Chinese Journal of Electronics
vol. 33, no. 1, pp. 43–57, January 2024
https://doi.org/10.23919/cje.2022.00.339

RESEARCH ARTICLE

A Secure Mutual Authentication Protocol Based on Visual Cryptography Technique for IoT-Cloud

Brou Bernard Ehui[1], Chen CHEN[1,2], Shirui WANG[3,4], Hua GUO[1,2], and Jianwei LIU[1]

1. *School of Cyber Sciences and Technology, Beihang University, Beijing 100191, China*
2. *State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China*
3. *Sino-French Engineering School, Beihang University, Beijing 100191, China*
4. *Beihang Hangzhou Innovation Institute Yuhang, Hangzhou 310051, China*

Corresponding author: Hua GUO, Email: hguo@buaa.edu.cn
Manuscript Received October 4, 2022; Accepted April 4, 2023

Abstract — Because of the increasing number of threats in the IoT cloud, an advanced security mechanism is needed to guard data against hacking or attacks. A user authentication mechanism is also required to authenticate the user accessing the cloud services. The conventional cryptographic algorithms used to provide security mechanisms in cloud networks are often vulnerable to various cyber-attacks and inefficient against new attacks. Therefore, developing new solutions based on different mechanisms from traditional cryptography methods is required to protect data and users' privacy from attacks. Different from the conventional cryptography method, we suggest a secure mutual authentication protocol based on the visual cryptography technique in this paper. We use visual cryptography to encrypt and decrypt the secret images. The mutual authentication is based on two secret images and tickets. The user requests the ticket from the authentication server (AS) to obtain the permission for accessing the cloud services. Three shared secret keys are used for encrypting and decrypting the authentication process. We analyze the protocol using the Barrows-Abadi-Needham (BAN)-logic method and the results show that the protocol is robust and can protect the user against various attacks. Also, it can provide a secure mutual authentication mechanism.

Keywords — Internet of things, Visual cryptography, Security, Mutual authentication, IoT-cloud.
Citation — Brou Bernard Ehui, Chen CHEN, Shirui WANG, *et al.*, "A Secure Mutual Authentication Protocol Based on Visual Cryptography Technique for IoT-Cloud," *Chinese Journal of Electronics*, vol. 33, no. 1, pp. 43–57, 2024. doi: 10.23919/cje.2022.00.339.

I. Introduction

The Internet of things (IoT) has become one of the most promising technologies that aim to increase productivity, reduce cost and improve human quality of life. It is a global term used for describing scenarios in which Internet connectivity and computing capability extend to a variety of objects, devices, sensors, and every item to generate, exchange and consume data with the minimum human intervention [1]. At present, the IoT is transforming how businesses are run, how lives are lived, and how society functions in general. Now, many industry sectors including healthcare, automotive, manufacturing, consumer electronics, home, etc., have found the potential for incorporating IoT technology into their products, services, and operations. With the concept of IoT, everyday objects are transformed into smart objects by using digital entities such as sensors, radio-frequency identification (RFID), and localization technologies. These smart objects are able to interpret and interact with each other without human intervention. The objects embedded with the sensor can monitor, sense, and collect data from the environment and human social life. The main function of these embedded objects is to collect data from the environment and transfer such data to the cloud through the network, where intelligent analytics is applied to extract meaningful information which is used to predict, optimize, and improve the business and the public infrastructure operations. These smart objects can generate a vast amount of data for individuals and organizations. Ac-

Associate Editor: Assoc. Prof. Ju REN, Tsinghua University.

cording to reference [2], by 2025, 180 billion TB of data will be generated every year. This generated data must be available and accessible anytime and anywhere on the network.

With the development of sensor technology, wireless communication technologies, and the Internet of things, cloud computing is becoming a vast growing technology in various sectors. Many users are connecting to the cloud using lightweight resources constrained devices. In addition, cloud computing platform provides various methods of data storage, data backup and recovery, data processing, and a variety of services for data access. Although cloud computing can offer a better solution to handle a large number of concurrent processes that exceed the computing capacity of a traditional computer or workstation, there are still various security challenges to be addressed. These challenges include data privacy, data security, cloud computing platform security, cloud computing services security, and mutual authentication. Authors in references [3]–[5] have described various security threats in the cloud environment. These threats include data leakage, data breaches, data loss, DDoS attacks, malicious insiders, traffic hijacking, and misuse of cloud facilities. Many others vulnerabilities and threats exist. For example, data traffic can be intercepted and altered by a hacker during the data transmission process. Unauthorized access to the cloud will result in the leakage and misuse of personal information. In addition, the lack of a secure authentication process can increase the risk of stealing data from the cloud platform. These fundamental challenges need to be considered critical issues. Therefore, an authentication protocol is required to manage and secure the identity of users accessing the cloud platform. Moreover, a sophisticated security mechanism is needed to guarantee the confidentiality, integrity, and authenticity of data traveling through the network and safeguard the proper function of the cloud platform. Serious efforts to find strict security solutions need to be prioritized.

Many researchers are focusing their work on solving security challenges in IoT networks. These challenges include interoperability, data privacy, access control, and lightweight protocol which aims to provide a solution for resource-constrained devices by reducing the key size, cycle rate, throughput rate, power consumption, and area [6]–[8]. They are also trying to improve the traditional cryptography schemes or to develop a new security protocol to protect data transmission between the cloud platform and users [9], [10].

However, most of these methods are often vulnerable to various cyber-attacks like man-in-the-middle and others. They are also inefficient against new cyber-attacks. Therefore, developing new solutions based on different mechanisms from traditional cryptography algorithms is required to protect data and users' privacy from attacks. In this paper, we suggest a distinctive method different from the conventional cryptography scheme to provide a secure mutual authentication mechanism in the IoT-cloud platform.

The proposed protocol is based on the visual cryptography technique. Visual cryptography is a technique of splitting a secret image into several shares such that the original image is revealed by stacking sufficient shares of the secret image. It was proposed in 1994 by Naor and Shamir [11] to encrypt secret images such that the original image can be retrieved using a human visual system. It does not need any cryptography computation. The protocol is also based on a hash function, mask secret, and symmetric encryption algorithm. The hash function is used to provide data integrity. Data confidentiality is ensured by encrypting and decrypting the message using the symmetric key shared between the user and the servers (authentication server, cloud server). The mask secret is generated by taking the XOR of the user password and a random number generated by the user. It is used to encrypt the secret image. Before establishing a secure communication session between the user and the cloud server, a mutual authentication process is required. The user requests the ticket from the authentication server (AS), to obtain the permission for accessing the cloud services. The protocol guarantees that only authenticated and authorized users can access the cloud. The proposed scheme enhances security with the use of the secret image as an authenticator. The security analysis shows that the protocol can provide a high-security mechanism to guarantee data integrity, confidentiality, and authenticity. It is also robust and efficient to resist many attacks.

The rest of the paper is organized as follows: the next section presents related work. Section III describes the network architecture and security requirements. Section IV discusses the proposed protocol in detail. Section V discusses the security analysis of the proposed protocol and Section VI concludes.

II. Related Work

1. Visual cryptography

The visual cryptography mechanism was first introduced in 1994 by Naor and Shamir [11]. In their proposed scheme they demonstrated a new type of cryptographic which can decode concealed images without any computation. This proposed mechanism was only applied to black and white images where n transparencies of the original image are generated. Then, by stacking k of these n transparencies, the original image can be revealed. Many studies have been carried out to improve visual cryptography systems [12]–[15]. Moreover, an enhanced visual cryptography technique and survey are presented in [16], [17]. In these papers, the black and color images are divided into different shares. The shares are then encrypted with traditional cryptography algorithms. Their proposed technique provides an efficient way to secure digital images. All these proposed schemes

can provide an efficient solution to guarantee image security. Because of its high-level security mechanism, it is crucial to consider visual cryptography technique in designing a secure mutual authentication protocol for ensuring confidentiality, data integrity, and authenticity in IoT-cloud environment.

2. Mutual authentication protocols in cloud computing

As traditional system cannot handle the massive volumes of data generated by IoT devices, the cloud computing technology is a better solution to handle such super large amount of data called big data. It offers various services to make human life easier and simple. However, security remains a major challenge in this field. To address security challenges in cloud computing, many researchers have analyzed security issues on various areas of cloud computing [18]–[20]. Others have proposed authentication schemes based on password, biometric, steganography, and traditional symmetric and asymmetric methods to ensure data security in cloud environment [21]–[23]. The drawback of these proposed schemes is that they are based on traditional authentication mechanism such as username, password, etc., to gain access to the cloud services. However, users' credentials (username, password, ID, etc.) can easily be compromised, leading to severe vulnerabilities of cloud services. This can also help the attacker to gain access to the cloud services and perform attacks. Arora *et al* [24] proposed authentication protocol based on multifactor authentication mechanism for ensuring data security in cloud ecosystem. They used hybrid cryptography system and one time password (OTP) to design the protocol. OTP is sent to the user by email for login process. Once successfully login, users are able to send or retrieve data from cloud platform. Data can be encrypted and stored using RSA and AES algorithms. Their protocol was robust and resist various attacks. However, in 2019, Islam *et al* [23] analyzed the reference [24] and found that the protocol is vulnerable to impersonate attack and propose an improved scheme to address this drawback. They declared in their paper that if attacker obtains user's credentials he/she can gain access to the cloud, but he/she will not be able to access data or modify that data. This means that their improved protocol can only ensure data security within cloud platform, but it is vulnerable to various attacks such as impersonate attacks, replay attack, and brute force attack.

Moreover, many works have proved the weakness of these traditional methods. In 2020, Fan *et al* [25] presented a lightweight cloud-based authentication protocol for IoT networks. They show that their protocol was robust and secure against various attacks. However, in 2021, Adeli *et al* [26] pointed out that this protocol is vulnerable to disclosure attacks and proposed an improved protocol called χperbp. Furthermore, Nimmy *et al* [27] used secret sharing and steganography to design a novel mutual authentication protocol for cloud computing. They claimed that their proposed protocol was efficient and secure against various known attacks. However, authors in [28] show that the proposed scheme suffers from offline password guessing attack and is also vulnerable to denial of service attack. As a part of our contribution, we propose a novel secure mutual authentication method different from the traditional method. We consider the visual cryptography technique to design the protocol.

III. Network Architecture and Security Requirements

1. Network architecture

The network architecture we consider in this work is shown in Figure 1. As we can see in this figure, the network contains three main parts: IoT environment, IoT-cloud platform, and user platform.

IoT environment IoT environment is where sensors are used to collect data from many sources across the globe. These sensors can be found in any area including, agriculture, healthcare, automotive, home, smart city, smart transport, consumer electronics, aeronautic, metrological, space, and much more. Sensors represent the eyes, ears, nose, and fingers of IoT. Today, there are thousands of different types of sensors. They include, temperature sensor, motion, moisture, tactile pressure, gravity, chemical, sound, detecting light, and much more. When they are embedded in physical objects, they become smart and they can sense, monitor, and collect data from the environment. In addition, sensors can detect and measure the concentration of pollution or toxic substances in the atmosphere without human intervention. They can also decide by themselves to increase productivity, reduce cost and improve human quality of life. These smart objects generate a vast amount of data. According to [29], by 2025, the data volume created by IoT connections will reach 79.4 Zettabytes.

IoT-cloud platform Cloud is where data from the IoT environment is stored, processed, and analyzed. After collecting data from the environment, sensors send such data securely to the cloud platform where analytical techniques are used to extract useful information that can improve business and human quality of life. Cloud offers many advantages to IoT technology. It allows processing, analyzing, storing, and accessing data in the same location.

User platform To access data stored in the cloud platform, the user needs to login to the cloud by using different kinds of devices like mobile phones, tablets, computers, and any other devices that can connect to the Internet. Many users can access data from the cloud including doctors, patients, families, engineers, etc.

In this work, we assume that a secure data transmission mechanism is established between the IoT environment and the IoT-cloud platform. All the data traffic between these two parts are encrypted with an advanced cryptography algorithm. The proposed protocol is used

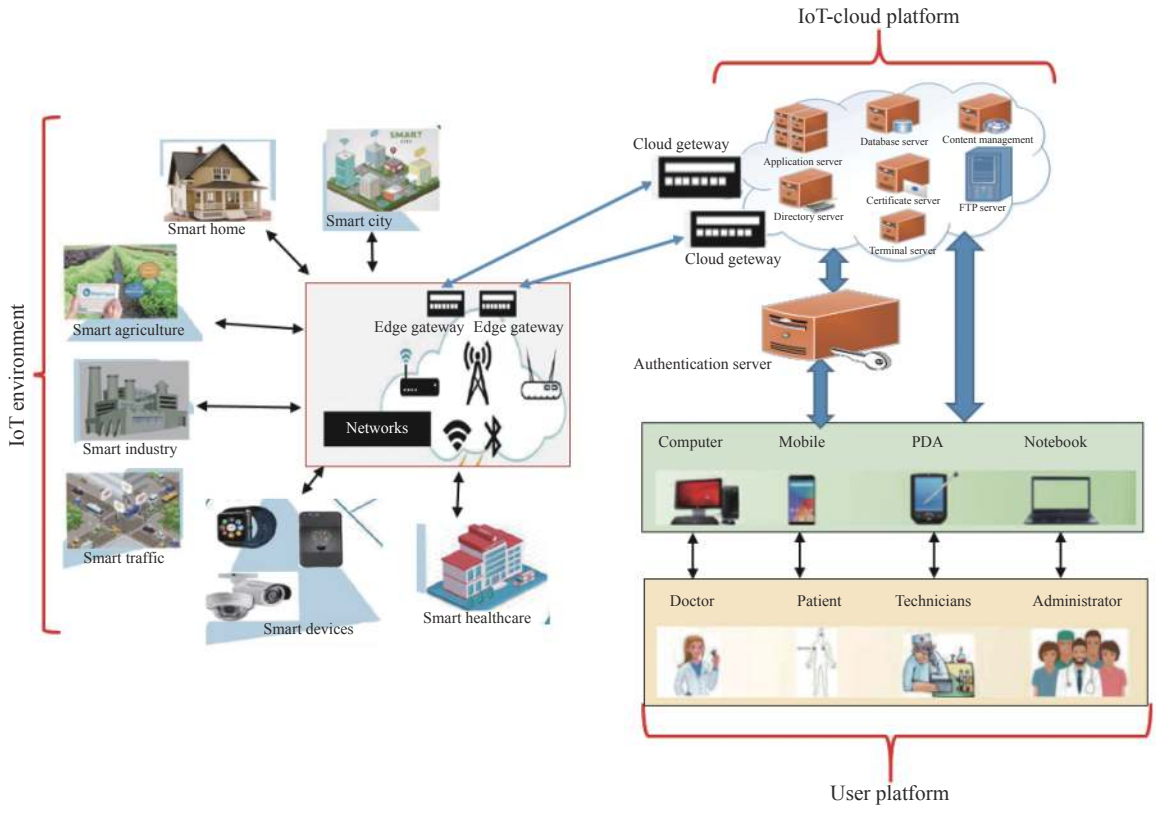

Figure 1 Network Architecture.

between the user platform and the IoT-cloud platform. Before establishing secure communication between the user and the cloud server, an authentication process is required between the user and the authentication server (AS). The user sends an authentication request to the AS. The AS authenticates the user and returns a ticket to the user. The user uses this ticket to establish secure communication with the cloud server.

2. Security requirements

In this section, we present the different attack types that attackers can launch in our proposed network. As we have already mentioned in the previous section, the proposed protocol is applied between the user platform and the cloud. There are many weaknesses that attackers can explore to get unauthorized access to the cloud. Various attacks types can be performed by the attacker during data transmission between these two parts. They include:

1) Impersonate attack: Attacker can try to masquerade as a valid user by using user's information such as username, user ID, and password.

2) Dictionary attack: The attacker can use attack tools to discover the user's password. Today, many tools can be used to perform a dictionary attack.

3) Unauthorized database access: Before accessing the cloud platform, the user needs to create an account by providing his username, password, user ID, and much more. All these credentials are stored in a database for the future authentication process. This database can be compromised by the attacker, leading to unauthorized access to the cloud services.

4) Replay attack: The attacker can eavesdrop on the mutual authentication message between the user and the cloud server and replay this captured message later to establish an authentication session.

5) Man-in-the-middle attack: The attacker will secretly take over confidential communication between the user and the cloud server. He can intercept, read, modify and replace the communication traffic.

6) Brute force attack: The attacker can try to guess the user's password to decrypt the message between the user and the cloud server.

In any of these cases, the attacker will be able to gain access to services and user's personal data. Hence, an unquestionable authentication protocol needs to be designed to protect the system against these security issues. This protocol may be able to provide data integrity, confidentiality, and authenticity. The proposed protocol is designed to deal with these security issues.

3. Visual cryptography

Visual cryptography is a secret sharing scheme that consists of breaking an original image into image shares. Overlaying the image shares on top of one other reveal the secret image. It is an encryption technique to hide information in an image in such a way that it can be decrypted by human vision [30]. It does not require any computation. Visual cryptography was first introduced in 1994 by Naor and Shamir [11]. In their work, they

demonstrated that black and white secret image can be encoded into n shares. These shares are individually printed on transparencies and distributed to n participants. The participants can decrypt the secret image by stacking their transparencies together. Having $n-1$ shares cannot reveal the secret image. At present, many types of visual cryptography exist. The most commonly used are k out of k visual cryptography, k out of n visual cryptography, general access structure, and halftone visual cryptography. In the k out of k visual cryptography technique, the original image is divided into k shares (mostly $k = 2$). To reveal the secret image, all the k shares are needed. Having $k-1$ shares will not be able to reveal any secret information about the secret image. If one share is lost in the k out of k technique the secret image cannot be revealed. To deal with this problem, Naor and Shamir have introduced the k out of n visual cryptography technique. In the (k, n) technique, the original image is divided into n shares. By stacking k of n shares (k is between 2 to n) together, the secret image is revealed. If less than k shares are stacked together the secret image cannot be revealed. To enhance the security in the (k, n) technique authors in [31] have developed a new technique called general access structure. In this proposed scheme, the original image is split into n shares, then the n shares are divided into two subsets namely qualified and a forbidden subsets of shares. Only k shares from the qualified subset of shares can reveal the secret image. Less than k shares from a qualified subset cannot decode the secret information. However, the forbidden subset cannot reveal the secret image even if k or more shares are stacked together. In halftone visual cryptography [32], a secret binary pixel called a halftone cell is encoded into an array of sub-pixels. Halftone share can be obtained by using halftone cells of an appropriate size. This technique provides good contrast and also increases the quality of the share.

IV. Proposed Protocol

In this section, we present the different component of the proposed protocol. The protocol is an authentication protocol based on the visual cryptography technique. Visual cryptography is a technique of splitting a secret image into several shares such that stacking a sufficient number of shares of the secret image reveals the original image with simple computation for decryption [33]. The proposed scheme is used for ensuring authentication mechanisms in IoT-cloud platform. The system model considered in this work contains three main components: user platform, IoT-cloud platform, and authentication server (AS). The user platform is any device equipped with an application that can request service from the IoT cloud server. The IoT cloud platform is where data is stored. It provides specific services for users and shares authentication credentials with the AS to authenticate the user. AS holds all the authentication credentials of the user and the server. It represents the central point of the network system and plays the role of a trusted third party (TTP). In addition, AS distributes a set of credentials to users and servers for mutual authentication. The proposed scheme enhances security with the use of the secret image as an authenticator and provides a security mechanism to guarantee data integrity, confidentiality, and authenticity. The protocol is based on four security mechanisms including visual cryptography, mask secret, symmetric encryption, and hash function. Visual cryptography is used to split the original image into different shares. Then, the mask secret is applied to encrypt the shares. The mask secret has the same size as the original image and it is not shared through the network. Hence, the attacker is prohibited from revealing the secret image without having the mask secret. Furthermore, the user's password used to generate the mask secret is never shared through the network. Hash function is used to provide data integrity during the authentication process and normal communication process. The hash of the whole message is calculated before sending it to the server. When the receiver receives the message from the user, it calculates the hash of the received message. If the received hash matches the calculated hash, then the authentication process continues. Otherwise, it fails. The symmetric encryption algorithm is used to ensure data confidentiality by encrypting and decrypting the message with the shared secret keys.

The proposed protocol uses three secret keys (SK_{UA}, SK_{AServ}, SK_{UServ}. The first two keys (SK_{UA}, SK_{AServ}) are generated by the AS and distributed to the user and the cloud server before the authentication process. SK_{UA} is used between the user and the AS and SK_{AServ} between the AS and the server. SK_{UA} is not revealed to the server and vice versa. The third key (SK_{AServ}) is calculated by both the user and the server using the XOR of the hash of user's password ($h(U_{pw})$), R^i, and the hash of the XORed random numbers ($h(N_1 \oplus N_2)$). It can be represented as follows:

$$SK_{UServ} = h(U_{pw}) \oplus R^i \oplus h(N_1 \oplus N_2)$$

where $R^i = P(C^i)$ (using physical unclonable function (PUF)), C^i: the challenge for i-th iteration, and R^i: the response of PUF for C^i.

According to [7], PUF is defined as a function that maps a set of challenges to a set of responses based on intractably complex physical system. It is characterized by a challenge-response pair (CRP). It can be represented by the following equation:

$$R = P(C)$$

where C is the challenge, R is the response, and P is the function. For each input challenge (C), the PUF generates a unique output that cannot be characterized using an invasive or a side-channel attack. The set of notations used to describe the proposed protocol are given in Table 1.

Table 1 Notations

Notation	Description
U, S, AS	User, server, and authentication server identifier.
U_{id}, AS_{id}, S_{id}	User, authentication server, and server ID
$\oplus$	XOR operation
U_{pw}	User's password
$auth_{ticket}$	Authentication ticket
Ser_{ticket}	Server ticket used to access services in the target server
Sk_{UA}	Secret key, used between the user and AS
SK_{AServ}	Secret key, used between the AS and server
SK_{UServ}	Secret key, used between the user and cloud server
N_i	Random number used for one
shareImgA	Secret share image used for authentication
shareImgB	Image retrieved by the AS used for authentication
$shareImgB\_1$, $shareImgB\_2$	The first part and the second part of the image generated by AS
ShareG, ShareB, ShareR	Shares secret image
$Life_{time}$	Ticket's validity
timestamp	The maximum session time
C^i	Challenge for the i-th interaction
R^i	Response of the PUF for C^i
NA, NB	Values computed by the AS
NC	Value computed by user and server.

1. System model

The system model adopted in this work is shown in Figure 2. As already mentioned in the previous section, the system model consists of three main components: user platform, authentication server, and IoT-cloud platform. The authentication process is considered for 7 communications scenarios. i) The user using a mobile phone or other devices sends a request to join the cloud platform to AS. ii) AS verifies the user's request and sends the granted ticket to the user. This ticket contains all the credentials for establishing a connection between the user and the cloud server. iii) At the same time the AS sends the user's authentication credentials to the specific server that the user wants to access. iv) The user then sends an authenticator message received from the AS to the server. v) The server verifies the authentication credentials received from the user and sends an authentication request to AS. vi) The AS verifies the authentication message received from the server and sends a confirmation message to the server if the received credentials match the stored credentials. Otherwise, it sends an unauthorized access message. vii) The server authenticates the user according to the acknowledgment message received from the AS. If the server receives an authorized access message from AS, then, it accepts the user request and establishes a communication session with the user. Otherwise, it rejects the user's request and the authentication process fails.

2. Encryption phase

The main goal of the encryption process is to hide all the secret information contained in the original image using the visual cryptography technique. This makes the image meaningless. Even if the encrypted image is intercepted by the attacker during the authentication process, he/she cannot reveal the original image. Only the user and the server can retrieve the secret image. In addition, the secret image is kept secret between the user and the receiver. To enhance the security robustness in images, various image encryption schemes have been developed. These schemes provide new technologies in the area of image cryptography which requires less computation and less storage.

The encryption method adopted in our proposed protocol is proposed by Khalid *et al.* [15]. This image encryption method is used to encrypt halftone color image by generating two shares random and key share, which are the same size as the original color image. These two shares are based on a private key shared by the sender and the receiver. At the receiving side the original color image is revealed by stacking the two shares and exploiting the human vision system. As we can see in the Figure 3, the encryption process consists of 4 steps including generation of the halftone image, split the halftone image, the generation of the mask secret, and the generation of the secret image. The details of each step are discussed as follows.

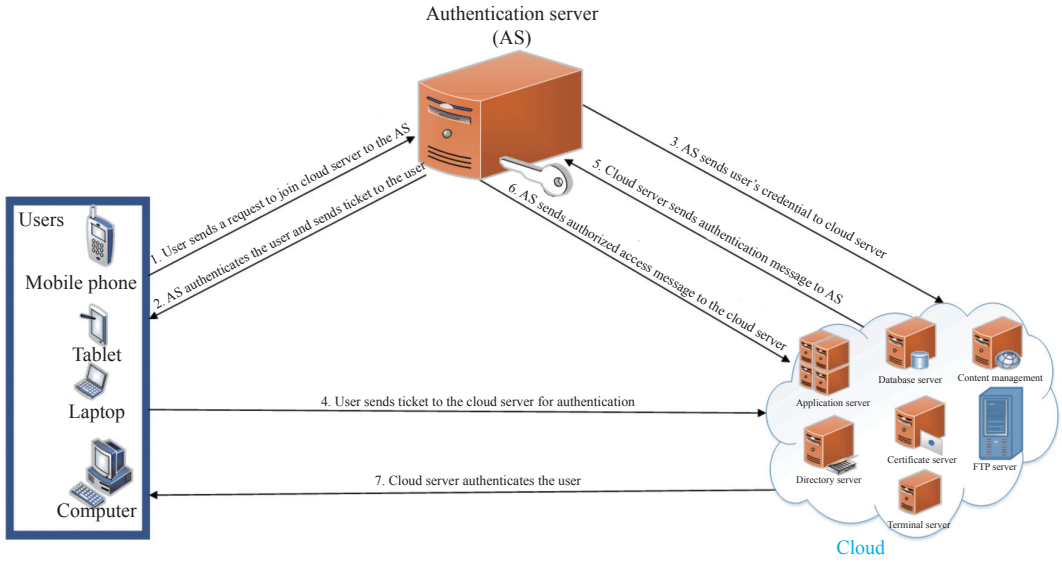

Figure 2 System model.

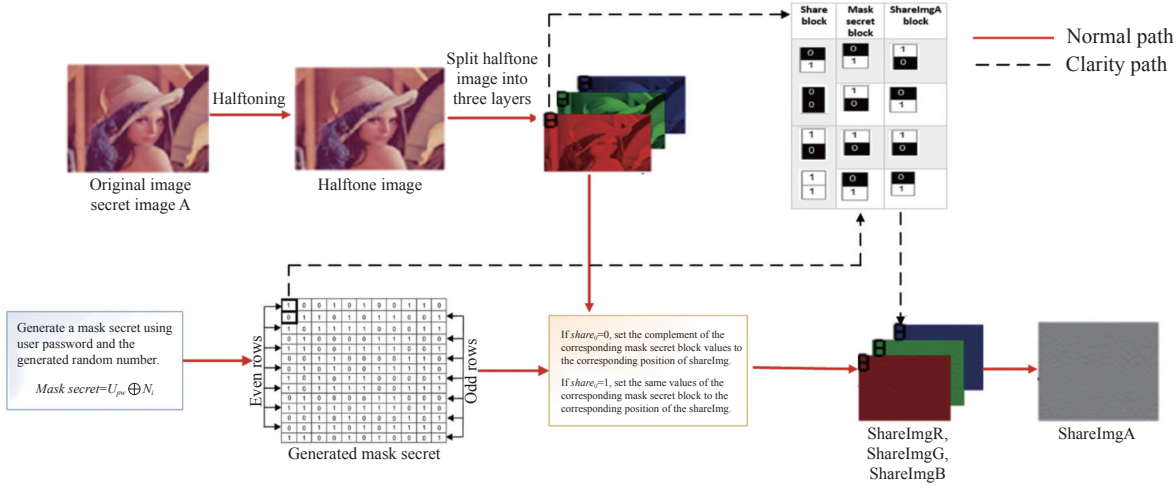

Figure 3 Image encryption process.

Step 1 Create the halftone image.

The dithering technique is used to halftone the original image into binary image.

Step 2 Split the half toned image.

The half toned image is split into three layers (ShareR, ShareG, and ShareB).

Step 3 Generate the mask secret.

The mask secret is a random share generated using the XOR of the user password and a random number generated by the user. The mask secret is composed of a set of rows and columns and has the same size as the original image. There are two types of rows including even rows and odd rows. The pixel in the even rows are filled with a random value (either 0 or 1). The odd rows are filled with the complement value of the pixels in the even rows directly above it. For example, if the pixel in the even rows is black then the pixel in the odd row beneath it is white. The mask secret is generated on both the user and the AS sides and is not shared through the network. If the attacker intercepts the authenticator (shareImgA), he/she will not be able to decrypt it without having the mask secret. The mask secret is used to generate the secret image (i.e., ShareImgA).

Step 4 Generate the secret image.

The secret image (ShareImgA) has three layers (i.e., ShareImgR, ShareImgG, ShareImgB). Each layer of the secret image is constructed based on the mask secret and the layers of the half toned image. Each layer of the half toned image is divided into non-overlapping blocks of size 2×1 pixels. The secret image is generated block by block. If the pixel in the half toned block is black the corresponding block of the secret image is filled with the complement value of the two pixels in the corresponding block of the mask secret. However, if the pixel in the half toned block is white then the corresponding block of the secret image is filled with the same values of the corresponding block of the mask secret. The secret image is formed by combining the resulting layers (i.e., ShareIm-

gR, ShareImgG, and ShareImgB). This secret image has the same size as the original image without pixel expansion, reducing the storage space.

3. Decryption phase

The decryption process is shown in Figure 4. The secret image (ShareImgA) received from the user is split into three layers (ShareImgR, ShareImgG, ShareImgB). To reveal the original image, the Authentication Server uses the user password stored in his database and the random number received from the user to generate the same mask secret based on the same process as the encryption process. Each layer is stacked with the mask secret to generate another three layers (ShareR, ShareG, and ShareB). By combining these three layers the original image is retrieved.

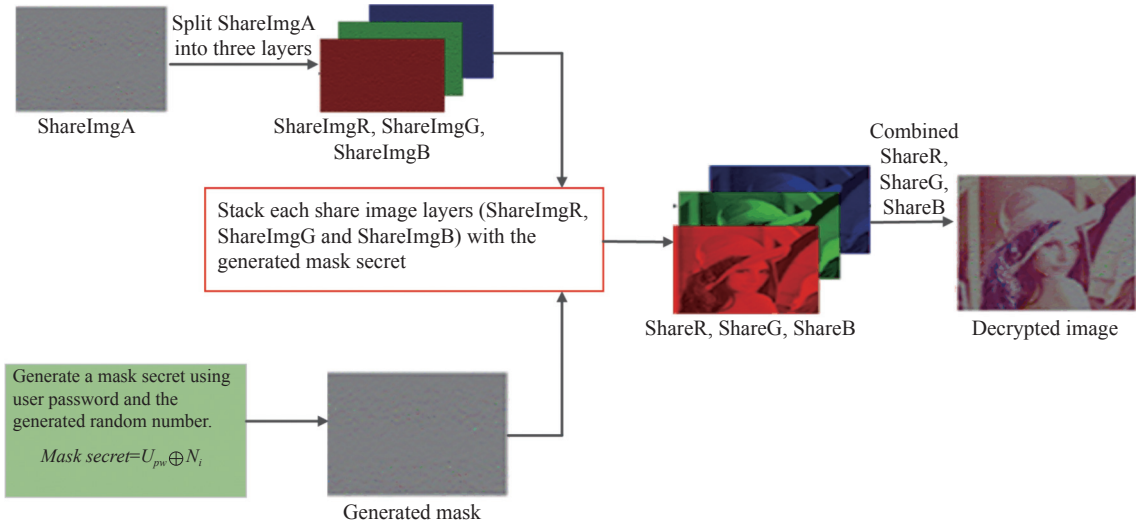

Figure 4 Image decryption process.

4. Mutual authentication process

In this section, we present the mutual authentication process. The different steps of the mutual authentication process are shown in Figure 5. The role of authentication is to ensure that only authorized users can access the cloud server. When the user wants to access the cloud platform, he needs to request a ticket from the authentication server. The AS authenticates the user and returns a ticket that permits him to access the services. Only authenticated users can get the ticket for accessing the server. Before the mutual authentication process, an initial registration process is required. At this step, the user needs to create an account by providing his credentials (username, ID, password, etc.). These credentials are saved in AS database. The next time, if the user wants to log in, he needs to input his ID and password on the device. The input password is XORed with the random number to generate the mask secret which is used to encrypt the image. The mask secret is generated from both sending and receiving sides using the same password and random number. As the user password is not shared through the network, the hash of the input password is sent to the AS to verify that the user input password and the stored password in the AS database are similar. When the AS receives the message from the user, it calculates the hash of the user's stored password and compares the calculated hash value with the received hash. If they are different, the AS returns a password error message to the user who needs to type the right password. Otherwise, the AS starts to generate the mask secret using the XOR of the user's stored password and the random number received from the user to decrypt the secret image.

The proposed protocol is based on two secret images. The first secret image called "secret image A" is used to ensure the authentication mechanism between the user and the AS using the visual cryptography technique. This image is randomly selected among a set of images stored in the user application platform interface (API) and AS database. The selected image is first half toned into binary image using the dithering technique. Then the halftone image is split into three shares called ShareR, ShareG, and ShareB. The user input password is XORed with the random number to generate the mask secret which is used to encrypt each layer of the selected image. The mask secret is stacked with each layer of halftone image to form three meaningless layers called ShareImgR, ShareImgG, and ShareImgB. These three layers are combined to create the secret image (ShareImgA) which is sent to the AS as an authenticator. The image decryption process at the AS side consists of splitting the secret image into three layers (ShareImgR, ShareImgG, ShareImgB). After that, the user password stored in the AS database is XORed with the random number received from the user to generate the mask secret. Each layer is stacked with the mask secret to form three shares image (ShareR, ShareG, ShareB). The original image is retrieved by combining the shares image.

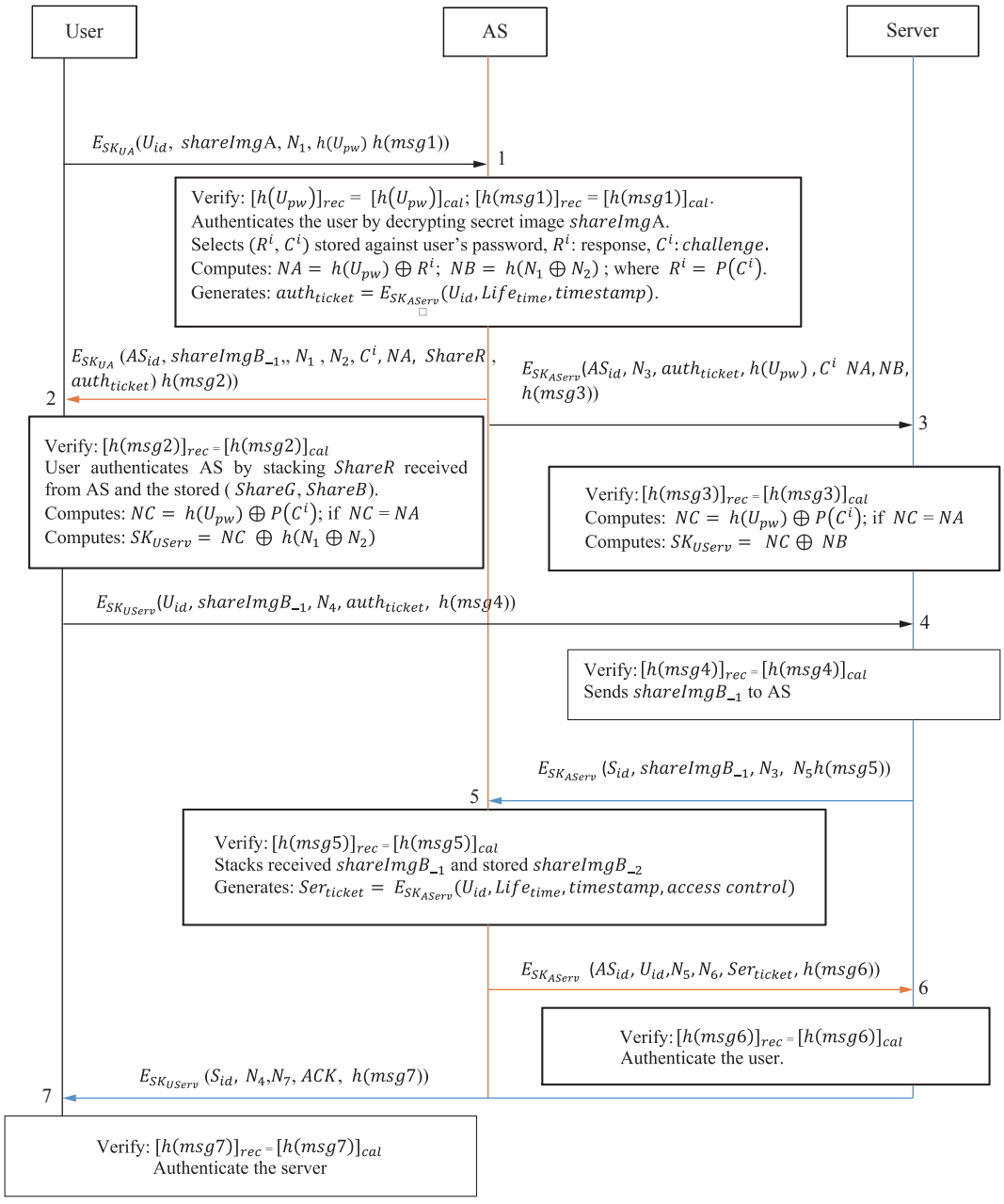

Figure 5 Network architecture.

The AS then compares the retrieved image with the stored image. If they match, the AS authenticates the user and returns a ticket to the user which is used for authentication process with the cloud server. The AS also generates the second image called "secret image B". This image is used for ensuring authentication between the user and the target server and it is only known by the AS. The AS splits the generated image into two shares $shareImgB\_1$ and $shareImgB\_2$ using visual cryptography technique. The first share is sent to the user and the second share is secretly kept by the AS. The user then sends the first part to the server. Once receiving the message from the user, the server transmits $shareImg\_1$ to the AS. The AS then stacks the two shares to retrieve a new image called ShareImgB. After that, the retrieved image is compared with the original image stored in his database. If the two images match, the AS sends an authorized access message to the server. It also generates a service ticket that will allow the user to access cloud server resources. This ticket contains all the user's credentials including the user's ID, timestamp (the maximum session time), lifetime (the ticket expired time), and the user access control policy. Once receiving the message from the AS, the target server authenticates the user by establishing a communication session with the user for data exchange. The mask secret, the authentication ticket, and the service ticket can be represented in the following equations:

$$\text{Mask secret} = U_{pw} \oplus N_i$$

$$auth_{\text{ticket}} = E_{SK_{Aserv}}(U_{id}, Life_{\text{time}}, timestamp)$$

$$Ser_{\text{ticket}} = E_{SK_{Aserv}}(U_{id}, Life_{\text{time}}, timestamp, accesscontrol)$$

The different steps of the authentication process are explained as follows:

Step 1 User $\longrightarrow$ AS

The user initiates the mutual authentication process by sending his ID (U_id), the hash of the input password ($h(U_pw)$, the secret image (shareImgA) the randomly generated challenge number (N_1) to the AS (Authentication Server). This message is encrypted using the secret key (SK_{UA}) shared between the user and the AS during the configuration phase. To ensure data integrity, the hash of the whole message is calculated.

Step 2 AS $\longrightarrow$ User

Once the AS receives the message from the user, it first decrypts it, using the shared secret key (SK_{UA}). Then, it verifies the user's ID to be sure that the user has finished the initial registration. After that, it calculates the hash of the user's password stored in his database and compares it with the received hash value ($[h(U_{pw})]_{rec} = ([h(U_{pw})]_{cal}$). If they are different, the AS returns a password error message to the user and the authentication fails. The user needs to type the right password to start a new authentication process. Otherwise, it calculates the hash of the whole received message and compares the calculated hash with the received hash ($[h(msg1)]_{rec} = [h(msg1)]_{cal}$). If they match, the authentication process continues. Otherwise, the authentication fails. The AS uses the user password and the random number received from the user to generate the mask secret which is used to decrypt the secret image (shareImgA) according to the decryption process explained in Section IV.3. Once the original image is revealed, the AS compares this image with the image stored in his database. If the two images match, the AS authenticates the user. Otherwise, the authentication fails. After that, the AS generates a new image called "secret image B" and splits it into two shares ($shareImgB\_1, shareImgB\_2$) using the visual cryptography technique. The AS also generates a ticket containing the user's ID, username, lifetime, and timestamp. The AS selects the CRP (C^i, R^i) saved against the user's ID (U_{id}) and generates NA, NB used to generate the secret key (SK_{UServ}).

The AS then sends the first part of the secret image B ($shareImgB\_1$), AS_{id}, N_1, N_2, NA, the authentication ticket ($auth_{\text{ticket}}$), and the challenge (C^i) to the user. It also selects one layer among the three layers of the secret image (ShareR) and sends it to the user. The shared secret key (SK_{UA}) is used as the encryption key. The hash of the whole message is also calculated.

Step 3 AS $\longrightarrow$ Server

The AS sends his ID (AS_{id}), N_3, the authentication ticket ($auth_{\text{ticket}}$), $h(U_{pw})$,NA,NB, the challenge (C^i), and the hash of the whole message to the server. Once receiving the message, the server compares the received hash and the calculated hash ($[h(msg3)]_{rec} = [h(msg3)]_{cal}$), if they match, it computes NC using the XOR of the hash of the user password and the output of PUF using C^i as input value. Otherwise, the authentication process fails. Then, the server compares NC with NA received from the AS. If they match, it computes the secret key (SK_{UServ}) using the XOR of NC and NB ($SK_{UServ} = NC \oplus NB$) since the response R^i used by the AS to calculate NA equal the calculated PUF output value ($R^i = P(C^i)$)

Step 4 User $\longrightarrow$ Server

When the user receives the message from the AS, it decrypts it, using the shared secret key. After that, it calculates the hash of the whole received message and compares it with the received hash (i.e., $[h(msg4)]_{rec} = [h(msg4)]_{cal}$). If it is a success the authentication process continues. Otherwise, the authentication fails. The user combines the received layer (ShareR) with the stored layers (ShareG, ShareB). The resulted image is compared to the original image. If the two images match, the user authenticates the AS and the selected original "secret image A" used for the authentication is destroyed at the user side. However, this image is marked as an already used image and saved in a different database at AS side. If an intruder tries to reuse the same image for authentication process, the AS will know that the image had been used and the authentication fails. A new image will be used for other authentication process. The user also computes NC using the XOR of the hash of his password and the output of PUF using C^i as input value and compares NC with NA received from the AS. If they match, it computes the secret key (SK_{UServ}) using the XOR of NC and the hash of XORed value of N_1 and N_2 ($SK_{UServ} = NC \oplus h(N_1 \oplus N_2)$) since the response R^i used by the AS to calculate NA equal the calculated PUF output value ($R^i = P(C^i)$). After that, the user sends $U_{id}, shareImgB\_1, N_2, N_3, auth_{\text{ticket}}$, the hash of the whole message to the target server. SK_{UServ} is used as the encryption key.

Step 5 Server $\longrightarrow$ AS

Once receiving the message from the user, the server first decrypts it, using SK_{UServ}. Then, the hash of the whole received message is calculated and compared to the received hash ($[h(msg5)]_{rec} = [h(msg5)]_{cal}$). If they match, the authentication continues. Otherwise, it fails. The server checks the ticket to see if it matches the user's credentials received from the AS. It also verifies the timestamp to make sure the ticket has not expired. The server sends $shareImgB\_1$ received from the user to the AS for the verification process. For security issues, the "secret image B" is only known by the AS. It is not shared with any party.

Step 6 AS $\longrightarrow$ Server

When the AS gets the message from the server, it decrypts it using the shared secret key (SK_{AServ}). Then, it calculates and compares the calculated hash and the received hash. If they are different, the authentication process fails. Otherwise, the authentication process continues by stacking $shareImgB\_1$ received from the serv-

er with the second part of the "secret image B" already stored in his database to retrieve a new image called ShareImgB. The AS compares the resulted image with the original image. If the two images match, the AS generates a new ticket called service ticket (Ser_{ticket}) which will allow the user to access to server's resources. The service ticket contains user's ID, timestamp, lifetime, and user's access control policy. All encrypted with SK_{AServ}.

Step 7 Server $\longrightarrow$ User

Once the server receives the message from the AS, it decrypts it and verifies the integrity of the received message by calculating and comparing the hash of the whole message to the received hash ($[h(msg7)]_{rec} = [h(msg7)]_{cal}$). If verification match, the authentication process continues. Otherwise, it fails. After that, the server checks the service ticket to see all the access control policy related to the user request. It also checks the lifetime which determines the ticket expired time and the timestamp, the maximum session time. Once the checking is done, the target server authenticates the user and establishes a communication session with the user for data transmission. The user can now engage in a secure communication session with the server.

Once the mutual authentication process is completed the user and the server secretly share the session key for secure communication session. In addition, the AS computes the new Challenge-response pair (CRP) and saves it against the user's ID for next mutual authentication process. It computes as follows:

$$C^{i+1} = h(U_{pw}) \oplus h(N_i \oplus N_{i+1}) \oplus h(NA \oplus NB)$$
$$R^{i+1} = P(C^{i+1})$$
$$NA = h(U_{pw}) \oplus R^{i+1}$$
$$NB = h(N_i \oplus N_{i+1})$$

V. Security Analysis

1. Theoretical security analysis

The proposed protocol can provide various security mechanisms to protect the network against attacks and reduce the IoT-cloud security threats. The use of visual cryptography technique for ensuring mutual authentication enhances the security robustness, making it efficient to protect the user's privacy in IoT-cloud. Furthermore, the security of the different shares is enhanced using the mask secret which is generated by taking the XOR of the user's password and a random number generated by the user. The mask secret is not shared over the network. As a result, if the attacker intercepts the secret image during the transmission process, the original image cannot be retrieved without having the mask secret. Even if, the user's password is revealed, the attacker still cannot generate the mask secret without knowing the random number. In addition, the symmetric algorithm is used for ensuring data confidentiality and protecting the network against men-in-the-middle attacks, replay attacks, etc. Before sending the message, the sender uses the shared secret key to encrypt the message. Once the message is received from the sender, the receiver uses the same key to decrypt the message. All the secret keys are not shared through the network. The secret key shared between the user and the server is computed using the physical unclonable function (PUF) and the XOR operation and never shared over the network. Making it impossible or infeasible for disclosing. The use of PUF to provide authentication also increases the security robustness of the protocol. Moreover, data integrity is provided using the hash function. Indeed, the hash of the whole message is calculated before transmitting the message. Once the receiver gets the message from the sender, it first calculates the hash of the whole received message and compares it with the received hash if they match, the receiver authenticates the sender as he knows that the message is not altered during the transmission. In this proposed scheme, if the hacker performs a brute force attack or password guessing to get the user's password, it cannot succeed. Because the password is combined with a random number to generate the mask secret at any time the user logs in the system. As the random number is used once, the attack cannot succeed using the password alone. The authentication mechanism in the proposed protocol is ensured by the authentication server. Before accessing the cloud platform, the user needs to get a ticket from this server. This ticket grants him permission to access to the IoT-cloud services. Mutual authentication ensures that only authenticated and authorized users can access the cloud.

2. Attacks and security proprieties comparison

To validate the security robustness and performances of the proposed protocol, we compare in Table 2 the proposed scheme with other related works in terms of various attacks and security proprieties such as man-in-the-middle attack, mutual authentication, replay attack, unauthorized data base access, dictionary attack, brute force attack, impersonate attack and much more. The related protocols are based on conventional cryptography algorithms such as symmetric and asymmetric algorithms, physical unclonable function, elliptic curve cryptography (ECC), and much more. However, our proposed scheme is a combination of different security mechanisms including visual cryptography, symmetric encryption, hash function, and physical unclonable function.

We used security mechanisms different from traditional cryptography to design the protocol. As we can see in the comparison table, the proposed protocol can resist all attacks presented in that table compared to the related protocols. Most of the existing related schemes cannot resist various attacks such as replay attack, dictionary attack, brute force attack, and impersonate attack. They also cannot achieve mutual authentication and unauthorized database access. For these reasons, we can confirm the supremacy of the proposed protocol over the

Table 2 Attacks and security proprieties comparison with other related protocols

Attacks and security proprieties	Ref. [34]	Ref. [35]	Ref. [36]	Ref. [37]	Ref. [38]	Proposed protocol
Man-in-the-middle attack	Yes	Yes	Yes	Yes	Yes	Yes
Mutual authentication	Yes	Yes	Yes	Yes	No	Yes
Replay attack	Yes	Yes	Yes	No	No	Yes
Unauthorized database access	No	No	No	No	No	Yes
Dictionary attacks	No	No	No	No	Yes	Yes
Brute force attack	No	No	No	Yes	No	Yes
Impersonate attack	No	Yes	No	Yes	No	Yes
Password guessing	No	No	Yes	No	NO	Yes
Server masquerade	No	No	No	No	NO	Yes
Protect user privacy	No	No	Yes	No	NO	Yes
Data integrity	Yes	Yes	Yes	Yes	Yes	Yes
Confidentiality	Yes	Yes	Yes	Yes	Yes	Yes
Access control policy	No	No	No	No	No	Yes

Note: No: does not resist attack and does not preserve the security properties; Yes: resist attack and preserve the security properties.

other related protocols in terms of security and performance. This proves that, designing a secure mutual authentication protocol based on visual cryptography is a better solution to protect user's privacy and data in the IoT-clod platform.

3. Communication cost comparison

The communication cost is an important factor to evaluate the efficiency of the designed protocol. It helps determining the total number of bits transmitted during the authentication process. The lower the communication cost is, the higher the performance the protocol. To determine the efficiency of our protocol, we analyzed the designed protocol and make a comparison with other proposed scheme.

Without loss of generality, we assume that the size of encryption and decryption value is 512 bits, the size of the hash function output is 160 bits, the size of pseudo identity and other elements presented in Table 1 is 128 bits. By considering the communication phase in Figure 5, we can see that all the authentication credentials are encrypted before transmitting to the receiver. As the mutual authentication process is completed after seven steps, the communication cost in our work is $512 \times 7 = 3584$ bits. The total cost during the mutual authentication between the user and the cloud server is 3584 bits. By comparing this value with the related scheme shown in Figure 6 and Table 3, we find that our protocol requires low communication cost compared to [39] and [36]. it has a higher communication cost compared to [40] and [41]. But the difference is acceptable. Therefore, we can confirm that the proposed scheme is efficient and can provide secure mutual authentication mechanism in IoT-cloud environment.

4. BAN-logic security proof

The security proof plays an important role in designed network security protocol. In order to validate the

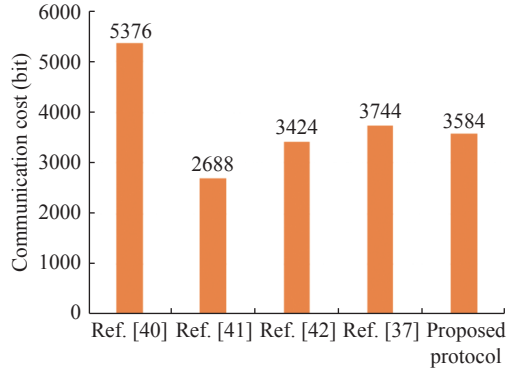

Figure 6 Communication cost Comparison.

Table 3 Communication cost comparison

Protocols	Communication cost (bit)
Ref. [39]	5376
Ref. [40]	2688
Ref. [41]	3424
Ref. [36]	3744
Proposed protocol	3584

designed protocol, the designers need to find the best method to analyze and prove the security robustness of the protocol. Many security proof methods exist. Among these methods, BAN-logic method is the widely-used. The BAN-logic method is a set of rules for defining, analyzing, and proving the security robustness. It proves how the designed protocol can provide security mechanism to protect the network against attacks. It is also, a robust validation logic method to prove the mutual authentication of an authentication protocol. In this work, we used the BAN-logic method to prove the security robustness of the proposed protocol. We started the BAN-logic analysis by defining the different security goals (4 goals). After that, we made the assumptions about the

initialization and set the idealization form of the exchanged message. By using the BAN-logic rules, we performed the security proof of the protocol step by step. We achieved all the defined goals after 15 steps. This proves that the proposed protocol has a good security and performance. It can be used to protect user's privacy in IoT-cloud platform. Furthermore, the protocol is very efficient in protecting IoT-cloud network against various attacks. The combination of visual cryptography, hash function, and symmetric algorithm enhances the security robustness of the protocol. Making it an efficient protocol to provide a secure mutual authentication mechanism. The BAN-logic security proof process is presented below

1) The assumptions: We make assumptions about the initialization as follows.

R_1: $\dfrac{P \mid\equiv Q \overset{K_p,Q}{\longleftrightarrow} P, P \triangle \{X\}_{K_pQ}}{P \mid\equiv Q \mid\sim X}$

R_2: $\dfrac{P \mid\equiv \sharp(X), P \mid\equiv Q \mid\sim X}{P \mid\equiv X}$

R_3: $\dfrac{P \mid\equiv Q \mid\Rightarrow X, P \mid\equiv Q \mid\equiv X}{P \mid\equiv X}$

2) Security goals: We aim to satisfy the following security goals

$Goal_1$: $User \mid\equiv User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server$

$Goal_2$: $Server \mid\equiv User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server$

$Goal_3$: $User \mid\equiv Server \mid\equiv User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server$

$Goal_4$: $Server \mid\equiv User \mid\equiv User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server$

$Secret_1$: $shareImgA, h(U_{pw})$

$Secret_2$: $shareR$

3) The assumptions: We make assumptions about the initialization as follows.

P_1: $User \mid\equiv User \overset{SK_{UA}}{\longleftrightarrow} AS$

P_2: $AS \mid\equiv AS \overset{SK_{UA}}{\longleftrightarrow} User$

P_3: $Server \mid\equiv AS \overset{SK_{Aserv}}{\longleftrightarrow} Server$

P_4: $AS \mid\equiv AS \overset{SK_{Aserv}}{\longleftrightarrow} AS$

P_5: $User \mid\equiv \sharp(N_1)$

P_6: $AS \mid\equiv \sharp(N_3)$

P_7: $User \mid\equiv \sharp(N_4)$

P_8: $Server \mid\equiv \sharp(N_5)$

P_9: $User \mid\equiv AS \mid\Rightarrow User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server$

P_{10}: $Server \mid\equiv AS \mid\Rightarrow User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server$

P_{11}: $User \mid\equiv AS \mid\Rightarrow User \overset{SK_{Userv}}{\Longleftrightarrow} Server$

P_{12}: $Server \mid\equiv AS \mid\Rightarrow User \overset{SK_{Userv}}{\Longleftrightarrow} Server$

P_{13}: $Server \mid\equiv \sharp(auth_{\text{ticket}})$

4) The idealized form: The idealized from message exchange is as follow:

msg_1: $User \longrightarrow AS(AS_{id}, Secret_1, N_1)_{SK_{UA}}$

msg_2: $AS \longrightarrow User(AS_{id}, User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server, N_1, Secret_2, User \overset{SK_{Userv}}{\longleftrightarrow} Server, auth_{\text{ticket}})_{SK_{UA}}$

msg_3 : $AS \longrightarrow Server(AS_{id}, N_3, auth_{ticket}, User \overset{SK_{Userv}}{\longleftrightarrow} Server)_{SK_{Aserv}}$

msg_4: $User \longrightarrow Server(U_{id}, User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server, N_4, Secret_2, auth_{\text{ticket}})SK_{Userv}$

msg_5: $Server \longrightarrow AS(S_{id}, User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server, N_3, N_5)_{SK_{Aserv}}$

msg_6: $AS \longrightarrow Server(AS_{id}, U_{id}, N_5, User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server, N_3, N_5)_{SK_{Aserv}}$

msg_7: $Server \longrightarrow User(S_{id}, N_4, N_7)_{SK_{Userv}}$

5) BAN-logic proving process

$Step_1$: From msg_1, P_2, using R_2 we get $AS \mid\equiv User \mid\sim (U_{id}, Secret_1, N_1)$

$Step_2$: From msg_2, P_1, using R_1 we get $User \mid\equiv AS \mid\sim (User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server, User \overset{SK_{Userv}}{\longleftrightarrow} Server)$

$Step_3$: From $Step_2, P_5$, using R_2, we get $User \mid\equiv AS \mid\equiv (User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server, User \overset{SK_{Userv}}{\longleftrightarrow} Server)$

$Step_4$: From $Step_3, P_9$, using R_3, we get $User \mid\equiv (User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server, User \overset{SK_{Userv}}{\longleftrightarrow} Server), get$, $Goal_1$

$Step_5$: From msg_3, P_3, using R_1, we get $Server \mid\equiv AS \mid\sim (AS_{id}, N_3, User \overset{SK_{Userv}}{\longleftrightarrow} Server)$

$Step_6$: From $Step_5, P_{13}$, using R_2, we get $Server \mid\equiv AS \mid\equiv (N_3, User \overset{SK_{Userv}}{\longleftrightarrow} Server)$

$Step_7$: From $Step_6, P_{12}$, using R_3, we get $Server \mid\equiv (User \overset{SK_{Userv}}{\longleftrightarrow} Server)$

$Step_8$: From $Step_7, msg_4$, using R_4, we get $Server \mid\equiv User \mid\sim (U_{id}, N_3, User \overset{SK_{Userv}}{\longleftrightarrow} Server, N_4, auth_{ticket})$

$Step_9$: From msg_5, P_4, using R_1, we get $AS \mid\equiv Server \mid\sim (S_{id}, User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server, N_3, N_5)$

$Step_{10}$: From msg_6, P_3, using R_1, we get $Server \mid\equiv Server \mid\sim (AS_{id}, U_{id}, N_5, User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server)$

$Step_{11}$: From $Step_{10}, P_8$, using R_2, we get $Server \mid\equiv AS \mid\equiv (AS_{id}, U_{id}, N_5, User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server)$

$Step_{12}$: From $Step_{11}, P_{10}$, using R_3, we get $Server \mid\equiv User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server)$, we get, $Goal_2$

$Step_{13}$: From $Step_8, P_{13}$, using R_2, we get $Server \mid\equiv User \mid\equiv (User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server)$, we get, $Goal_4$

$Step_{14}$: From $msg_7, Step_4$, using R_1, we get $User \mid\equiv Server \mid\sim (N_4, User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server)$

$Step_{14}$: From $Step_{14}, P_7$, using R_2, we get $User \mid\equiv Server \mid\equiv (User \overset{shareImgB_{-1}}{\Longleftrightarrow} Server)$, we get, $Goal_3$, (end)

VI. Conclusion

This paper presents a secure authentication protocol based on the visual cryptography technique. This work aims to ensure user authentication mechanisms in IoT cloud platforms. The proposed protocol architecture contains three main components. IoT platform, IoT-cloud platform, and user platform. The role of the IoT platform is to collect data from various sources across the world. Such data are then stored and processed in

the cloud platform using big data technologies. Before accessing to such data, the user needs to authenticate himself and get a ticket from the AS. This ticket permits him to access to IoT-cloud services. The authentication process is based on two secret images (secret image A, secret image B) and two tickets (authentication ticket, service ticket). Furthermore, the protocol ensures that only authenticated and authorized users can access the cloud. The security robustness is enhanced by using four security mechanisms: visual cryptography, mask secret, symmetric algorithm, and hash function. We analyze the proposed protocol using BAN-logic method and the results prove that the proposed protocol can provide a high-security mechanism to guarantee data integrity, confidentiality, and authenticity. The protocol is also robust and can resist many attack types. As future work, we will enhance our proposed scheme to protect the privacy of biometric data in biometric authentication by splitting the enrolled biometric image into several shares.

Acknowledgement

This paper was supported by the Natural Science Foundation of China (Grant Nos. U21B2021, 61932014, and 61972018) and the Beijing Natural Science Foundation (Grant No. 4202037).

References

[1] Rose Karen, Eldridge Scott, and Chapin Lyman, "The internet of things: An overview," *The internet society (ISOC)*, vol. 80, pp. 1–50, 2015.

[2] X. William, "GIV 2025 unfolding the industry blueprint of an intelligent world, " Available at: https://www.huawei.com/minisite/giv/Files/whitepaper_en_2018.pdf, 2018.

[3] A. Aich and A. Sen, "Study on cloud security risk and remedy," *International Journal of Grid and Distributed Computing*, vol. 8, no. 2, pp. 155–166, 2015.

[4] R. Kalaiprasath, R. Elankavi, and R. Udayakumar, "Cloud security and compliance-a semantic approach in end to end security," *International Journal on Smart Sensing and Intelligent Systems*, vol. 10, no. 5, pp. 482–494, 2017.

[5] A. Hendre and K. P. Joshi, "A semantic approach to cloud security and compliance," in *Proceedings of 2015 IEEE 8th International Conference on Cloud Computing*, New York, NY, USA, pp. 1081–1084, 2015.

[6] A. B. Rabiah, K. K. Ramakrishnan, E. Liri, *et al.*, "A lightweight authentication and key exchange protocol for IoT," in *Proceedings of the Workshop on Decentralized IoT Security and Standards*, San Diego, CA, USA, pp. 1–6, 2018.

[7] M. N. Aman, K. C. Chua, and B. Sikdar, "A light-weight mutual authentication protocol for IoT systems," in *Proceedings of 2017 IEEE Global Communications Conference*, Singapore, Singapore, pp. 1–6, 2017.

[8] Y. H. Chuang, N. W. Lo, C. Y. Yang, *et al.*, "A lightweight continuous authentication protocol for the internet of things," *Sensors*, vol. 18, no. 4, article no. 1104, 2018.

[9] R. K. Sheu, M. S. Pardeshi, and L. C. Chen, "Autonomous mutual authentication protocol in the edge networks," *Sensors*, vol. 22, no. 19, article no. 7632, 2022.

[10] A. M. Abdul, S. Jena, and M. B. Raju, "Secure authentication protocol to cloud," *International Journal of Computer Sciences and Engineering*, vol. 7, no. 5, pp. 1551–1557, 2019.

[11] M. Naor and A. Shamir, "Visual cryptography," in *Proceedings of the Workshop on the Theory and Application of of Cryptographic Techniques*, Perugia, Italy, pp. 1–12, 1995.

[12] A. Walke, J. Bhanushali, A. Rajgor, *et al.*, "Enhanced password processing scheme using visual cryptography and steganography," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 6, no. 4, pp. 35–37, 2018.

[13] H. Y. Qin, T. Matsusaki, Y. Momoi, *et al.*, "Dual visual cryptography using the interference color of birefringent material," *Journal of Software Engineering and Applications*, vol. 10, no. 8, pp. 754–763, 2017.

[14] T. Matsuzaki, H. Y. Qin, and K. Harada, "Color visual cryptography with stacking order dependence using interference color," *Open Journal of Applied Sciences*, vol. 7, no. 7, pp. 329–336, 2017.

[15] R. I. Al-Khalid, R. A. Al-Dallah, A. M. Al-Anani, *et al.*, "A secure visual cryptography scheme using private key with invariant share sizes," *Journal of Software Engineering and Applications*, vol. 10, no. 1, pp. 1–10, 2017.

[16] D. Devakumari and K. Geetha, "A survey of visual cryptographic method for secure data transmission," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 6, no. 6, pp. 270–274, 2017.

[17] P. Chouksey, R. Miri, K. Srinivas, *et al.*, "A secret share and key generation based visual cryptography approach for retaining 2D and 3D RGB color using transposition," *Turkish Journal of Computer and Mathematics Education*, vol. 12, no. 12, pp. 695–708, 2021.

[18] Z. S. Xu, J. B. Xu, and L. D. Kuang, "A token-based authentication and key agreement protocol for cloud computing," in *Proceedings of the IEEE 6th International Conference on Smart Cloud*, Newark, NJ, USA, pp. 38–43, 2021.

[19] H. Al-Refai, K. Batiha, and A. M. Al-Refai, "An enhanced user authentication framework in cloud computing," *International Journal of Network Security & Its Applications*, vol. 12, no. IJNSA, pp. 59–75, 2020.

[20] Alshammari Abdulaziz, Alhaidari Sulaiman, Alharbi Ali, and Zohdy Mohamed, "Security threats and challenges in cloud computing," *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*, New York, NY, USA, pp. 46–51, 2017.

[21] Kazuki Murakami, Ryota Hanyu, Qiangfu Zhao, and Yuya Kaneda, "Improvement of security in cloud systems based on steganography," in *Proceedings of 2013 International Joint Conference on Awareness Science and Technology & Ubi-Media Computing (iCAST 2013 & UMEDIA 2013)*, Aizu-Wakamatsu, Japan, pp. 503–508, 2013.

[22] A. Y. AlKhamese, W. R. Shabana, and I. M. Hanafy, "Data security in cloud computing using steganography: A review," in *Proceedings of 2019 International Conference on Innovative Trends in Computer Engineering*, Aswan, Egypt, pp. 549–558, 2019.

[23] S. M. J. Islam, Z. H. Chaudhury, and S. Islam, "A simple and secured cryptography system of cloud computing," in *Proceedings of 2019 IEEE Canadian Conference of Electrical and Computer Engineering*, Edmonton, Canada, pp. 1–3, 2019.

[24] A. Arora, A. Khanna, A. Rastogi, *et al.*, "Cloud security ecosystem for data security and privacy, " in *Proceedings of the 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence*, Noida, India, pp. 288–292, 2017.

[25] K. Fan, Q. Luo, K. Zhang, *et al.*, "Cloud-based lightweight secure RFID mutual authentication protocol in IoT," *Information Sciences*, vol. 527, pp. 329–340, 2020.

[26] M. Adeli, N. Bagheri, S. Sadeghi, *et al.*, "χperbp: A cloud-based lightweight mutual authentication protocol, " *Peer-to-Peer Networking and Applications*, in press.
[27] K. Nimmy and M. Sethumadhavan, "Novel mutual authentication protocol for cloud computing using secret sharing and steganography," in *Proceedings of the Fifth International Conference on the Applications of Digital Information and Web Technologies*, Bangalore, India, pp. 101–106, 2014.
[28] C. Vorugunti, M. Sarvabhatla, and G. Murugan, "A secure mutual authentication protocol for cloud computing using secret sharing and steganography," in *Proceedings of 2014 IEEE International Conference on Cloud Computing in Emerging Markets*, Bangalore, India, pp. 1–8, 2014.
[29] S. R. Department, "Internet of Things - Number of connected devices worldwide 2015-2025," Available at: https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide, 2016-11-27.
[30] R. Dirk, "Visual cryptography," Available at: https://www.ciphermachinesandcryptology.com/en/visualcrypto.htm, 2022-02-25.
[31] G. Ateniese, C. Blundo, A. De Santis, *et al.*, "Visual cryptography for general access structures," *Information and computation*, vol. 129, no. 2, pp. 86–106, 1996.
[32] Z. Zhou, G. R. Arce, and G. Di Crescenzo, "Halftone visual cryptography," *IEEE Transactions on Image Processing*, vol. 15, no. 8, pp. 2441–2453, 2006.
[33] J. A. Kumar and G. Ganapathy, "A modified approach for Kerberos authentication protocol with secret image by using visual cryptography," *International Journal of Applied Engineering Research*, vol. 12, no. 21, pp. 11218–11223, 2017.
[34] Y. Zheng and C. H. Chang, "Secure mutual authentication and key-exchange protocol between PUF-embedded IoT endpoints," in *Proceedings of 2021 IEEE International Symposium on Circuits and Systems*, Daegu, Korea, pp. 1–5, 2021.
[35] P. K. Panda and S. Chattopadhyay, "A secure mutual authentication protocol for IoT environment," *Journal of Reliable Intelligent Environments*, vol. 6, no. 2, pp. 79–94, 2020.
[36] W. Li, X. L. Li, J. T. Gao, *et al.*, "Design of secure authenticated key management protocol for cloud computing environments," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1276–1290, 2021.
[37] M. A. Kiran, S. K. Pasupuleti, and R. Eswari, "A lightweight two-factor mutual authentication scheme for cloud-based IoT," in *Proceedings of the 4th International Conference and Workshops on Recent Advances and Innovations in Engineering*, Kedah, Malaysia, pp. 1–6, 2019.
[38] M. A. Al Sibahee, S. F. Lu, Z. A. Abduljabbar, *et al.*, "Lightweight secure message delivery for E2E S2S communication in the IoT-cloud system," *IEEE Access*, vol. 8, pp. 218331–218347, 2020.
[39] V. Odelu, A. K. Das, and A. Goswami, "A secure biometrics-based multi-server authentication protocol using smart cards," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 1953–1966, 2015.
[40] A. G. Reddy, E. J. Yoon, A. K. Das, *et al.*, "Design of mutually authenticated key agreement protocol resistant to impersonation attacks for multi-server environment," *IEEE Access*, vol. 5, pp. 3622–3639, 2017.
[41] D. B. He, S. Zeadally, N. Kumar, *et al.*, "Efficient and anonymous mobile user authentication protocol using self-certified public key cryptography for multi-server architectures," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 9, pp. 2052–2064, 2016.

Brou Bernard Ehui received the M.S. degree in science degree in computer science and technology from Beijing Technology and Business University, Beijing, China, in 2018. He is currently working towards the Ph.D. degree in cyber security at Beihang University, Beijing, China. His research interests include the Internet of things, mutual authentication, wireless networks and cryptography.
(Email: brou.ehui@yahoo.com)

Chen CHEN received the B.E. degree in computer science and technology from Beihang University, Beijing, China, in 2017. He currently is studying for the Ph.D. degree in Beihang University. His research direction focuses on security protocols and cryptography. (Email: chen4chen@buaa.edu.cn)

Shirui WANG received the B.S. degree at Beihang University in 2022. She is currently pursuing the M.S. degree in electronic information at the Sino-French Engineering School of Beihang University, China. The research interests include biometric encryption and security protocol. (Email: valleyxht@126.com)

Hua GUO received the Ph.D. degree from Beihang University, China, in 2011. She is currently an Associate Professor in School of Cyber Science and Technology, Beihang University, China. Her research interest includes cryptography and security protocol.
(Email: hguo@buaa.edu.cn)

Jianwei LIU received the B.S. and M.S. degrees in electronics and information from Shandong University, Shandong, China in 1985 and 1988, respectively. He received the Ph.D. degree in communication and electronic systems from Xidian University Shaanxi, China in 1998. He is now a Professor of electronic and information engineering at Beihang University, Beijing, China. His current research interests include wireless communication networks, cryptography, and information & network security.
(Email: liujianwei@buaa.edu.cn)