Digital Communications and Networks xxx (xxxx) xxx

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.keaipublishing.com/dcan

Study on the transaction linkage technique combined with the designated terminal for 5G-enabled IoT

Kyungroul Lee [a], Kangbin Yim [b,*]

[a] School of Computer Software, Daegu Catholic University, Gyeongsan, 38430, South Korea

[b] Department of Information Security Engineering, Soonchunhyang University, Asan, 31538, South Korea

ARTICLE INFO

Keywords:
Transaction-linkage technique
Designated terminal
Internet banking
Online-authentication mechanism
5G-enabled IoT

ABSTRACT

With the growth of the scale of the market for Internet banking and e-commerce, the number of Internet-based financial markets has been increasing. Meanwhile, hacking incidents continuously affect the Internet-banking services. For this reason, a countermeasure is required to improve the security of the online identification process. The current security and authentication mechanisms applied to financial services, such as Internet banking services for 5G-enabled IoT, do not ensure security. In this paper, a transaction-linkage technique with which the designated terminal is combined is proposed to solve this fundamental problem. The technique improves the security of online identification mechanisms because it is possible to counteract all of the existing security threats. The proposed technique supports mutual authentication and is safe from eavesdropping attacks, replay attacks, spoofing attacks, and service-denial attacks. Moreover, the technique supports non-repudiation by storing the transaction history in a transaction-linkage device. We believe that the security of Internet-banking services for 5G-enabled IoT will be increased through the utilization of the proposed technique.

1. Introduction

With the growth of the scale of the market for Internet banking and e-commerce, the exchange of goods and services on the Internet has become a large part of the national economy [1]. Even though a variety of security techniques are applied in the processes of building these systems, hacking incidents still affect Internet banking services. Moreover, this kind of damage is continuous. Therefore, general security applications and techniques are needed for the online financial service to ensure security requirements such as confidentiality, integrity, availability, and non-repudiation [2]. Various cryptography-based mechanisms have been developed to satisfy these requirements over the past few decades, and their effectiveness was sufficiently proved through the utilization of proven mathematical tools [3]. Nevertheless, most of the security problems emerge in the process or the environment of applying the security techniques rather than in the cryptography-based technologies, so there is a need to research the vulnerabilities beyond the cryptography-based technologies and to identify measures that can counteract these vulnerabilities properly.

The identification methods are classified into *offline identification methods* and *online identification methods*, as shown in Fig. 1. The offline identification methods are further classified into the entrance and exit controls, confirmation of identity, and privilege settings. The entrance and exit controls verify the users through surveillance, for which monitoring devices such as CCTV (Closed Circuit TeleVision) are used [4]. The confirmation of identity verifies users through an identification card and a face-to-face examination. The privileged setting verifies users through an admission card that assigns privileges to users based on their confirmed identity, and the areas the user can access are restricted based on the privilege level. The online identification methods are further classified into account management, device protection, owner proof, and environment proof. Account management verifies users through knowledge-based information such as the ID/password combination [5]. Device protection protects input and output devices such as the keyboard, secure keyboard, and virtual keyboard [6]. Owner proof verifies users through the OTP [7], certificates [8], secure card, and multi-factor authentication [9] that come from the owner's device and certificate. And environment proof verifies users through the designated PC [10], multi-channel authentication [11], and biometric authentication [12,13] that come from the user's environment, which is a specific environment.

The insider field is applied to cryptography technology and platform security, such as the secure keyboard to protect inside-identification methods. And the outside field is applied to network security and

* Corresponding author.
E-mail addresses: carpedm@cu.ac.kr (K. Lee), yim@sch.ac.kr (K. Yim).

https://doi.org/10.1016/j.dcan.2020.12.003
Received 26 October 2019; Received in revised form 30 October 2020; Accepted 6 December 2020
Available online xxxx

Please cite this article as: K. Lee, K. Yim, Study on the transaction linkage technique combined with the designated terminal for 5G-enabled IoT, Digital Communications and Networks, https://doi.org/10.1016/j.dcan.2020.12.003

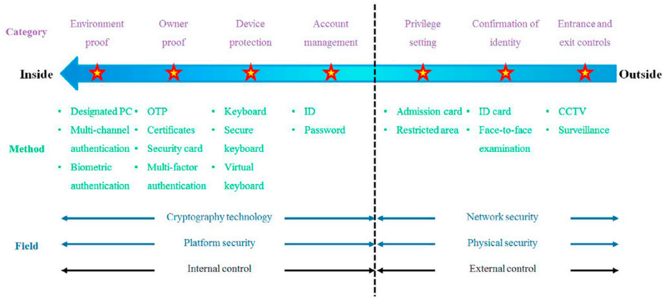


Fig. 1. The scope of the identification methods.

physical security such as SSL (Secure Socket Layer) [14]. The online identification methods do not ensure security due to the existing and new security threats regarding the above identification methods [15,16]. These existing and new security threats are classified as Transport Layer Security (TLS) attacks and keyboard data exposure. Duong et al. presented a chosen plaintext attack against SSL and TLS that allowed an attacker to decrypt authentication tokens [15]. A blockwise chosen-boundary attack was introduced to obtain plaintext HTTP headers by man-in-the-middle attackers [17]. Nadhem et al. presented ciphertext-only plaintext recovery attacks against TLS [18]. [19,20] introduced and ciphertext-only plain-text recovery attacks when using the RC4 encryption algorithm. For Internet banking services, sensitive information related to privacy is input from the keyboard device. Lee et al. introduced new findings on keyboard data attack techniques related to hardware [16,21]. In this paper, we propose a transaction-linkage technique with which the designated terminal is combined to solve the threat-exposure problem regarding the Internet-banking service. The currently applied security technologies ensure security because they can counteract most of the researched security threats, but this does not ensure security against new threats. Therefore, we analyze some new major security threats with respect to the supported environment for online identification methods and Internet banking services.

For this reason, the transaction-linkage technique for which the designated terminal is combined is proposed to fundamentally solve the problem. In the cases where existing transaction-linkage techniques are used, the exposure problem rising from the security threats analyzed in this paper can be solved. These techniques can, however, be abused when the transaction-linkage device is stolen, which is the biggest problem of the possession-based identification methods. And the linkage code is exposed because the code is input via the keyboard [22,23]. In addition, the techniques do not satisfy mutual authentication because they are one-way authenticated, which does not satisfy the non-repudiation of the financial institutions for a user because the transaction history is stored by the financial institutions. Therefore, to solve the above problems, we propose a new transaction-linkage technique with which the designated terminal is combined, as the transactions are only approved for a designated terminal. The proposed technique only deals with transactions for the designated terminal registered by the user. This technique counteracts the issue when a device is stolen, supports non-repudiation by storing the transaction history in a transaction-linkage device, and provides mutual authentication. The proposed technique can therefore counteract most existing security threats by applying the above functions, thereby improving the security of online identification methods for Internet-banking services in 5G-enabled IoT.

Our key contributions are summarized below:

- The existing transaction linkage technology has a security threat of exposing the linkage code. In other words, the technology does not ensure its security because the code is input from the keyboard. On the other hand, in the proposed technique, the linkage code is generated from the transaction-linkage device without the input of the linkage code from the keyboard and is safe from the memory hacking attack by directly checking the transaction information requested by the user in the transaction-linkage device. Moreover, the proposed technique is robust against security threats because transactions are only approved for designated terminals.
- The proposed technique is safe from the eavesdropping attack, replay attack, spoofing attack, and denial-of-service attack. Moreover, its security is improved by providing mutual authentication that is not provided by existing Internet banking systems, and the non-repudiation is supported by storing the transaction history in the transaction-linkage device. Due to these security improvements, the proposed technique counteracts most security threats and improves the security of online identification mechanisms.

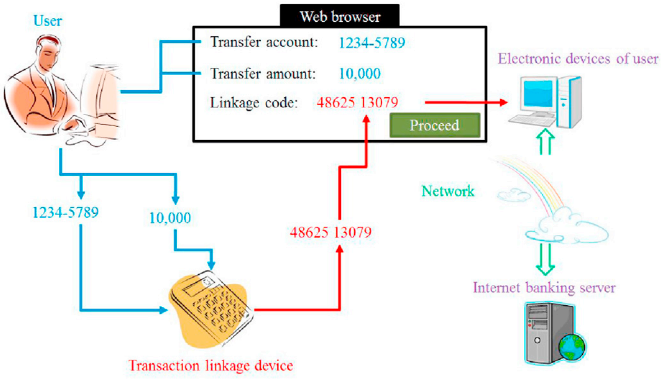

User
Web browser
Transfer account: 1234-5789
Transfer amount: 10,000
Linkage code: 48625 13079
Proceed
Electronic devices of user
Network
1234-5789
10,000
48625 13079
Internet banking server
Transaction linkage device

Fig. 2. Operational process of the transaction-linkage technique.

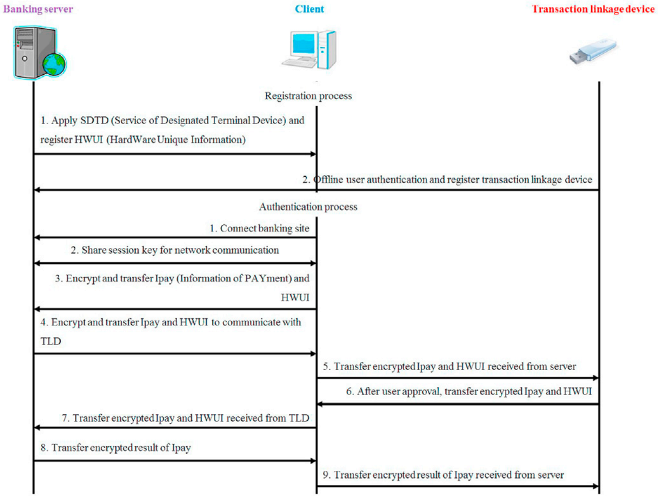

Banking server
Client
Transaction linkage device
Registration process
1. Apply SDTD (Service of Designated Terminal Device) and register HWUI (HardWare Unique Information)
2. Offline user authentication and register transaction linkage device
Authentication process
1. Connect banking site
2. Share session key for network communication
3. Encrypt and transfer Ipay (Information of PAYment) and HWUI
4. Encrypt and transfer Ipay and HWUI to communicate with TLD
5. Transfer encrypted Ipay and HWUI received from server
6. After user approval, transfer encrypted Ipay and HWUI
7. Transfer encrypted Ipay and HWUI received from TLD
8. Transfer encrypted result of Ipay
9. Transfer encrypted result of Ipay received from server

Fig. 3. The operational process of the proposed technique.

The rest of this paper is organized as follows. In Section 2, we describe the prior knowledge for the proposed method. In Section 3, we describe the proposed transaction linkage technique combined with the designated terminal, and presents a security assessment of the proposed method. Our conclusions and future work are in Section 4.

2. Prior knowledge

The transaction-linkage technique is shown in Fig. 2. When a user inputs the transaction information, such as the account number, the transfer amount, and so on, into the transaction-linkage device, the device displays the linkage code (verification code), the generation of which is based on the key shared between the Internet-banking server and the device. The user then inputs the displayed code into a web browser, and the code is subsequently transferred to the Internet-banking server [2]. The operational process of the proposed technique is shown in Fig. 3.

Step 1. In the registration process, the user applies for the SDTD (Service of Designated Terminal Device) to the financial institution and registers the HWUI (HardWare Unique Information) of the electronic device that the user wants to register for the transaction linkage.
Step 2. After applying for the SDTS, the user identities himself or herself through an offline authentication to visit the financial institution directly, and then obtains the transaction-linkage device after the offline authentication. The server and the transaction-linkage device share the seed value to generate an encryption/decryption key, and time synchronization is applied in this step.
Step 3. The user starts the financial transaction by accessing the financial transaction site in the authentication process.
Step 4. The user and the financial institution share a session key to establish a secure channel for network communication.
Step 5. The user sends the transfer information, which comprises the encrypted input transaction information and the HWUI of the designated device, to the server.
Step 6. The server sends the received encrypted transaction information and the HWUI based on the shared encryption key between the server and the transaction-linkage device to communicate with the device.
Step 7. The user authenticates the server based on the received information and sends the encrypted transaction information and the HWUI to the transaction-linkage device.
Step 8. The transaction-linkage device is displayed to an extra module such as an LCD panel for user recognition after decrypting the received transaction information, and the user approves the transaction after confirming whether the transaction information is correct or not. When the transaction is approved, the device sends the encrypted transaction information and the HWUI approved by the user to the server. If the transaction information is not correct, the mingling transaction information is filled with random information and then sent to the server to disturb the communication process.
Step 9. The user sends the received information from the device to the server directly.
Step 10. The server decrypts the received transaction information from the device and detects any manipulation by comparing the decrypted transaction information with the received transaction information from the user. If the compared result is correct, this transaction is properly approved, and the encrypted result, which is the processed transaction result, is then sent.
Step 11. The user sends the received transaction result to the transaction-linkage device.
Step 12. The transaction-linkage device displays the received decrypted transaction result, and when the user finally confirms the transaction result, the transaction result is stored inside the device for non-repudiation.

The server and the transaction-linkage device generate an encryption/decryption key based on the generated time stamp based on shared-seed value and time synchronization, and the generated key consists of the hash-chain type according to the time stamp to prevent encryption/decryption of the transaction information and the HWUI [24,25] that is based on the same encryption/decryption key. Moreover, the session key for the network communication in Step 4 is changed for every session to prevent the replay attack. A fixed password method, one of the knowledge-based identity-verification methods, is applied to the transaction-linkage device for the improvement of the security of the transaction. In addition, the transaction-linkage device can be applied flexibly to a variety of devices through wireless communication or through a connector that can be inserted into the PC or mobile device. In terms of the safety of the proposed technique, it is safe from debugging and the reverse-engineering attack because the transaction information is encrypted and decrypted in the server and the device by generating a key to the hash-chain type based on the shared-seed value and time stamp. For this reason, the information is safe during the sending and receiving processes between the network, server, host, and device. Moreover, the communication process between the host and the device is concealed from attackers. Because the communication process is of the one-sided transfer type, it does not constitute the challenge-response structure.

3. The proposed transaction linkage technique combined with the designated terminal

In this section, a secure protocol of transaction-linkage technique for which the designated terminal is combined is proposed, and the proposed protocol satisfies mutual authentication between the server and the transaction-linkage device during transactions. In the proposed protocol, the server, user, and transaction-linkage device communicate by wire or wireless means, especially, 5G network, and it is assumed that every communication channel is not safe. The transaction-linkage device is issued to the user after the user's identity is verified offline, and it is assumed that the time synchronization between the server and the transaction-linkage device is done during the offline authentication. It is also assumed that the environment allows the server and the user to share the session keys.

The server and the transaction-linkage device share the seed value after the offline authentication, and they then generate the encryption/decryption key. The server authenticates both the user and the transaction-linkage device based on the proposed protocol, and it performs a process to ensure that the transaction information is validated for both the user and the transaction-linkage device. At this time, the user sends the encrypted transaction information input by the user and the HWUI of the designated terminal to the server. Then the server sends the received encrypted information based on an initially generated key that is based on the shared-seed value and the time stamp between the server and the transaction-linkage device, but only when the received HWUI is compatible with the registered information. The device requests a confirmation of the transaction information by the user after decrypting the received information.

When the user approves the received transaction, the device sends the encrypted approved transaction information based on a second generated key that is based on the shared-seed value and the time stamp between the server and the device. The server approves the transaction when the received transaction information from the user is equal to the received transaction information from the device. Then the server sends the encrypted information of the transaction result based on a third generated key that is based on the shared-seed value and the time stamp between the server and the device. The device sends the received transaction result to the user and then stores it to prevent the non-repudiation of the server. Through the above process, the server satisfies the mutual authentication between the user and the transaction-linkage device and prevents the manipulation of the transaction information by third parties. Moreover, the proposed protocol is safe from the eavesdropping attack, replay attack, spoofing attack, and denial-of-service attack, and it satisfies anonymity and uniqueness because the encryption/decryption key is changed for each session and each transmission through time synchronization.

3.1. Security requirements

The security requirements for counteracting the security threats are described because the network and input/output devices such as USB are vulnerable to attacks from third parties. The security requirements are a defined condition for ensuring the security of mutual-authentication support and the protection against the eavesdropping attack, replay attack, spoofing attack, and asynchronous attack.

Mutual authentication is an authentication process to confirm the

Table 1
Terminology of the registration process.

Terminology	Description
BS	Banking Server
C	Client
TLD	Transaction-linkage Device for 5G-enabled IoT
ID	Identification
HWUI	HardWare Unique Information
SDTD	Service of Designated Terminal Device
SK	Shared Session Key between BS and C
SS	Shared-seed value between BS and TLD
M1	Encrypted result of ID and HWUI based on SK
E_K ()	Encryption operation based on key K
D_K ()	Decryption operation based on key K

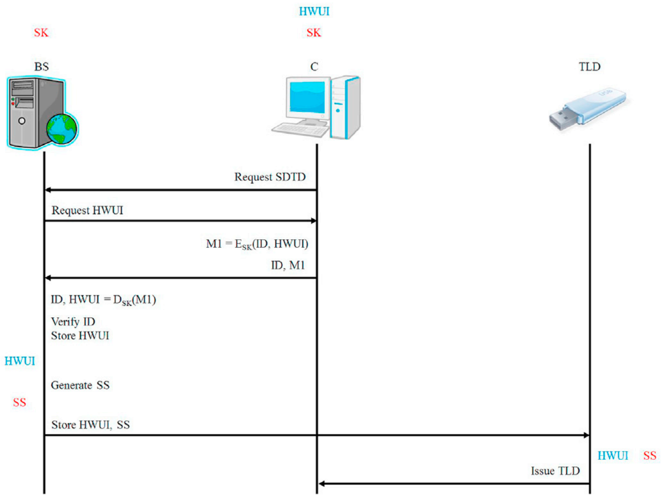


Fig. 4. Registration process of the proposed protocol.

legitimate entities for the server, user, and transaction-linkage device in Internet-banking services. This process authenticates entities by confirming the encryption/decryption result based on the shared value. The eavesdropping attack steals the information transferred between the server and the user and between the user and the transaction-linkage device to manipulate the transaction information. To counteract this attack, the attacker must be prevented from obtaining secret information or modified transaction information even though eavesdropping is utilized. The replay attack denotes that an unauthorized user requests service by reusing the information after stealing it from transfers between entities. If the replay attack is available, the attacker is authenticated as legitimate through using information stolen during a previous session. To counteract this attack, the transferred information must be meaningless when the session is changed. The spoofing attack denotes stealing information or deceiving users by pretending to be authorized devices, servers or users. This is possible when an attacker generates a correct response to a protocol challenge. In addition, in terms of the asynchronous attack, two entities, namely, the server and the transaction-linkage device, must synchronize to transfer the transaction information. If one entity does not send the information, the transaction will not be dealt with properly. For this reason, the attacker blocks the transfer of information between the server and the device and then tries an attack when the transaction is not processed properly; and this kind of attack is called the “asynchronous attack.” This type of attack is related to the denial-of-service attack, so if the information between the server and the device is not synchronized, the communication protocol needs a countermeasure to detect the presence of an attack.

Table 2
Terminology of the proposed protocol of the authentication process.
Step 1. C → BS: ID, M2

Terminology	Description
BS	Banking Server
C	Client
TLD	Transaction-linkage Device for 5G-enabled IoT
HWUI	HardWare Unique Information
I_{PAY}	Information of PAYment
I	Index
TSi	*i*-th Time Stamp
SK	Shared Session Key between BS and C
SS	Shared-seed value between BS and TLD
Ki	Generated *i*-th Key using HMAC based on TSi and SS
M2	Encrypted result of ID, I_{PAY}, and HWUI based on SK
M3	Encrypted result of ID, I_{PAY}, and HWUI based on Ki
M4	Encrypted result of ID based on SK
M5	Encrypted result of ID, I_{PAY}, and HWUI based on Ki+1
M6	Encrypted result of ID and I_{PAY} based on Ki+2
E_K ()	Encryption operation based on key K
D_K ()	Decryption operation based on key K
HMAC (A,B)	Hashed result of B based on key A

3.2. *Registration process*

It is assumed that the communication channels between the server and the user and between the user and the transaction-linkage device are not safe, and the server and the transaction-linkage device share the seed value after the offline authentication, and the server and the user share the session key. Table 1 denotes the terminology, and Fig. 4 shows the registration process of the proposed protocol.

Step 1. C → BS: request SDTD

C applies for an SDTD to the BS.

Step 2. BS → C: request HWUI

The BS requests the HWUI for the use of the terminal from C.

Step 3. C → BS: ID, M1

C sends the ID and the M1 of the ID and the HWUI based on the SK that is shared between the BS and C.

Step 4. ID, HWUI ← D_{SK}(M1)

The BS extracts the ID and the HWUI by decrypting the M1 received from C based on the SK between C and BS, and BS verifies the ID validation by comparing the received ID with the decrypted ID. When the ID verification is completed successfully, the HWUI is stored in the database to be utilized in the authentication process.

Step 5. Generate SS

The BS generates an SS to be shared with the TLD, and the BS then stores the generated SS and the received HWUI to the TLD for their issuance to C.

Step 6. Issue TLD

The BS issues the TLD to C, and the device has the SS to be shared with the TLD and the HWUI of the designated terminal from C.

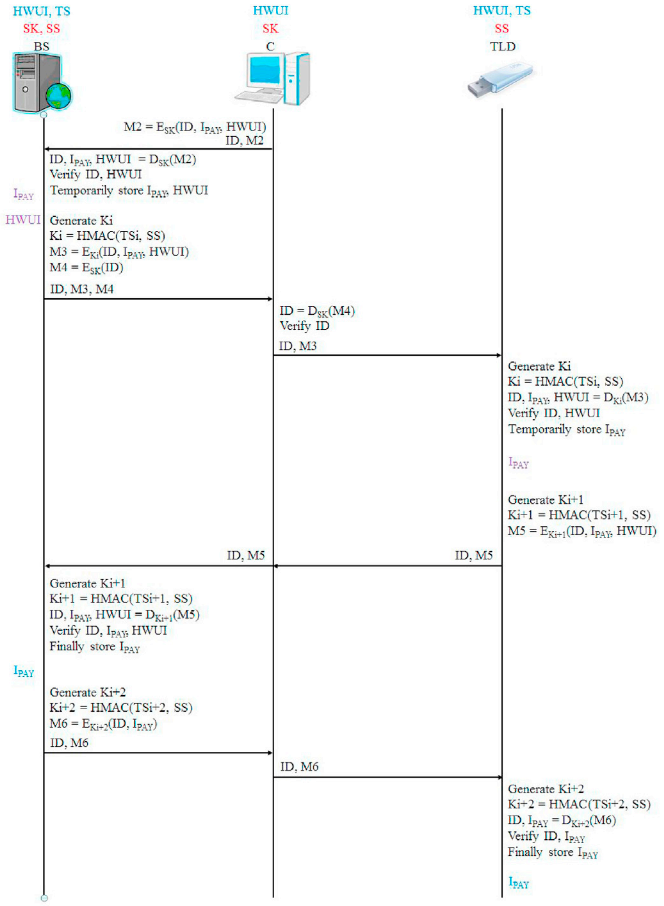


Fig. 5. Authentication process of the proposed protocol.

3.3. Authentication process

As with the registration process, it is assumed that the communication channels between the server and the user and between the user and the transaction-linkage device are not safe, and the server and the transaction-linkage device share the SS and the HWUI. Moreover, it is also assumed that the server and the transaction-linkage device synchronize the Time Stamp (TS) using time synchronization, and that the server and the user share the SK. Table 2 denotes the terminology, and Fig. 5 shows the authentication process of the proposed protocol.

C sends the ID and the M2 to the server. The M2 is the encrypted ID, I_{PAY}, and HWUI based on the SK between the BS and C.

Step 2. ID, I_{PAY}, HWUI ← D_{SK}(M2)

The BS extracts the ID, I_{PAY}, and HWUI by decrypting the M2 received from C based on the SK between the BS and C, and the BS then performs verifications by comparing the received ID and HWUI with the ID and HWUI stored in the database. When the ID and HWUI verifications are completed successfully, I_{PAY} is stored temporarily so that the transaction is processed soon.

Step 3. BS → C: ID, M3, M4

The BS generates the Ki using the HMAC operation based on the *i*-th synchronized TS and the SS between the BS and the TLD. After the ID, I_{PAY}, and HWUI are then encrypted as M3 based on the Ki, the ID is encrypted as M4 based on the shared SK between BS and C. Finally, the BS sends the ID and the encrypted results M3 and M4 to C.

Step 4. C → TLD: ID, M3

C extracts the ID by decrypting the M4 received from the BS based on the shared SK between the BS and C, and C verifies the ID validation by comparing the received ID with the decrypted ID. When the ID verification is completed successfully, C sends the ID and M3 to the TLD.

Step 5. ID, I_{PAY}, HWUI ← D_{Ki}(M3)

The TLD generates the Ki using the HMAC operation based on the *i*-th synchronized TS and the SS between TLD and the BS. After the TLD extracts the ID, I_{PAY}, and HWUI by decrypting the M3 received from C based on the Ki, the TLD verifies the ID and HWUI by comparing the received ID and HWUI with the decrypted ID and HWUI. After the ID and HWUI validations are completed successfully, the information is stored temporarily for further approved comparison, and the extracted I_{PAY} will be displayed to the extra-display device for recognition by C. If the user confirms that the transaction information is matched, he/she sends an approval regarding the result to the TLD by a direct action like the pushing of a button. For this step, if one of the previous steps has been manipulated, it is possible to be detected because the user confirms the input transaction information directly.

Step 6. TLD → C → BS: ID, M5

The TLD generates the Ki+1 using the HMAC operation based on the *i*+1-th synchronized TS and the SS between the TLD and the BS. After the TLD encrypts the M5, namely, the ID, the I_{PAY} approved by C and the HWUI based on the Ki+1, the M5 is sent to C. C also sends the M5 to the BS.

Step 7. ID, I_{PAY}, HWUI ← $D_{K\ i+1}$ (M5)

The BS generates the Ki+1 using the HMAC operation based on the *i*+1-th synchronized TS and the SS between the BS and the TLD. The BS extracts the ID, the I_{PAY} approved by the user, and the HWUI by decrypting the M5 received from C based on the Ki+1, and the BS verifies the ID validation by comparing the received ID with the decrypted ID. Moreover, the BS also verifies the I_{PAY} and HWUI by comparing the decrypted I_{PAY} and HWUI with the temporarily stored I_{PAY} and HWUI from Step 2. After the ID, I_{PAY}, and HWUI validations are completed successfully, the BS finally approves the transaction and stores transaction results to prevent C from rejecting the transaction.

Step 8. BS → C → TLD: ID, M6

The BS generates the Ki+2 using the HMAC operation based on the *i*+2-th synchronized TS and the SS between the BS and the TLD. The BS encrypts the M6, namely, the ID and the I_{PAY} approved by the BS based on Ki+2, and C sends the M6 to the TLD directly.

Step 9. ID, I_{PAY}← $D_{K\ i+2}$ (M6)

The TLD generates the Ki+2 using the HMAC operation based on the *i*+2-th synchronized TS and the SS between the TLD and the BS. The BS extracts the ID and the I_{PAY} approved by the BS by decrypting the M6 received from the BS based on the Ki+2, and the BS verifies the ID and I_{PAY} by comparing the received ID and the temporarily stored ID from Step 5 with the received I_{PAY} and the temporarily stored I_{PAY} from Step 5. When the ID and I_{PAY} validations are completed successfully, the TLD displays the I_{PAY} approved by the BS. For this step, the user can identify that the transaction is successfully approved. Moreover, the TLD stores the result of the transaction in the repository to prevent the BS from rejecting the transaction.

Table 3
Security comparison results.

Online identification mechanisms	Eavesdropping attack	Replay attack	Spoofing attack	Service denial attack
ID-Password	O	O	O	X
Image/Graphic	O	O	O	X
OTP	O	O	O	X
Certificate	O	O	O	X
Designated terminal	O	O	O	O
Transaction linkage	O	O	O	X
Proposed method	X	X	X	X

3.4. Security assessment

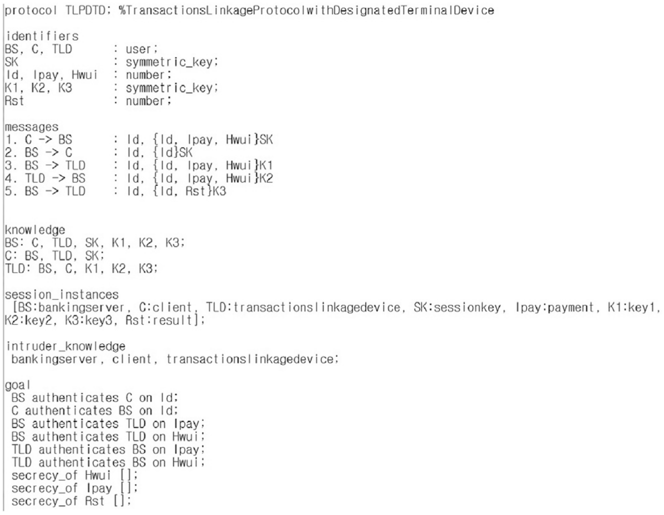
In this section, we verify the security through an analysis that satisfies the security requirements described in section 3.1. The proposed protocol supports mutual authentication and is not subject to eavesdropping attacks, replay attacks, spoofing attacks, and the service-denial attacks. In addition, the proposed protocol detects the transaction attempt of an undesignated terminal by verifying the HWUI of the designated terminal. The security of the proposed protocol is verified by satisfying the security requirements for which the AVISPA (Automated Validation of Internet Security Protocols and Applications) is used as the formal verification tool [26].

Mutual authentication is an authentication process wherein all entities involved in the communication are legitimate to each other. The proposed protocol authenticates the server to the user with the M2 message, and the user is authenticated to the server with the M4 message. Moreover, the server authenticates the transaction-linkage device with the M5 message, and the transaction-linkage device authenticates the server with the M6 message. In the communication process, the I_{PAY} and the HWUI are not exposed directly, and only the authorized users can obtain the I_{PAY} and the HWUI by using the generated key based on the SS and the TS. The server and the TLD encrypt and decrypt the transferred data based on Ki, Ki+1, and Ki+2 using the shared TS and the SS between the server and the TLD, and the server and the user encrypt and decrypt the transferred data based on the SK.

The eavesdropping attack occurs when an attacker steals the transaction-related information by eavesdropping as it is being transferred between each entity. The proposed protocol does not obtain the SK, SS, TSi, TSi+1, TSi+2, Ki, Ki+1, and Ki+2, because the attacker can only obtain the ID, M1, M2, M3, M4, M5, and M6. Therefore, the attacker cannot obtain the I_{PAY} and the HWUI during the network communication. If the SK is stolen on the user's terminal by reverse engineering, the I_{PAY} and the HWUI are then exposed to or manipulated by users, and the attacker does not obtain the Ki, Ki+1, and Ki+2 through the eavesdropping attack between the user and the transaction-linkage device. For this reason, any of the attacks related to transactions fail when only manipulated information is used. Consequently, the proposed protocol is safe from the eavesdropping attack.

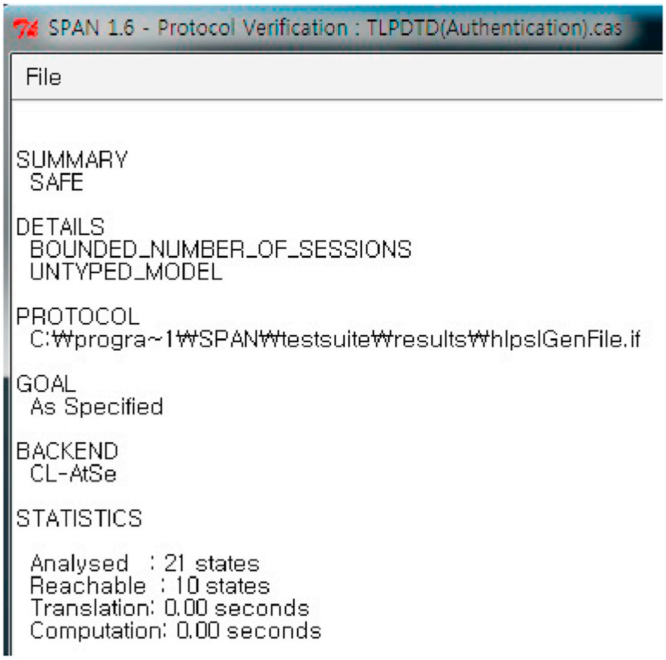
The replay attack occurs when an attacker detects the transferred information between each entity during the current session, and the information is then reused in the next session to approve a transaction successfully. The proposed protocol is able to sniff the ID, M1, M2, M3, M4, M5, and M6, but the attacker can not be authenticated as an authorized user by replaying the message. This is because the messages without the ID are encrypted using the Ki, Ki+1, and Ki+2 and the SK based on the SS, TSi, TSi+1, and TSi+2. The SK is generated every session, and the Ki, Ki+1, and Ki+2 are generated with every TS for every session. Therefore, the proposed protocol is safe from the replay attack because the attacker is detected when he/she tries to enact the replay attack.

The spoofing attack occurs when an attacker steals the authentication-related information by deceiving legitimate entities. To deceive the legitimate entities, the attacker has to calculate the M2, M3, M4, M5, and M6, but he/she does not get the Ki, Ki+1, and Ki+2. For this reason, the attacker does not generate the above authentication-related information, so the proposed protocol is safe from the spoofing attack.



```
protocol TLPDTD: %TransactionsLinkageProtocolwithDesignatedTerminalDevice

identifiers
BS, C, TLD       : user;
SK               : symmetric_key;
Id, Ipay, Hwui   : number;
K1, K2, K3       : symmetric_key;
Rst              : number;

messages
1. C -> BS       : Id, {Id, Ipay, Hwui}SK
2. BS -> C       : Id, {Id}SK
3. BS -> TLD     : Id, {Id, Ipay, Hwui}K1
4. TLD -> BS     : Id, {Id, Ipay, Hwui}K2
5. BS -> TLD     : Id, {Id, Rst}K3


knowledge
BS: C, TLD, SK, K1, K2, K3;
C: BS, TLD, SK;
TLD: BS, C, K1, K2, K3;

session_instances
 [BS:bankingserver, C:client, TLD:transactionslinkagedevice, SK:sessionkey, Ipay:payment, K1:key1,
K2:key2, K3:key3, Rst:result];

intruder_knowledge
 bankingserver, client, transactionslinkagedevice;

goal
 BS authenticates C on Id;
 C authenticates BS on Id;
 BS authenticates TLD on Ipay;
 BS authenticates TLD on Hwui;
 TLD authenticates BS on Ipay;
 TLD authenticates BS on Hwui;
 secrecy_of Hwui [];
 secrecy_of Ipay [];
 secrecy_of Rst [];
```

Fig. 6. Verification code of the applied AVISPA for the proposed protocol.



SPAN 1.6 - Protocol Verification : TLPDTD(Authentication).cas
File

```
SUMMARY
 SAFE

DETAILS
 BOUNDED_NUMBER_OF_SESSIONS
 UNTYPED_MODEL

PROTOCOL
 C:₩progra~1₩SPAN₩testsuite₩results₩hlpslGenFile.if

GOAL
 As Specified

BACKEND
 CL-AtSe

STATISTICS

 Analysed   : 21 states
 Reachable  : 10 states
 Translation: 0.00 seconds
 Computation: 0.00 seconds
```

Fig. 7. Assessment result for the security of the proposed protocol according to the AVISPA.

The service-denial attack occurs when an attacker disrupts the synchronization of the information and enacts an asynchronous transmission by inducing inconsistencies that block the transmission of information between each entity. Regarding the proposed protocol, the attacker has to obtain the information transferred to the TLD or disrupt the synchronization of information to try an asynchronous transmission, but the server and the device synchronize by time. Therefore, the proposed protocol is safe from service-denial attacks. Table 3 shows the results of security comparison between the proposed technique and some existing online identification mechanisms.

The designated-terminal authentication blocks any transaction attempts from an undesignated terminal by verifying the HWUI on the designated terminal from the registration process with the HWUI in the authentication process. In the proposed protocol, the server authenticates the designated terminal using the M2 message, and the transaction-linkage device authenticates the designated terminal using the M3 message. Therefore, the server and the TLD authenticate the designated terminal using the above messages.

Finally, we describe the protocol-verification result for which the AVISPA is used as a formal verification tool. The AVISPA assesses security by deriving the possible security threats. Fig. 6 shows the verification code applied by the proposed protocol, and Fig. 7 shows the verification result. A is shown SAFE is displayed in the SUMMARY, meaning that the proposed protocol is safe.

4. Conclusions

A designated PC service was adapted to restrict a terminal from using a service when the identity-verification methods supporting the existing Internet-banking services were under security threats. Nevertheless, the designated PC service did not pass the security assessment and did not define the evaluation criteria, so the service was still exposed to security threats. The existing transaction-linkage technique generated a linkage code by combining the transaction information with the secret information to counteract this problem. However, this technique was also exposed to various security threats. For these reasons, the current designated-PC service and transaction-linkage technique do not ensure security, so we proposed a transaction-linkage technique for which the designated terminal is combined to solve those problems. The proposed technique in this paper is capable of counteracting all of the mentioned security threats, thereby improving the online identity verification methods. We believe that the security of the Internet-banking services will be robustly supported by the application of the proposed protocol.

Declaration of competing interest

The authors declare no conflict of interest.

Acknowledgements

This work was partially supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2018R1A4A1025632) and the Soonchunhyang University Research Fund.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.dcan.2020.12.003.

Disclosure

A part of this paper was presented at a Conference on International Conference on Big Data Technologies and Applications (BDTA), November 23–24, 2017, Gwangju, South Korea.

References

[1] M. Jun, S. Cai, The key determinants of internet banking service quality: a content analysis, Int. J. Bank Market. 19 (7) (2001) 276–291.
[2] A. Hiltgen, T. Kramp, T. Weigold, Secure internet banking authentication, IEEE Secur. Privacy 4 (2) (2006) 21–29.
[3] H. Kim, J. H. Huh, R. Anderson, On the Security of Internet Banking in south korea.
[4] K. Lee, K. Yim, M.A. Mikki, A secure framework of the surveillance video network integrating heterogeneous video formats and protocols, Comput. Math. Appl. 63 (2) (2012) 525–535.
[5] M. Zviran, W.J. Haga, Cognitive passwords: the key to easy access control, Comput. Secur. 9 (8) (1990) 723–736.
[6] X. Suo, Y. Zhu, G.S. Owen, Graphical passwords: a survey, in: 21st Annual Computer Security Applications Conference (ACSAC'05), IEEE, 2005, pp. 10–pp.
[7] P. Hoyer, Otp and challenge/response algorithms for financial and e-government identity assurance: current landscape and trends, in: ISSE 2008 Securing Electronic Business Processes, Springer, 2009, pp. 281–290.
[8] S. Kiljan, K. Simoens, D.D. Cock, M.V. Eekelen, H. Vranken, A survey of authentication and communications security in online banking, ACM Comput. Surv. 49 (4) (2017) 61.
[9] A. Ometov, V. Petrov, S. Bezzateev, S. Andreev, Y. Koucheryavy, M. Gerla, Challenges of multi-factor authentication for securing advanced iot applications, IEEE Network 33 (2) (2019) 82–88.
[10] K. Lee, K. Yim, A guideline for the fixed pc solution, in: Proc. Of the 2012 International Conference on Smart Convergence Technologies and Applications (SCTA'12), 2012, pp. 74–76. Gwangju, Korea, August.
[11] H.-N. You, J.-S. Lee, J.-J. Kim, M.-S. Jun, A study on the two-channel authentication method which provides two-way authentication in the internet banking environment, in: 5th International Conference on Computer Sciences and Convergence Information Technology, IEEE, 2010, pp. 539–543.
[12] L. Ogiela, M.R. Ogiela, Bio-inspired cryptographic techniques in information management applications, in: 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA), IEEE, 2016, pp. 1059–1063.
[13] L. Ogiela, M.R. Ogiela, U. Ogiela, Efficiency of strategic data sharing and management protocols, in: 2016 10th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), IEEE, 2016, pp. 198–201.
[14] A. Kesharwani, S. Singh Bisht, The impact of trust and perceived risk on internet banking adoption in India: an extension of technology acceptance model, Int. J. Bank Market. 30 (4) (2012) 303–322.
[15] This poodle bites: exploiting the ssl 3.0 fallback. https://www.openssl.org/~bodo/ssl-poodle.pdf.
[16] K. Lee, K. Yim, Keyboard security: a technological review, in: 2011 Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IEEE, 2011, pp. 9–15.
[17] Here come the ninjas. http://netifera.com/research/beast/beast_DRAFT_0621.pdf.
[18] Vulnerability summary for cve-2011-3389. https://nvd.nist.gov/vuln/detail/CVE-2011-3389.
[19] N. AlFardan, D.J. Bernstein, K.G. Paterson, B. Poettering, J.C. Schuldt, On the security of rc4 in {TLS}, in: Presented as Part of the 22nd {USENIX} Security Symposium ({USENIX} Security 13), 2013, pp. 305–320.
[20] Beast followup. https://www.imperialviolet.org/2012/01/15/beastfollowup.html.
[21] Is beast still a threat?. https://blog.qualys.com/ssllabs/2013/09/10/is-beast-still-a-threat.
[22] K. Lee, K. Bae, K. Yim, Hardware approach to solving password exposure problem through keyboard sniff, Interfaces 1 (2009) 2.
[23] K. Lee, W. Kim, K. Bae, K. Yim, A solution to protecting usb keyboard data, in: 2010 International Conference on Broadband, Wireless Computing, Communication and Applications, IEEE, 2010, pp. 108–111.
[24] K. Lee, K. Lee, J. Byun, S. Lee, H. Ahn, K. Yim, Extraction of platform-unique information as an identifier, JoWUA 3 (4) (2012) 85–99.
[25] K. Lee, H. Yeuk, K. Yim, S. Kim, Analysis on manipulation of the mac address and consequent security threats, in: Proceedings of the 8th ACM CCS International Workshop on Managing Insider Security Threats, ACM, 2016, pp. 113–117.
[26] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuéllar, P.H. Drielsma, P.-C. Héam, O. Kouchnarenko, J. Mantovani, et al., The avispa tool for the automated validation of internet security protocols and applications, in: International Conference on Computer Aided Verification, Springer, 2005, pp. 281–285.