the Bitcoin wallet. In this research, the Elliptic Curve Digital Signature Algorithm (ECDSA), which is a two-party signature protocol, is used to sign the overall transaction with the senders' private key where the users' smartphone is used as the second authentication [19].

A Mobile phone device two-factor authentication system is based on the usage of a wrist-worn wearable Photoplethysmography (PPG) sensor, which is proposed for nonintrusive and secure mobile authentication [7]. In this system, users can use a signature, password, or PIN as a first factor, and a cancelable PPG signals template for each legitimate that is generated by using single or multiple cardiac cycles and a non-invertible transform method that is stored in the database servers as a second factor. This template is used to verify authorized users who send an access request with the real-time incoming PPG signals [7]. The two-factor authentication system and key agreement scheme for wireless sensor networks (WSNs) is proposed. In this scheme, the use of constant message request parameters is eliminated and dynamic identities are used for users to protect users' privacy. To create any two request messages independently, temporary secret keys are used rather than using log-time constant secret keys from users and sensor nodes to encrypt the communication messages and to minimize redundant variables [8].

To access a secure web portal, the two-factor authentication solution that is integrated with the Pulse Connect Secure features, Apache HyperText Transfer Protocol Server features, and custom 2FA Gateway Application is recommended to provide unauthorized protection against backend applications access by using Uniform Resource Locator (URL) manipulation attacks [21]. The message requests to join the procedure of a Long Range Wide Area Network (LoRaWAN) between network servers and devices, which are not encrypted, and which are susceptible to several types of attacks. The Ethereum blockchain based two-factor authentication mechanism is proposed in order to provide the LoRaWAN joining procedure with an extra authentication level of security for building trust between LoRaWAN network components [10]. Within the proposed solution, an extra independent blockchain network with a smart contract is implemented in parallel with the LoRaWAN network for authentication purposes by performing a match between the LoRa the end device information, which is stored in the blockchain network and the current generated join message request [10].

To conclude, several solutions are proposed and implemented as two-factor authentication solutions aiming to protect important data and provide access permission for authorized users/devices only. Several technologies and methods are used in different domains, for specific purposes to be achieved in predefined domains of implementation such as online banking, healthcare, mobile phone, IoT protection, and web applications. In contrast, this research aims to provide the users with an

efficient solution that can be implemented in any authentication solution as the second factor, with no special requirements and can be used to authenticate the users and control recourses access permission. Table 1 provides a summary of some of the recent above-mentioned related studies in the context of the problem proposed scheme, scheme design, and obtained results that are used for a two-factor authentication solution.

Table 1: A summary of recent two-factor authentication solutions.

Study	Problem	Proposed Scheme	Scheme Design	Results
Breuer et al. (2021) [6]	To safeguard users' transactions in the context of online banking.	Integrate security policies in the context of online banking systems.	Integrate security policies on banking systems in the blockchain settings.	Blockchain decentralized two-factor authentication system. The performance evaluation of the proposed scheme demonstrates that the incorporation of 2FA into Ethereum would result in a minimal increase in cost.
Quadry et al. (2021) [26]	Security threat of stolen verifier attack for e-services.	A Stolen verifier attack-resistant authentication scheme.	Using a graphical password instead of a text password with smart cards or a USB token is used as a second factor.	The proposed scheme is efficient, primarily because it relies on hash functions that are widely known to be computationally efficient. Moreover, the approach has been demonstrated to be impervious to shoulder-surfing attacks.
Cao et al. (2020) [7]	Most mobile two-factor authentication requires user involvement.	Nonintrusive and Secure mobile two-factor authentication via wrist-worn wearables.	Two-stages algorithm to clean heartbeat from PPG signals, a repeatable and non-invertible method for feature templates	PPGPass demonstrates excellent efficiency, security, and usability. Cancelable generated feature templates compared with stored user templates for final authentication decision.
Fotouhi et al. (2020) [14]	To protect patients' data, when patients gain access to some medical services remotely.	Two-factor authentication based on a hash-chain and secure forward in healthcare IoT.	Use searchable encrypted keywords and Information Retrieval (IR), decryption data key which is available for authorized users.	A lightweight secure authentication protocol for body area networks WBANs. Provision of additional security features in contrast to similar schemes.
Nance & Bengston. (2020) [21]	PCS is not capable of providing the necessary features to meet all the authentication requirements.	Two-factor authentication Using Pulse Connect Secure and Apache Server	Integration of PCS features Apache HyperText Transfer Protocol Server and a custom application.	By merging the features and functionalities of PCS, Apache, and a tailored 2FA Gateway Application, the consolidated solution deters unauthorized users from accessing backend applications through URL manipulation.
Sharma & Nene. (2020)	Protect online mobile-based	Two-factor authentication based	Use of quantum (QOTP) generating. Integrate	Overcomes security challenges experienced

[30]	transactions and improve the security of OTP solution based.	on quantum computing.	QOTP with user biometrics, and share it with the server for user authentication.	by existing OTP implementation methods by employing QOTP and biometrics-based aims to authenticate the user rather than the user's device.
Yin et al. (2020) [39]	Leakage of the shared secret key between the client and the server to generate the OTP	Merkle tree-based One-Time Password (MOTP) algorithm	Combination of the T/Key OTP algorithm and Merkle tree to construct many OTPs.	Reducing the number of hash operations to generate the OTP with a lower processing time and more efficient performance.
Aman et al. (2018) [4]	To protect IoT devices from a range of attacks such as spoofing or impersonation.	IoT Two-factor authentication protocol.	Using physical non-cloned identification such as fingerprints, use two features of wireless signals, received signal strength and link quality indicator.	IoT two-factor authentication assumes that the IoT device is equipped with a protected PUF that cannot be modified. Protect IoT systems against spoofing and various other types of attacks.
Gope & Sikdar. (2018) [15]	To provide a more reliant IoT devices authentication process.	Two-factor authentication scheme of IoT devices	Use of PUFs as a second authentication factor. Provide an unduplicated one-way function with a password or shared secret key.	The security and performance assessments indicate that the suggested approach is not only resistant to multiple forms of attacks but also highly efficient in terms of computational efficiency.
Xu et al. (2018) [38]	Overcome authentication-and-key-agreement scheme issues, and improve computational efficiency.	A multi-server two-factor-and-key-agreement authentication scheme.	Using elliptic curve public key cryptography (ECC).	Improved multi-server two-factor authentication scheme can be used as a solution in IoT and WSNs. Offering user anonymity and perfect forward security.
Zhou et al. (2018) [42]	Improve secure and convenient User authentication on smartphones.	Acoustics and vision-based two-factor authentication systems for smartphone users' method.	(CNN) and visual facial landmark locations to extract unique acoustic and facial landmark features. (SVM) for the final authentication.	Smartphones two-factor authentication system with no extra sensors required. Additionally, no image or video-based attacks were detected to be successful in spoofing the system.
Mann & Loebenberger (2017) [19]	Bitcoin wallets safety.	Two-Factor Authentication For The Bitcoin Protocol.	Elliptic Curve Digital Signature Algorithm (ECDSA), and private key via the user's smartphone.	Fully operational prototype was successfully developed and made compatible with the Bitcoin production network for the first time.

Wu et al. (2017) [36]	Security of Patients' real time data collected by wireless sensors	Improved two-factor authentication scheme for the Wireless Medical Sensor Network (WMSN)	implementation of an AES symmetric key system, one-way SHA1 hash function, and X-OR operation	Secure (WMSNs) scheme supported by a comparison with several recently developed approaches for (WMSNs).

3 An overview of signcryption

Data confidentiality and integrity are considered as main goals that should be achieved in cryptography [29, 41]. The encryption algorithm is used for achieving confidentially where integrity is attained by using an authentication technique i.e. digital signatures, there are two operations performed in the order of signature-then-encrypt, which implies that a heavy computation is required for each step. To provide confidentiality, unforgeability, and non-repudiation security services, Signcryption was put forward by Youliang Zheng in 1997 as a scheme that integrates the public-key encryption with the digital signature into a single step to reduce the required computation processes (58% less computation time), including the reduction of the cost of communication, which leads to the acquire shorter, securer and cheaper text (70% less message expansion) rather than the ordinary combination of the public-key encryption and digital signatures [29, 41].

Schnorr Signcryption is a grouping of public-key encryption and Schnorr digital signature algorithm, which consists of five stages: Setup where the output are parameters, which are required by the signcryption schemes. KeyGenS for generating a pair of keys for the sender. KeyGenR for generating a pair of keys for the receiver. Signcrypt, which is a probabilistic algorithm. Unsigncrypt is a deterministic algorithm. Figure 2 illustrates an overview about the Schnorr Signcryption main steps based on [13, 28, 29].

1)Setup
Schnorr Signcryption parameters:
p = a large prime number, public to all
q = a large prime factor of p-1, public to all
g = an integer with order q modulo p, in [1,… , p-1], public to all
hash = a one-way hash function
KH = a keyed one-way hash function = KHk(m) = hash(k, m)
(E, D) = the algorithms which are used for encryption and decryption of a private key cipher.
Alice sends a message to Bob.
2)KeyGen sender
Alice has the pair of keys (Xa, Ya):
Xa = Alice's private key, chosen randomly from [1, .., q-1]
Ya = Alice's public key = g^{-Xa} mod p
3)KeyGen receiver
Bob has the pair of keys(Xb, Yb):
Xb = Bob's private key, chosen randomly from [1, .., q-1]
Yb = Bob's public key = g^{-Yb} mod p.
4)Signcryption
In order to signcrypt a message m to Bob, Alice has to accomplish the following operations:
Calculate
$k = hash(Yb^X) \ mod \ p$
Split k in k1 and k2 of appropriate length.
Calculate r = KHk2(m) = hash(h2, m)
Calculate s = x + (r * Xa) mod q
Calculate c = Ek1(m) = the encryption of the message m with the key k1.
Alice sends to Bob the values (r, s, c).
5)Unsigncryption
In order to unsigncrypt a message from Alice, Bob has to accomplish the following operations:
Calculate k using r, s, g, p, Ya and Xb
$k = hash((g^s \cdot Ya^r)^{-Xb} \ mod \ p)$
Split k in k1 and k2 of appropriate length.
Calculate m using the decryption algorithm m = Dk1(c).
Accept m as a valid message only if KHk2(m) = r.

Figure 2: An overview of the Schnorr signcryption algorithm.

4 Proposed solution

In this section, the main proposed solution is to explore the entire essential details. Before starting the discussion, it is important to highlight that the proposed scheme implementation environment is based on client-server architecture. The client (user device) can be any end device such as a smartphone or a computer device, which communicates with the server by using a secure communication channel such as SSL/TLS protocols.

As illustrated in Figure 3, the general working procedure starts at the client side by sending the authentication request to the server. The server calculates the required parameters and sends the generated data to the client. The client starts the local calculation using the received data from the server and sends generated parameters to the server to accomplish the final second-factor authentication process. Since the main goal is to provide a secure second authentication factor, it is assumed that users have completed the registration phase previously along with users' accounts' credentials i.e. usernames and passwords that are stored in a local server's database.

The proposed two-factor authentication scheme is based on the usage of a modified copy of the Schnorr Signcryption algorithm, which is used as a second factor in the authentication scheme. A list of terminologies that are used in the discussion part comprises:

p: a large prime number,
q: a large prime factor,
g: calculated as q mod p,
m: randomly generated message,
x: Randomly generated integer < q-1,
Xa: client's private key and Ya: client's public key (public to the server),
Xb: server's private key and Yb: server's public key (public to the client),
hash: one-way hash function, and
kh: one-way keyed hash function.

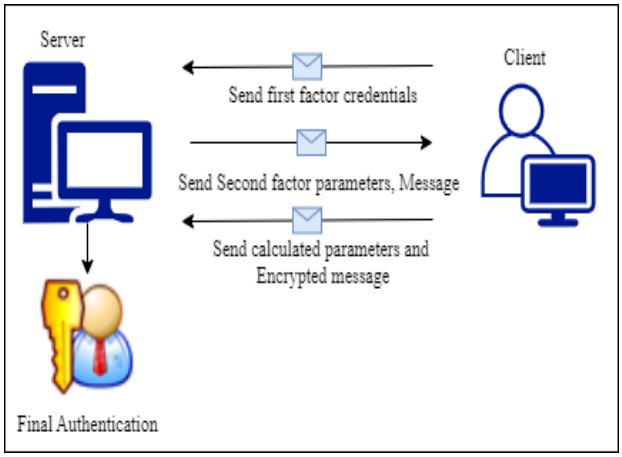

Figure 3: The general working procedure.

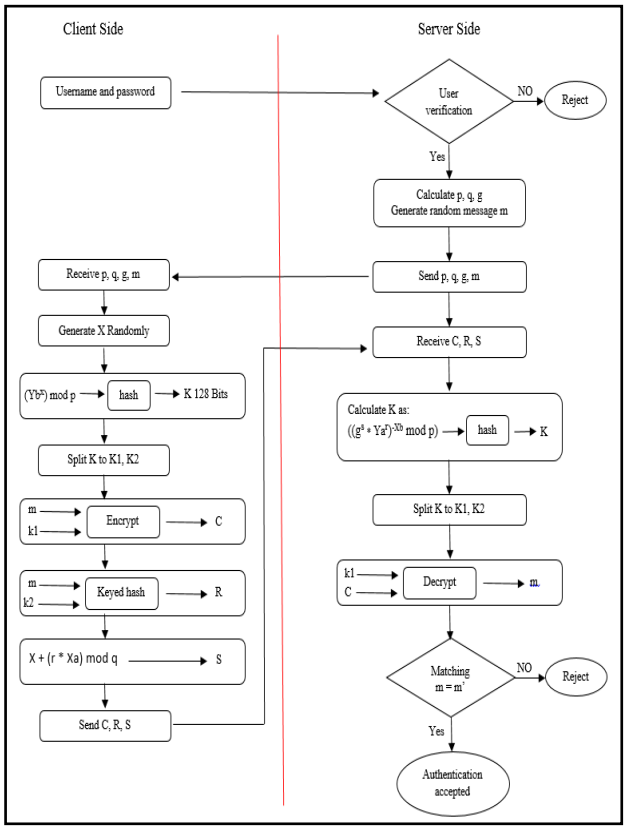

Figure 4: The proposed Two-Factor authentication scheme overall processes.

Referring to Figure 4, the proposed two-factor authentication working scenario is divided into three main operations, which start on the clients' side and complete on the server's side. This makes the final authentication decision according to the calculated results. The followings are detailed discussions about the septs that should be performed at the client's and the server's sides.

A. *Starting the authentication process*

- The proposed scheme starts with the clients' side by sending the first factor of the required parameters in this scenario, which comprise; the username and password of the user along through to the server.
- After receiving the sent identifications on the servers' side, the server starts the verification process by using the previously mentioned servers' database, and checks if the user verification process is accepted or not. The server will afterwards move along through to the next step, otherwise, the authentication request will be rejected at the servers' side.
- For each authentication request, the server will start a calculation process after the user is successfully verified, and the values of three required parameters will be generated, which include p, q, and g. Once the required parameters are generated, the server will generate a random message m and will send the entire generated data i.e. (p, q, g, and m) to the client by using a secure communication channel as mentioned previously.

B. *Clients' side calculations*

- At the clients' side, the client will receive the generated data that is sent by the server and will start a local calculation by generating the value of the x parameter, which is a random integer value that should meet the condition, which shows that x is less than the q-1 value.
- By using the server's public key and a one-way hash function (MD5), the client will calculate the value of the key parameter K, which is usually with a size of 128 bits. After the K value is calculated, the client will divide this value into two equal-sized keys, which are 64 bits for each K1 and K2 values.
- The client will use K1 and the random message that is received from the server as input to an encryption algorithm for generating the value of Ciphertext C and will use the combination of K2 and m as input to a one-way keyed hash function to generate the value of the R parameter.
- Finally, by using the client's private key, the randomly generated x, and the previously calculated r values, the client will calculate the value of the S parameter, followed by sending all calculated parameters c, r, and s back to the server to start the final authentication step at the servers' side.

C. *Servers' side calculations*

- At the servers' side, the server will receive the calculated parameters and will perform the final authentication process. By using the parameters that are sent by the client c, r, s, the client's public key, server's private key, p parameter, and the same one-way hash function, which are used by the clients' side are based on allowing the server to re-calculate the K value, which is 128 Bits size.
- The server will divide the K value into two equal-sized keys, where each key represents 64 bits K1 and K2. After that, the server will use K1 and the received Ciphertext C from the client as input to a decryption algorithm to generate the value of the original randomly generated message m, which refers to it by using the notation m'.
- Finally, the server will perform a final matching process between the randomly generated message m and the

decrypted m'. If the messages show a 100% match, the authentication process will be accepted. Otherwise, the authentication process will be rejected and access will not be granted to the user's device. The following codes show the pseudo-code of the proposed scheme, which summarizes the overall processes and the performed calculations:

Algorithm 1

Start
***Client** enters the username and password*
IF the credentials FALSE
Reject the access
Else
*The **server** calculates parameters p, q, g, and randomly generates an m message*
*Send all generated values to the **client** side*
*The **client** receives, stores, and performs the followings:*
Generate x randomly where (x< q-1)
Calculate K as hash (Yb^x) mod P
Split K to K1 and K2 with 64 bits size each
Calculate C by encrypt of the message m with the key K1.
Calculate R by using the keyed hash of the message m with the key K2.
*Calculate S as X + (r * Xa) mod q*
*Send C, R, and S to the **server***
*The **server** receives, stores, and performs the followings:*
Calculate K as hash ($(g^s * Ya^r)^{-Xb}$ mod p).
Split K to K1 and K2 with 64 bits size each.
Calculate m' by decrypt of the ciphertext c with the key K1.
*IF (m' from the **client** matches m from the **server**), then:*
An authentication is accepted
Else
The second factor is rejected
End

5 Implementation and results discussion

In this section, the implementation of the proposed scheme and the obtained results are discussed in detail. Referring to the above scheme, the proposed scheme is designed based on the modified client-server Schnorr Signcryption algorithm. The Java-based desktop application is developed and implemented at both sides of the proposed scheme i.e., server and client for testing purposes. At the server's side, the AES algorithm is used for encryption and decryption processes. While the private secure socket is used to communicate with the client and can exchange the required parameters, which are used to perform the authentication process. The MD5 hash function is used to calculate the value of the K key, which is divided into two keys each with 64 bits size. The keyed hash SHA is used with K2 in order to calculate the value of the R parameter. Moreover, when the client starts with the authentication process, a method is used every time at the server's side to generate a new random message M with a total size of 60 characters.

As previously indicated, the first factor that is used in the implementation represents the username and password, which are sent by the clients along to the server for initiating the authentication process.

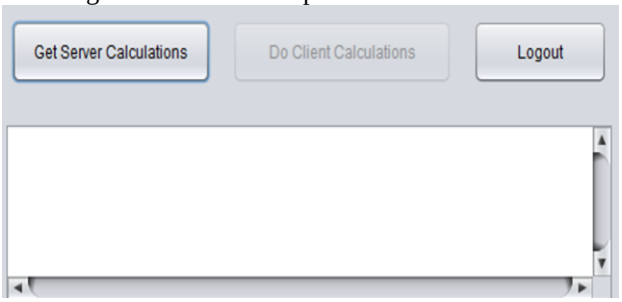

Figure 5: Clients' main interface for sending second factor authentication request.

Additionally, users are initially requested to complete the enrollment stage by the 'sign up' option on the system as new users to provide the server with the required details. After that, each user can login by using the user's account credentials. Once the first login process is successfully completed, the client can accordingly initiate the communication with the server and request the values of the server-side parameters to perform the second factor authentication processes. (see Figure 5).

With reference to Figure 6, the clients receive the calculated parameters from the server along with the random message M after the second factor request is sent by the client to the server to the server. The client performs all the previously mentioned calculations and sends the values of the parameters along with encrypted message ciphertext C to the server to perform the final authentication process, and to perform the final decision according to the last matching process that is mentioned in the proposed scheme. Moreover, during the testing process and after the random message M is sent by the server, the content of message M was modified at the client's side for integrity checking. Further, the ciphertext C is performed based on the manipulated message M and is sent through to the server. At the server's side, the authentication process of the second factor fails due to the non-matching result between the original generated message M and the decrypted modified message. Consequently, this failure leads to a final decision from the server by rejecting the authentication process where no access permission is granted to the client. Figure 7, illustrates the message integrity checking scenario.

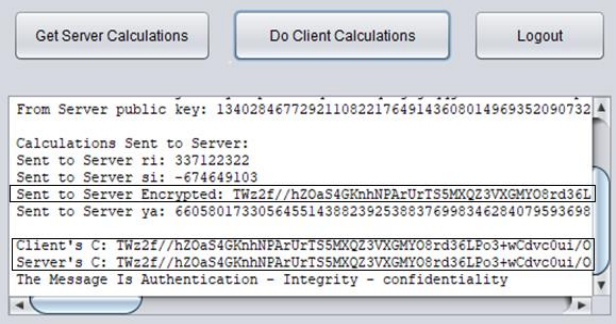

Figure 6: Client-server second factor authentication calculation with matching results.

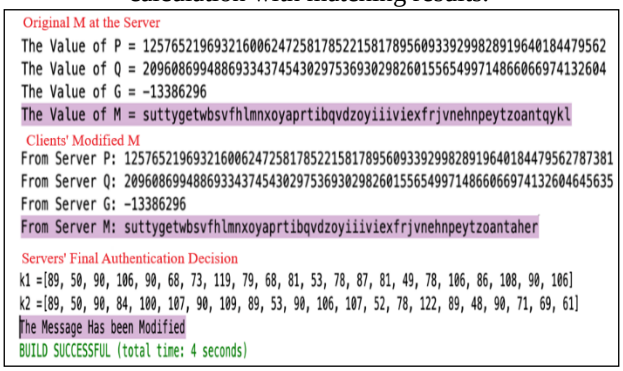

Figure 7: Modified M checking scenario.

The signcryption scheme's main aim is to provide a strong security solution at better performance and lower overhead for achieving message confidentiality, unforgeability, and non-repudiation services. The proposed solution takes into consideration these security aspects and a security analysis of each one is presented to make sure that all security aspects are achieved in the proposed solution.

The proposed solution achieves confidentiality by utilizing a public key encryption system so that the server and client will not require to exchange the secret key since they both use the same public key (Ya, Yb), and their private key (Yb, Xb) is kept secret and only used for encryption and decryption operations in authentication messages between them. Because each has a unique private key, the server and client did not exchange a secret key. However, confidentiality main goal is to make sure that no information can be obtained from the signcrypted message by the adversary i.e. attacker, which is in our scenario ciphertext C that is sent from the client to the server for final authentication. To make sure that confidentiality is achieved in the proposed solution, let us assume that the attacker can access the ciphertext C, and the other required parameters used at the server to calculate the K key value i.e (r, s, Ya). To unsigncrypt the message and retrieve its information the attacker needs to gain an access to Xb which is the servers' private key that is used as one of the parameters required in the process of calculating the K key value which will be divided into K1 and K2 (see Figure 4), and it is known only by the server as it is a private key and not published. According to that, the attacker is not able to unsigncrypt the ciphertext C under any situation unless the servers' private key is revealed with all other required parameters, due to the requirements of calculating the K key, splitting K into K1 and K2, using K1 for unsigncryption, which leads to deduce that the confidentiality is achieved in the proposed two-factor solution.

Unforgeability aims to prevent the ability to create forge valid signatures by the adversary. Referring to the used signcryption algorithm in Section 3, the sender calculates the value of the r parameter by using K2 and a one-way keyed hash function, which requires the use of the private key as previously discussed. Additionally, if the adversary can access the r parameter, and since it is using a one-way hash there is no possibility of retrieving any information about the value of K2 and the original message M which are used during the generation of the r value. Moreover, the receiver will accept and validate the decrypted M under the condition that the calculated r at the receiver side from the decrypted M is the same as the r value calculated at the sender side, which needs the private key to perform the calculation. Additionally, if the adversary can obtain the parameters p, q, g, and m from the server by attacking the client and server and utilizing the same hashing, encryption, and key hashing algorithms, then it generates the correct values for the parameters K, C, and R, but the parameter S will be incorrect because it depends on the authenticator client's private key. Therefore, when the server receives C, R, and S from the adversary, an incorrect value of the K parameter will be generated due to the incorrect value of S, which will then cause the decryption process to produce a different message than the original message, and the server will decline to authenticate this fake client (adversary). Thus, unforgeability is met under the discussed two scenarios from the proposed solution.

In terms of non-repudiation, only the intended receiver of the signcryption can verify the authenticity of the sender. Moreover, in this proposed solution, when the server sends the parameters p, q, g, and m to the client, the client should reply with the parameters C, R, and S back to the server. This is considered evidence of the server parameters that have been delivered to the client. Additionally, when the original signature on the server side is matched with the signature generated based on the parameters received from the client and the client is authenticated, this is considered proof of the identity of the client and therefore he/she cannot repudiate his signature in this case, and therefore the proposed scheme has achieved non-repudiation. However, since the final authentication is approved based on the matching process conducted at the server side to ensure that the received data was not intercepted under any circumstances i.e. integrity is achieved, all parties can be confident that data is secure (non-repudiation). Therefore, there is no repudiation of this proposed solution.

Referring to the above-mentioned related research listed in Table 1, different two-factor authentication solutions are proposed for a predetermined problem. Most solutions are implemented on a predefined domain and based on specific techniques. For instance, the usage of wearable devices, biometrics, USB token, etc. Blockchain decentralized setting is proposed as the second factor in [6] to safeguard users' transactions in the context of online

financial transactions, and for remote data access protection in the healthcare domain. A hash-chain and secure forward solution is proposed by [14], and for the security threat of stolen verifier attacks, the graphical password is used in [26, 36]. In the context of IoT, several solutions are proposed to improve the process of OTP generation process by using QOTP and biometrics-based approach proposed by [30], or by using the T/Key OTP algorithm and Merkle tree [39]. IoT devices protection is proposed by using the PUF [4, 15]. Multi-servers- two-factor authentication based on the usage of the ECC for IoT solution is proposed by [38]. Other solutions are proposed to improve convenient user authentication on smartphones [42]. Bitcoin wallet security protocol based on (ECDSA) is proposed by [19]. For example, the PCS approach is improved to meet all the authentication requirements [21], and decrease users' involvement in mobile two-factor authentication [7].

In comparison, this research successfully provides a two-factor authentication based on the usage of a modified copy of the Schnorr Signcryption algorithm, which can be used as a second factor with several characteristics. One of those characteristics is the implementation of any client-server architecture. Moreover, the provision of data, confidentiality, and integrity protection from different unauthorized access attempts. The provision of confidentiality, unforgeability, and non-repudiation security services. Additionally, no restrictions in the context or user device that is used in the authentication process. Furthermore, providing a solution with a lower computational process and lower communication cost.

In conclusion, the target aim of the proposed scheme is successfully achieved by providing a secure second authentication factor with significantly lower overheads and computational processes (58% less computation time and 70% less message size) that can protect the local resources from any unauthorized requested access [40,41]. Additionally, this factor aims to provide data confidentiality and integrity by implementing a modified Schnorr Signcryption algorithm as a second authentication factor. The Schnorr signcryption algorithm represents a public key cryptography algorithm, which is proposed based on the usage of the asymmetric key system. The sender and receiver will not share the used single key, but each side will use its secret key. Due to this asymmetry usage of secret keys, signcryption or the unsigncryption cannot be performed by an adversary due to the demand of the secret key and the calculated parameters. Accordingly, it can be inferred that the Schnorr signcryption scheme can protect the confidentiality and integrity features and can be implemented as a second authentication factor.

6 Conclusion

In this paper, the Two-factor authentication scheme is based on the usage of the modified Schnorr Signcryption algorithm that is proposed to provide a secure, protected and usable second authentication factor. This scheme can be used with any single factor such as a smart card or conventional username and password and can be implemented on any users' devices with no restrictions or special requirements within the client-server environment. Furthermore, this scheme can provide an adequate level of data confidentiality and integrity, and can provide protection from several unauthorized access attempts. The Java-based application is developed for testing intentions and is implemented on both sides of the scheme i.e., the client's and server's sides. The first authentication factor that is used comprises the username and password. It can be proven from the obtained results that the proposed signcryption-based scheme can successfully achieve the main aim where the authentication process is performed with an acceptable performance level and with a lower overhead.

Acknowledgement Authors would like to thank Arab Open University, Saudi Arabia, for supporting this study.

References

[1] S. Acharya, A. Polawar, and P. Pawar, "Two factor authentication using smartphone generated one time password," *IOSR Journal of Computer Engineering (IOSR-JCE)*, vol. 11, no. 2, pp. 85-90, 2013.

[2] G. Ali, M. Ally Dida, and A. Elikana Sam, "Two-factor authentication scheme for mobile money: A review of threat models and countermeasures," *Future Internet*, vol. 12, no. 10, p. 160, 2020.

[3] F. Aloul, S. Zahidi, and W. El-Hajj, "Two factor authentication using mobile phones," in *2009 IEEE/ACS international conference on computer systems and applications*, 2009: IEEE, pp. 641-644.

[4] M. N. Aman, M. H. Basheer, and B. Sikdar, "Two-factor authentication for IoT with location information," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3335-3351, 2018.

[5] M. F. Ayub, S. Shamshad, K. Mahmood, S. H. Islam, R. M. Parizi, and K.-K. R. Choo, "A provably secure two-factor authentication scheme for USB storage devices," *IEEE Transactions on Consumer Electronics*, vol. 66, no. 4, pp. 396-405, 2020.

[6] F. Breuer, V. Goyal, and G. Malavolta, "Cryptocurrencies with security policies and two-factor authentication," in *2021 IEEE European Symposium on Security and Privacy (EuroS&P)*, 2021: IEEE, pp. 140-158.

[7] Y. Cao, Q. Zhang, F. Li, S. Yang, and Y. Wang, "PPGPass: Nonintrusive and secure mobile two-factor authentication via wearables," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*, 2020: IEEE, pp. 1917-1926.

[8] I.-P. Chang, T.-F. Lee, T.-H. Lin, and C.-M. Liu, "Enhanced two-factor authentication and key agreement using dynamic identities in wireless sensor networks," *Sensors*, vol. 15, no. 12, pp. 29841-29854, 2015.

[9] B. Consulting. "GLOBAL TWO FACTOR AUTHENTICATION MARKET." https://www.blueweaveconsulting.com/report/global-two-factor-authentication-market (accessed.

[10] S. M. Danish, M. Lestas, W. Asif, H. K. Qureshi, and M. Rajarajan, "A lightweight blockchain based two factor authentication mechanism for LoRaWAN join procedure," in *2019 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2019: IEEE, pp. 1-6.

[11] S. Das, B. Wang, Z. Tingle, and L. Camp, "Evaluating user perception of multi-factor authentication: A systematic review. arXiv 2019," *arXiv preprint arXiv:1908.05901*, 2019.

[12] A. Dmitrienko, C. Liebchen, C. Rossow, and A.-R. Sadeghi, "On the (in) security of mobile two-factor authentication," in *International Conference on Financial Cryptography and Data Security*, 2014: Springer, pp. 365-383.

[13] A. Elshobaky, M. Rasslan, and S. Guirguis, "Implementation of schnorr signcryption algorithm on dsp," *International Journal of Security and Its Applications*, vol. 9, no. 11, pp. 217-230, 2015.

[14] M. Fotouhi, M. Bayat, A. K. Das, H. A. N. Far, S. M. Pournaghi, and M.-A. Doostari, "A lightweight and secure two-factor authentication scheme for wireless body area networks in health-care IoT," *Computer Networks*, vol. 177, p. 107333, 2020.

[15] P. Gope and B. Sikdar, "Lightweight and privacy-preserving two-factor authentication scheme for IoT devices," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 580-589, 2018.

[16] S. Ibrokhimov, K. L. Hui, A. A. Al-Absi, and M. Sain, "Multi-factor authentication in cyber physical system: A state of art survey," in *2019 21st international conference on advanced communication technology (ICACT)*, 2019 2019: IEEE, pp. 279-284.

[17] N. Karapanos, C. Marforio, C. Soriente, and S. Capkun, "{Sound-Proof}: Usable {Two-Factor} Authentication Based on Ambient Sound," in *24th USENIX security symposium (USENIX security 15)*, 2015, pp. 483-498.

[18] A. Mail and D. Box, "Two factor authentication," 2017.

[19] C. Mann and D. Loebenberger, "Two-factor authentication for the Bitcoin protocol," *International Journal of Information Security*, vol. 16, no. 2, pp. 213-226, 2017.

[20] M. Mukhtar, R. Sulaiman, A. Zin, and M. Alshare, "A USER PROTECTION MODEL FOR THE TRUSTED COMPUTING ENVIRONMENT," *Journal of Computer Science*, vol. 10, pp. 1692-1702, 09/01 2014, doi: 10.3844/jcssp.2014.1692.1702.

[21] B. P. Nance and A. S. Bengston, "An Integrated Two-Factor Authentication Solution Using Pulse Connect Secure and Apache HTTP Server," Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States), 2020.

[22] S. Nivetha, "Design of a two-factor authentication ticketing system for transit applications," in *2016 IEEE Region 10 Conference (TENCON)*, 2016: IEEE, pp. 2496-2502.

[23] A. Ometov, S. Bezzateev, N. Mäkitalo, S. Andreev, T. Mikkonen, and Y. Koucheryavy, "Multi-factor authentication: A survey," *Cryptography*, vol. 2, no. 1, p. 1, 2018.

[24] T. Petsas, G. Tsirantonakis, E. Athanasopoulos, and S. Ioannidis, "Two-factor authentication: is the world ready? Quantifying 2FA adoption," in *Proceedings of the eighth european workshop on system security*, 2015, pp. 1-7.

[25] Q. Pu, "An improved two-factor authentication protocol," in *2010 Second International Conference on Multimedia and Information Technology*, 2010, vol. 2: IEEE, pp. 223-226.

[26] K. M. Quadry, A. Govardhan, and M. Misbahuddin, "Design, Analysis, and Implementation of a Two-factor Authentication Scheme using Graphical Password," *International Journal of Computer Network & Information Security*, vol. 13, no. 3, 2021.

[27] K. Reese, T. Smith, J. Dutson, J. Armknecht, J. Cameron, and K. Seamons, "A Usability Study of Five {Two-Factor} Authentication Methods," in *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*, 2019, pp. 357-370.

[28] L. Savu, "Schnorr Digital Signature in Signcryption Scheme," 2012.

[29] L. Savu, "Signcryption scheme based on schnorr digital signature," *arXiv preprint arXiv:1202.1663*, 2012.

[30] M. K. Sharma and M. J. Nene, "Two-factor authentication using biometric based quantum operations," *Security and Privacy*, vol. 3, no. 3, p. e102, 2020.

[31] F. Sherali and S. Falah, "An Efficient Two Factor User Authentication and Key Exchange Protocol for Telecare Medical Information System," *Computer Science*, vol. 15, no. 4, pp. 1015-1027, 2020.

[32] J. Sun, R. Zhang, J. Zhang, and Y. Zhang, "Touchin: Sightless two-factor authentication on multi-touch mobile devices," in *2014 IEEE conference on communications and network security*, 2014: IEEE, pp. 436-444.

[33] D. Wang, N. Wang, P. Wang, and S. Qing, "Preserving privacy for free: Efficient and provably secure two-factor authentication scheme with user anonymity," *Information Sciences*, vol. 321, pp. 162-178, 2015.

[34] D. Wang and P. Wang, "Two birds with one stone: Two-factor authentication with security beyond conventional bound," *IEEE transactions on dependable and secure computing*, vol. 15, no. 4, pp. 708-722, 2016.

[35] F. Wu, X. Li, L. Xu, P. Vijayakumar, and N. Kumar, "A novel three-factor authentication protocol for wireless sensor networks with IoT notion," *IEEE Systems Journal*, vol. 15, no. 1, pp. 1120-1129, 2020.

[36] F. Wu, L. Xu, S. Kumari, and X. Li, "An improved and anonymous two-factor authentication protocol for health-care applications with wireless medical sensor networks," *Multimedia Systems*, vol. 23, no. 2, pp. 195-205, 2017.

[37] Q. Xie, N. Dong, D. S. Wong, and B. Hu, "Cryptanalysis and security enhancement of a robust two-factor authentication and key agreement protocol," *International Journal of Communication Systems*, vol. 29, no. 3, pp. 478-487, 2016.

[38] G. Xu *et al.*, "A multi-server two-factor authentication scheme with un-traceability using elliptic curve cryptography," *Sensors*, vol. 18, no. 7, p. 2394, 2018.

[39] X. Yin, J. He, Y. Guo, D. Han, K.-C. Li, and A. Castiglione, "An efficient two-factor authentication scheme based on the Merkle tree," *Sensors*, vol. 20, no. 20, p. 5735, 2020.

[40] J. Zhang, X. Tan, X. Wang, A. Yan, and Z. Qin, "T2FA: Transparent two-factor authentication," *IEEE Access*, vol. 6, pp. 32677-32686, 2018.

[41] Y. Zheng, "Signcryption and Its Applications in E cient Public Key Solutions."

[42] B. Zhou, J. Lohokare, R. Gao, and F. Ye, "Echoprint: Two-factor authentication using acoustics and vision on smartphones," in *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, 2018, pp. 321-336.