IJNS
International Journal of Network Security

ISSN 1816-353X (Print)
ISSN 1816-3548 (Online)

Vol. 23, No. 2 (March 2021)

INTERNATIONAL JOURNAL OF NETWORK SECURITY

PUBLISHING OFFICE

Min-Shiang Hwang
Department of Computer Science & Information Engineering, Asia University, Taichung 41354, Taiwan, R.O.C.
Email: mshwang@asia.edu.tw

International Journal of Network Security is published both in traditional paper form (ISSN 1816-353X) and in Internet (ISSN 1816-3548) at http://ijns.jalaxy.com.tw

PUBLISHER: Candy C. H. Lin

Volume: 23, No: 2 (March 1, 2021)

International Journal of Network Security

Research on Private and Seamless Roaming Cloud Service Authentications

Hsieh-Tsen Pan[1], Li-Chin Huang[2], and Min-Shiang Hwang[1,3]
(Corresponding author: Min-Shiang Hwang)

Department of Computer Science & Information Engineering, Asia University[1]
500, Lioufeng Rd., Wufeng, Taichung 41354, Taichung, Taiwan, R.O.C.
Department of Information Management, Executive Yuan, Taipei 10058, Taiwan[2]
Department of Medical Research, China Medical University Hospital, China Medical University, Taiwan[3]
(Email: mshwang@asia.edu.tw)
(Received Apr. 13, 2020; Revised and Accepted Dec. 24, 2020; First Online Feb. 20, 2021)

Abstract

With the rapid development of information technology and the popularization of mobile communication devices, we can connect to the Internet through various media, such as general broadband networks, WiFi, and 5G. However, each medium has its own authentication system and protocol, so it is difficult for users to achieve seamless migration when roaming. When users roam in such a network environment to provide seamless transmission services, they must first face data and network technology integration. IETF stipulates that mobile IP and various heterogeneous networks use IP to solve this problem; however, in the current network environment, each network medium has its own authentication mechanism, thereby forming users with multiple sets of accounts and passwords. Therefore, when the user roams to other networks, let the authentication center confirm its identity. At the same time, the user can protect his access to the service when accessing the service. Privacy is the focus of research; in path exploration, the process of information transmission can easily be eavesdropped on and intercepted. Therefore, how to realize the security of information transmission is also a major issue. Moreover, mobile devices are limited by power and computing capabilities. The high computational security mechanisms used in data transmission or access services are inappropriate. Therefore, this research will use the mobile IP protocol to assist users in roaming in different network management system mechanisms and add an AAA server for authorization, authentication, accounting, and key management. It is used in conjunction with the identity-based identity code password system. It is a security mechanism suitable for roaming and identity verification in a network environment.

Keywords: Accounting Seamless Roaming; Authentication; Authorization; Cloud Computing; Identity-based Cryptosystem; Privacy Protection

1 Introduction

Due to the development of information and internet technologies, cloud computing has been widely discussed and applied in recent years. Cloud computing allows different computers on the network to do one thing for you simultaneously, greatly increasing the processing speed [17]. Active processing through the cloud computing framework will greatly reduce costs and achieve better results. And because this technology will obviously provide consumers with a complete service and environment, cloud computing has been regarded as an important business opportunity for the next wave of technology industry after Web 2.0.

In today's diversified network services, uses will connect to the Internet anytime and anywhere through different network types to obtain the information they need. This is actually the original intention of the concept of pervasive computing: Any device can be used anytime, anywhere to obtain any information needed. The so-called ubiquity means that it can use at any time, any place, any place, anyone. In 1991, Mark Weiser, a computer scientist at Xerox Laboratories, first proposed the concept of ubiquitous computing, which pointed out that computers or terminal devices can be connected to the Internet anywhere, thereby realizing an information society that can be connected anywhere. This means that we can use future information technology at any time, anywhere, and anywhere, and has the following three characteristics, including the embedding of computing devices into every object and place in people's daily lives. In the future, computing devices will have smart interfaces, making it easier for users to use; connecting to networks through various computing devices will allow users to access the information they need at any time and place. In short, this is a human-centered environment where users can use any device (whether it is a PDA, mobile phone, notebook computer, or any future mobile device) under safe condi-

tions, get any information or services they need.

Ubiquitous computing can connect many different networks to form a network that integrates wired and wireless communications. These networks may include ordinary telephone networks (PSTN), the Internet and asynchronous transmission technology (ATM), wireless networks, fifth-generation mobile communications (5G), adhoc networks and wireless sensor networks, etc. Based on data and voice integration, any information and services can be transmitted through the public platform. In short, in the future, mobile devices can be converted into computers, ATM cards, radios, TVs, medical diagnostic programs, crisis warning devices, etc., in addition to being used as phones. Besides, with the development of networked household appliances, household appliances used independently in the past can be connected with the Internet's development. Even if you are not there, you can easily operate household appliances from a remote location. Through the interconnection of household appliances, computers, mobile phones, etc., the ideal state of energy saving, high efficiency, safety assurance, and easy operation can be realized.

2 Research Motivations

For users roaming on different networks, foreign domain resources are bound to be frequently used. The IETF (Internet Engineering Task Force) defines the mobile IP protocol to help users roam in different administrative domains so that IP has mobility. There are two versions of Mobile IP: Mobile IPv4 and Mobile IPv6. Because IPv6 has the above-mentioned advantages, using Mobile IPv6 for related research and system development will improve the network's communication quality and provide higher security. Besides, it can also establish a good connection between the network, application services, and devices. Anyone can use the Internet as a connection anywhere to share information in the ubiquitous Internet society free.

When we use mobile IPv6 and AAA architecture to solve roaming problems in various network environments, the following security issues should be dealt with together:

Motivation 1. User Authentiocation and Roaming

Generally, in accessing the service, the service provider first confirms the user's identity and whether the user has the authorization to access the service. However, traditional security is based on user authentication schemes where users successfully register in advance and obtain access rights. In short, the user has obtained the session key of the service provider in advance to access the service. However, it is impossible to obtain the session key required to access the service in advance through the traditional authentication scheme in a diversified network media environment. The main reason is that it is composed of multiple different networks. When accessing services for different networks, the network's authentication center still needs to perform identity verification and authorization.

Besides, users can request access to multiple services from multiple service providers. In this case, even if the security domains are different, users need to establish session keys and multi-factor authentication with multiple service providers, which will cause many problems in key authentication. For example, during the authentication and session process, the attacker is attacked by a malicious attacker, or the attacker forges the service provider's identity. Therefore, it can increase user privacy—sexual contact crisis. Therefore, many researchers have proposed authentication schemes for these problems, such as public key infrastructure (PKI) [1] based on asymmetric cryptography. Besides, mobile users can also use ticket-based access services in wireless communications [2, 3] when a user moves from a certain base station to another base station that does not belong to the user. It is assumed that the base station has established a mutual trust agreement in advance. The user can use a ticket-based authentication scheme to implement actions. However, there are still some unresolved problems in the above-mentioned schemes. For example, there will be a public key management problem in the public key infrastructure; in the ticket-based access service scheme, the difference between the requested service date and the required service still cannot provide an effective solution.

When a user roams to another network to request a service, It must reconfirm the user's identity and authority through the authentication center that requested the service. Therefore, how to achieve authentication and authorization between different networks will be a problem to be solved in this environment.

Roaming in this environment can cause many problems. For example, since there are multiple network media, users can connect to the network differently. However, each network medium can be composed of networks. Therefore, each network has its own authentication mechanism, which causes users to have multiple sets of identities and passwords, which also causes management problems; or when users roam to other networks, how to let the identity verification center confirm the user, while also allowing the user to access protect their privacy while serving. Besides, since users obtain services by roaming on mobile devices, and the mobile devices themselves have functions and computing capabilities, implementing roaming access services under such restrictions will be a big test. Regarding mobile management, scholars such as Fogelstroem *et al.* proposed Regional Registration [4] and Layered Foreign Agents [5] to solve when users move in the same non-local domain, there is no need to repeat the problem of the domain. Therefore, this research topic will be based

on combining mobile IPv6 and AAA, referring to the research on security issues such as identity verification [6–8], and proposing a fast scheme to establish a safer and more efficient integrated environment.

Motivation 2. Private Issues

In the Internet environment, the most worrying thing is privacy and security. Especially when information technology penetrates people's lives, personal privacy and security will become more important. Since many network environments are open environments, when many different network domains share information, protecting users' privacy is of utmost importance. Therefore, in such an environment, privacy research is the most important thing: To achieve privacy protection through authentication mechanisms under different usage domains and service choices to expand the availability of services and allow consumption. The increase of users' willingness to use will increase the income of service providers or network operators. At the same time, it can also protect personal identity and important information so as not to harm consumers' rights and interests.

However, in a seamless roaming network, the following two privacy issues are required:

1) IT should not expose the user's identity during the communication between the user and the service provider, nor should it be eavesdropped on by a third party. When users access different services, there should not be any association between the services in the user's information flow to prevent the association between the service and the service from becoming a channel to expose the user's identity.
2) The user's service content and other additional information (i.e., location, usage time, or service request type, etc.) must also be kept private. Therefore, the research topic aims at the above-mentioned privacy and proposes to protect the anonymity of users and the privacy of service content through blind signature technology and elliptic curve cryptosystem.

Motivation 3. Data Transmission Security

Nowadays, wireless networks are widely used by users due to their convenience. Users can move freely within the range of Access Points and communicate with other nodes or networks at any time. However, due to wireless networks' characteristics, many data are vulnerable to malicious attacks, such as packet modification attacks, packet forgery attacks, and denial of service attacks. If some malicious nodes abuse these packet attacks, they will pose a considerable threat to the entire network and cause network functions to fail. Regarding the wireless network problem, many scholars have also proposed many schemes [9–11]. In 2002, Guerrero-Zapata and Asokan [12] proposed a secure routing method based on AODV [13]. This scheme can prevent forgery attacks but cannot achieve the confidentiality and integrity of user data. Besides, because the data that smartphones can use is more restricted than other mobile devices (such as notebooks, etc.), smartphones have developed rapidly in the past few years. Therefore, in the future, we should focus on the research and design of a low-computing secure routing and data transmission protocol and solve the security problems of routing and key agreement by integrating secure routing and key agreement to realize wireless network.

3 Research Issues

This research uses identity-based cryptography as the main application. One of its greatest advantages is that the trusted key generation center in the system generates the corresponding private key based on the individual's identity information. Therefore, other users can directly use it without additional verification of the relationship between the public key and the user. Besides, it can simplify key management's difficulty and reduce the cost of traditional public key management. Besides, due to the limited capabilities and computing power of mobile devices in the ubiquitous computing environment, the advantage of elliptic curve cryptography is that it achieves the same security strength with a smaller key length has received widespread attention. Therefore, this research will abandon the traditional PKI encryption and decryption security technology and use the bilinear pairing characteristics of elliptic curve cipher and elliptic curve to design some mechanisms and apply them identity authentication and group secure communication. It will increase the security requirements for networks, communications, and information.

This research will be divided into two research topics. The first research topic will address the above-mentioned authentication problems in a seamless network roaming environment, propose using identity-based encryption technology in combination with NAI (Network Access Identifier), and add an AAA server to achieve authentication, authorization, and accounting. The second research topic will solve the security problem of seamless roaming network access services and propose privacy mechanisms to protect personal access services.

3.1 Research on Service Access and Authentication in a Seamless Roaming

In a society with advanced information technology, everyone is surrounded by many account numbers and passwords. Whether it is a financial card, access control system, or information system, a person has several sets of account numbers and passwords, which can also cause management problems. Therefore, scholars have pro-

posed a single sign-on (SSO) authentication mechanism to solve the inconvenience of remembering the array's account and password.

However, in addition to the above-mentioned single sign-on authentication mechanism, there are still ticket-based mechanisms [2, 3] to reduce users' need to repeatedly enter account numbers and passwords to access the same system service multiple times. In addition to solving the inconvenience caused by repeated personal logins, another more important issue is how to realize the exchange of personal and independent data through effective electronic operations when personal information is distributed among different independent organizations. The unit's identity verification, the confidentiality of data transmission, the integrity of the data itself, and the information system's access control rights.

The ticket-based authentication mechanism in the traditional user authentication system usually uses the user's account and password for simple verification. For example, in a Unix system, the user's verification table is stored in the system, and it compares the user input information with the verification table in the system. To prevent the user password from being stored in plain text format, it can undergo certain functional operations (such as a secure uplink hash function) and then store the calculated hash value in the verification table. However, in a complex environment, users may have to use multiple devices or workstations, which will cause inconvenience to administrators in managing users. The user needs to remember a different account and password for each server. When you want to change the password, you must change all the information. Therefore, the concept of a central authentication server is generated. All user accounts and passwords are stored in this authentication server to achieve single sign-on with one account per person.

The ticket-based authentication scheme is mainly used in distributed systems. The authenticated user may not have the right to use it on some servers, or when there is no personal account and password stored in the server, the third ticket-granting server is recommended to establish a mutual trust relationship with the server in the following locations Issue authorization tickets in advance and from a fair perspective. When the server receives the authorization ticket and verifies that it belongs to an impartial third-party ticket issuing organization, it can accept user login. However, it gets ticket-based access services for mobile users in wireless communications [2, 3]. When a user moves from a base station to another base station that does not belong to the user, it is assumed that a pre-established mutual trust agreement has been established between the base stations. The user can use the ticket-based authentication scheme to achieve mutual trust between each other. The authorization between actions facilitates the resolution of accounting and cost-sharing ratio issues between each other.

Currently, most users use the Kerberos architecture to enable users to access services provided by other servers on the network [14–16]. The user authentication and service request process can be divided into three stages. The first step is the message exchange of the authentication service. The main purpose is to enable the user to communicate with the authentication server (AS) to request to log in to the ticket-granting server (TGS); the second stage is the information exchange of the ticket approval service, and the main purpose is to obtain the ticket for service authorization. The third stage is the message exchange stage of mutual confirmation between the client and the server. The main purpose of the message exchange stage is to obtain server services.

If there is no authentication in Kerberos, there may be the following three types of attacks:

1) By gaining access to a specific server, a user can pretend to be another service user.

2) The user can change the server's network address so that the server's request after this change is mistaken for a fake server.

3) Users can exchange information through eavesdropping and gain access to the server through retransmission attacks or disrupt the server's operation.

In response to the above attacks, Kerberos has adopted three security measures:

1) Trust each client-server, ensure its user identity, and trust that each server will implement security policies based on user identity.

2) The client system needs to verify its identity with the server, but the client system user's relevant identity is trustworthy.

3) The user must verify each related service's identity; the server needs to verify its identity to the client.

Although some attack problems have been solved, there are still two authentication problems:

1) If the service date is different, the user must re-enter the password. For example, the user logs in to the mail server in the morning and enters the password once; he wants to check yesterday's mail and must re-enter the password. Besides, the more times the password is entered, the higher the risk.

2) When users request other services, they need to apply for new tickets.

In the first research focus, we will focus on the identity-based encryption system to achieve the security of authentication and authorization in a single domain (intra-domain) service access and solve the above-mentioned Kerberos persistence. We use identity-based encryption technology to solve Kerberos's process and establish a conference key in advance and transmit it through a secure channel. Besides, identity-based cryptography can also solve the huge storage space and computational cost

of traditional public-key cryptosystems, especially for mobile devices that run in ubiquitous computing and effectively reduce their power consumption.

Besides, we also use the AAA server in this field. It uses an identity-based cryptographic system to achieve authentication security [18, 19]. Therefore, we will use the AAA server for the functions of the Kerberos authentication center. The ticket authorization server is integrated, thereby reducing the steps of accessing the Kerberos service. Because the AAA server has authorization, authentication, and accounting, it can also further function of extending the key management function to use the AAA server to process the identity verification, authorization, and accounting services related to the subscribers of the heterogeneous access system. And then integrate the heterogeneous access system. A trustworthy universal communication platform between the system provides interconnection and mutual control security between access networks, effective wireless resource management, and mobility management. In addition to service quality control and coordination, it is also used as a connection interface for accessing different networks.

On the user side, it provides an IP-based mechanism with a single number and a single account. Each user uses the NAI (Network Access Identifier) [20] identification format, for example, user@realm, so that the AAA server can easily identify the user's home domain for identification; if it is a mobile phone user, use its IMSI (International Mobile Subscriber Identifier) to identify it, and it includes information in the NAI information. NAI represents each user's public and uniquely identifiable information as to their public key in the IBE scheme. In this way, it simples the difficulty of key management. And it reduces the cost of traditional public key management. At the same time, it can ensure the security of authorization, authentication, and accounting to prevent attacks by malicious attackers. Thise research topic will integrate AAA servers and identity-based encryption technology to provide a secure service access mechanism in pervasive computing.

3.2 Research on Privacy Protection in Seamless Roaming

Privacy and security are important issues in public networks. On the one hand, service providers must verify users' legitimacy and ensure that they can access the legally authorized services. On the other hand, users must maintain their necessary privacy, avoid tracking when and where they access the service, use the service, etc. Therefore, the research focus will focus on the seamless roaming network environment and propose an effective authentication mechanism and protect user privacy. In a seamless roaming network environment, users can roam at will, leading to important security issues, including how the identity verification center and service providers verify the user's identity and access rights? How users issue certificates so that network authentication centers roamed in the past can verify their identities, etc. This is another important issue that is also very important in the seamless roaming network. This problem is the privacy of users. Due to the rapid development of computer information technology and the Internet, it has become easier to obtain and infringe personal information. Therefore, everyone has gradually shifted the traditional concept of privacy. In this era, personal privacy has been unconsciously violated. How does the certification authority verify that the user's identity is legal? At the same time, users can protect their privacy when accessing the service. Therefore, it needs to compare services and certification centers to meet the anonymity of users' personal data, aliases, and the requirements of different environments or different combinations of services. We have classified the types of privacy in the future pervasive network, and the goals we hope to achieve are defined as follows:

Anonymity: User identity should not be exposed or eavesdropped on by a third party during the communication between the user and the service provider. When users access different services, there should not be any association between the services in the user's information flow to prevent the association between the service and the service from becoming a channel to expose the user's identity. Therefore, we will use blind signature technology and elliptic curve, and other cryptographic technologies to effectively protect users' anonymity.

Context Privacy: The content of the service used by the user and other additional information (such as location, time of use, or type of service request, etc.) must be kept private. Therefore, in this research, the privacy of users who use the service content will also be protected. In this part, we will use the bilinear pairing function for encryption and decryption and use this method to allow users to use the service while maintaining the service content's privacy.

Confidentiality and Integrity: The interaction between the user and the service must maintain its confidentiality and integrity. Therefore, in addition to using elliptic curve bilinear pairing for encryption and decryption, to ensure the message's integrity during transmission, we will also add a hash function or digital signature technology to maintain the integrity of the server and the message—security during service communication.

Due to mobile devices' limited capabilities and computing power, the use of elliptic curve cryptography has attracted widespread attention and expectations in recent years because its advantage is that it can achieve the same security strength with a smaller key length. Therefore, we will use the bilinear pairing characteristics of elliptic curve cryptography and elliptic curve to design some schemes and apply them to identity authentication and group secure communication to increase the network, communication, and information security requirements.

We have constructed three entities in the network environment; their identities are the authentication server (AAA server), the service provider (SP), and the user. Below is a brief description of the tasks and functions provided by these three entities.

Authentication Server: Its main function is to authenticate users in the same network and issue relevant certificates to users after authentication. The user can use this certificate to check whether the service provider wants to access the service.

Service Provider: Responsible for providing services and granting access to related services according to the user's authority level who wants to access the service.

User: The user who wants to access the service.

Figure 1 shows the roaming authentication security relationship. When a user is on the same network, he/she must first perform identity authentication on the authentication server and obtain a legal certificate and other related data before requesting services from a service provider on the same network. In the same network, users will have SA security identity verification relationships between different individuals (SA2, SA3, SA5). Besides, due to the need to provide a seamless roaming network, many networks need to be considered. The services in each network are different. Therefore, when the user roams to other networks, the SA will be generated in the different networks' authentication servers. So in the second research focus, we will explore the problem of secure identity verification in the seamless roaming network and at the same time solve the problem of secure communication.

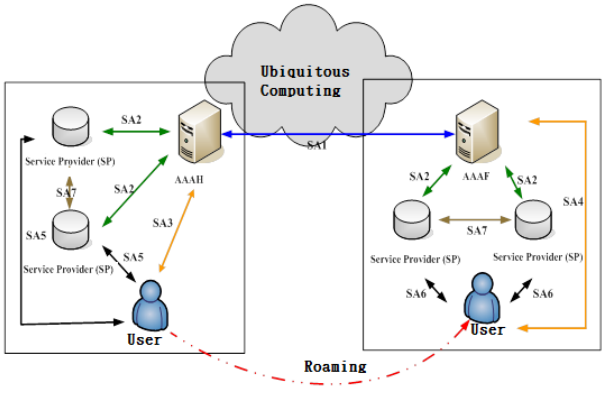


Figure 1: The roaming authentication security relationship

There are two categories in Figure 1: Pre-established security relationships and dynamically established security relationships. Pre-establishment is a security relationship that must be established based on specific interests when building a system. Besides, dynamic establishment means that users must provide correct credentials to establish a secure relationship, and the security relationship has a life cycle. The detailed security relationship is as follows:

Pre-established.

SA1: The security relationship between AAA servers in different domains requires exchanging information, such as authorization, identity verification, and accounting. In the SA1 security issue, we aim to simplify the authentication steps between AAA servers between systems in different fields, simplify the authentication data of users roaming between networks, and use simple information to achieve the most secure identity verification.

SA2: The security relationship between the AAA server and the service provider can prevent malicious attackers from imitating legitimate server providers for data theft and other tasks. In the security issue of SA2, effective two-party authentication will be proposed to establish a trust relationship between the AAA server and the server provider.

SA7: The security relationship between server providers under the same network requires information communication and other tasks.

Dynamically Established:

SA3: The establishment of this security relationship is based on the security relationship dynamically established when the user applies to join a certain network after the application is successful. The security relationship has a set lifetime. SA3 is a security relationship between the network and the user, and this security relationship can help realize user authentication. The security relationship between AAA_H (Home's AAA Server) and users can help AAA_H to authenticate users. In this security relationship, we will focus on effectively and quickly authenticate users so that users only need to authenticate in the local network. When the user roams to other networks, it can also shorten the AAA of other networks. The authentication process of the server can realize anonymity at the same time.

SA4: Establish this security relationship. When the user roams to other networks, the user must provide the correct credentials. After correct AAA_F (Foreign's AAA Server) authentication, it will establish a security relationship. This security relationship can help AAA_F quickly process user requests and resource accounting on the network.

SA5 and SA6: This security relationship is the security relationship between the service provider

and the user. After the user passes the AAA_H authentication, the service provider verifies the certificate issued to the user. The service provider can grant the authority according to the user's service usage authority. Get service. In the two security relationships between SA5 and SA6, we will focus on users' privacy when using services and avoid revealing the services used by users. Therefore, in the third year plan, we will also focus on one of the key points, in the user's privacy.

At present, most of the schemes proposed by many researchers are aimed at the privacy of users. There are few studies on solving authentication in a seamless roaming network. Therefore, this research topic will improve the methods proposed by previous scholars to solve privacy problems and research in conjunction with roaming authentication issues. Previously, scholars such as Konidala *et al.* [21] had raised improving users when using capability-based services. They pointed out in the paper that tickets should be issued based on the identity of the user. Before using the service, users should register with the certification center and be certified. In their scheme, they used blind signatures to hide their identities and achieve anonymity. Wait for the authentication center to pass the authentication, and then give the user a legal ticket function.

For safety reasons, the ticket's validity period should be set to one day, and the next is invalid. When the user wants to use the service, the certification center will issue a given ticket to request the service provider to provide the service. Although the above scheme achieves anonymity, in their scheme, the user's public key and private key are stored in the mobile device, and the public key and private key are used for signing and encryption throughout the scheme. Therefore, if the user's mobile device is lost, it will bring more security issues. Besides, this scheme has not yet considered roaming issues. Therefore, this scheme is not suitable for roaming to different networks to access various services. Therefore, in this research topic, we will improve the generation and storage of public and private keys and design a set of user authentication schemes to maintain user anonymity and privacy when accessing services while avoiding being in the same domain. As long as the certification authority verifies the legal identity, there is no need to re-register for identity verification when roaming in other domains in the future. The user only needs to submit the information that has been authenticated—access services in the domain.

4 Conclusions

This research proposes a seamless roaming service access and identity verification mechanism suitable for cloud computing and develops a security mechanism for inter-domain cloud computing. Considering that mobile devices' available resources are relatively limited, the security mechanism developed in this research will focus on low computing power and supplement it with other reliable security technologies to meet the requirements for security and efficiency.

This research has a practical and forward-looking security mechanism for cloud computing and provides a basic security mechanism for network roaming. By allowing users to use the network, it can increase its convenience without losing security and further improve the entire network, dispel the doubts of enterprises about the introduction of cloud computing, and make cloud computing more popular, information technology, and the Internet.

We summary the main works of these research issues as follows:

Topic 1: Research on Service Access and Authentication in a Seamless Roaming

1) Establish a seamless roaming network environment.
2) Establish an AAA server simulation environment.
3) Establish a security mechanism in a seamless roaming network environment.
4) Establish a service access mechanism under a seamless roaming network environment.

Topic 2: Research on Privacy Protection in Seamless Roaming

1) Establish a security mechanism for the inter-domain cloud service authentication platform.
2) Establish a service access mechanism in a seamless roaming network environment with privacy protection.

Acknowledgments

The Ministry of Science and Technology partially supported this research, Taiwan (ROC), under contract no.: MOST 108-2410-H-468-023 and MOST 108-2622-8-468-001-TM1.

References

[1] M. S. Hwang and I. C. Lin, *Introduction to Information and Network Security (6ed, in Chinese)*, Taiwan: Mc Graw Hill, 2017.

[2] B. Patel, J. Crowcroft, "Ticket based service access for the mobile user," in *Proceedings of the 3rd Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 223–233, 1997.

[3] H. Wang, J. Cao, Y. Zhang, "Ticket-based service access scheme for mobile users," *Australian Computer Science Communications*, vol. 24, no. 1, 2002.

[4] E. Fogelstroem, A. Jonsson, C. Perkins, *Mobile IP Regional Registration*, RFC 4857, IETF, 2007.

[5] C. Perkins, *Mobile-IP Local Registration with Hierarchical Foreign Agents*, IETF, Internet Draft, Feb. 22, 1996. (`https://tools.ietf.org/html/draft-perkins-mobileip-hierfa-00`)

[6] J. Li, P. Zhang, and S. Sampalli, "Improved security mechanism for mobile IPv6," *International Journal of Network Security*, vol. 6, no. 3, pp. 291-300, 2008.

[7] S. Qadir, M. U. Siddiqi, W. F. M. Al-Khateeb, "An investigation of the merkle signature scheme for cryptographically generated address signatures in mobile IPv6," *International Journal of Network Security*, vol. 17, no. 3, pp. 311-321, 2015.

[8] L. H. Chang, C. L. Lo, J. J. Lo, W. T. Liu, C. C. Yang, "Mobility management with distributed AAA architecture," *International Journal of Network Security*, vol. 4, no. 3, pp. 241-247, 2007.

[9] Z. A. Zardari, J. He, M. S. Pathan, S. Qureshi, M. I. Hussain, F. Razaque, P. He, and N. Zhu, "Detection and prevention of Jellyfish attacks using kNN algorithm and trusted routing scheme in MANET," *International Journal of Network Security*, vol. 23, no. 1, pp. 77-87, 2021.

[10] E. Sagatov, K. Lovtsov, and A. Sukhov, "Identifying anomalous geographical routing based on the network delay," *International Journal of Network Security*, vol. 21, no. 5, pp. 760-767, 2019.

[11] Y. Lv, K. Liu, D. Zhang, and Z. Miao, "A secure routing protocol based on reputation mechanism," *International Journal of Network Security*, vol. 20, no. 5, pp. 862-871, 2018.

[12] M. Guerrero-Zapata, N. Asokan, "Securing ad hoc routing protocols," in *Proceedings of the 3rd ACM Workshop on Wireless Security*, pp. 1-10, 2002.

[13] C. Perkins, E. Belding-Royer, S. Das, *Ad hoc On-Demand Distance Vector (AODV) Routing*, RFC 3561, IETF, July 2003.

[14] Z. Tbatou, A. Asimi, Y. Asimi, Y. Sadqi, A. Guezzaz, "A new mutuel Kerberos authentication protocol for distributed systems," *International Journal of Network Security*, vol. 19, no. 6, pp. 889-898, 2017.

[15] E. El-Emam, M. Koutb, H. Kelash, and O. S. Faragallah, "An authentication protocol based on Kerberos 5," *International Journal of Network Security*, vol. 12, no. 3, pp. 159-170, 2011.

[16] F. Al-Ayed, C. Hu and H. Liu, "An efficient practice of privacy implementation: Kerberos and Markov chain to secure file transfer sessions," *International Journal of Network Security*, vol. 20, no. 4, pp. 655-663, 2018.

[17] M. S. Malhi, U. Iqbal, M. M. Nabi, and M. A. I. Malhi, "E-learning based on cloud computing for educational institution: Security issues and solutions," *International Journal of Electronics and Information Engineering*, vol. 12, no. 4, pp. 162-169, 2020.

[18] C. de Laat, G. Gross, L. Gommans, J. Vollbrecht, D. Spence,, *Generic AAA Architecture*, RFC 29.3, IETF, Aug. 2000.

[19] S. Farrell, J. Vollbrecht, P. Calhoun, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, D. Spence, *AAAA Authorization Requirements*, RFC 2906, IETF, Aug. 2000.

[20] B. Aboba, M. Beadles, J. Arkko, P. Eronen, , *The Network Access Identifier*, RFC 4282, IETF, Dec. 2005.

[21] D. M. Konidala, D. N. Duc, D. Lee, K. Kim, "A capability-based privacy-preserving scheme for pervasive computing environments," in *Third IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 136-140, 2005.

Biography

Hsien-Tsen Pan received B.S. in Business Administration From Soochow University Taipei Taiwan in 1999; M.S in Information Engineering, Asia University Taichung Taiwan 2015; Doctoral Program of Information Engineering, Asia University Taichung Taiwan from 2015 till now. From 2011 to 2014, he was the manager in Enterprise Service Chunghwa Telecom South Branch Taichung Taiwan. From 2014 to 2017, he was the operation manager in Medium division Taiwan Ricoh Co., Ltd. Taichung Taiwan From 2017 Sep 20 he is the Apple MDM Server Service VP in Get Technology Co.Ltd. Taipei Taiwan.

Shu-Fen Chiou received a B.B.A degree in Information Management from National Taichung Institute of Technology, Taichung, Taiwan, ROC, in 2004; She studied M.S. degree in Computer Science and Engineering from National Chung Hsing University for one year, and she started to pursue the Ph.D. degree. She received a Ph. D. from Computer Science and Engineering from National Chung Hsing University in 2012. She is currently an assistant professor of Department of Information Management, National Taichung University of Science and Technology. Her current research interests include information security, network security, data hiding, text mining and big data analysis.

Min-Shiang Hwang received M.S. in industrial engineering from National Tsing Hua University, Taiwan in 1988, and a Ph.D. degree in computer and information science from National Chiao Tung University, Taiwan, in 1995. He was a distinguished professor and Chairman of the Department of Management Information Systems, NCHU, during 2003-2011. He obtained 1997, 1998, 1999, 2000, and 2001 Excellent Research Awards from the National Science Council (Taiwan). Dr. Hwang was a dean of the College of Computer Science, Asia University (AU), Taichung, Taiwan. He is currently a chair professor with the Department of Computer Science and Information Engineering, AU. His current research interests include information security, electronic commerce, database, data security, cryptography, image compression, and mobile computing. Dr. Hwang has published over 300+ articles on the above research fields in international journals.

An Efficient Lossless Dual Images Secret Sharing Scheme Using Turtle Shell Reference Matrix

Jiang-Yi Lin[1,2,3], Yu Chen[4], Chin-Chen Chang[3], and Yu-Chen Hu[5]
(Corresponding author: Chin-Chen Chang)

School of Computer and Information Engineering, Xiamen University of Technology, China[1]
Key Laboratory of Fujian Universities for Virtual Reality and 3D Visualization, China[2]
Department of Information Engineering and Computer Science, Feng Chia University[3]
No. 100, Wenhua Road, Xitun District, Taichung 40724, Taiwan
School of Information Science and Engineering, Fujian University of Technology. China[4]
Department of Computer Science and Information Management, Providence University[5]
(Email: alan3c@gmail.com)
(Received Ari. 12, 2019; Revised and Accepted Feb. 5, 2020; First Online Feb. 26, 2020)

Abstract

This work proposes a novel reversible (2, 2) secret sharing scheme based on turtle shells (TS) with dual images. According to the characteristic of the TS reference matrix in a 3×3 block, each pixel in the cover image can be used to conceal an octal digit by shifting it to the appropriate location, then two shadows with high quality are generated in the end. After the location conflict problem is solved, the hidden secret data can be extracted without any error as long as both shadows are gathered, and the cover image can be restored losslessly through the pixel orientational relationship located at two shadows. The experimental results demonstrate that the proposed scheme can achieve higher embedding capacity and maintain good image quality than some existing methods. In addition, the proposed scheme is effective against statistic attacks on pixel-value difference histograms (PDH).

Keywords: Pixel-Value Difference Histograms (PDH); Reference Matrix; Secret Sharing; Shadow; Turtle Shell (TS)

1 Introduction

To provide data protection, traditional cryptography can encrypt the required data using a secret key. Although cryptography can increase data security, both the encrypted phase and the decryption phase need tremendous computational complexity. This high computational complexity problem becomes more serious in a public-key cryptosystem. Data hiding [2,3,5,7,9,15,18,19,21–23,25] is another approach in which only a small amount of calculation is required. The important data can be embedded into a cover image by data hiding algorithm without causing significant distortion to generate a stego-image. Therefore, unlike cryptography, the embedding result is maintaining meaningful.

Besides data hiding, which only creates one stego image, secret sharing can create multiple stego images. The (k, n) threshold secret sharing scheme, where $k \leq n$, was first proposed by Shamir [20] in 1979. In his method, the secret data was torn into n pieces, and each piece was held by a participant. The secret data can be retrieved by the cooperation of k or more participants; otherwise, when an insufficient number of participants is gathered, i.e., less than k, the original secret data cannot be restored. The secret sharing principle can be extended to the image domain, called as secret image sharing [10,13,17,24]. In 2004, Lin and Tsai [13] introduced a novel (k, n) threshold secret image sharing scheme that provides the additional features of authentication and steganography. Although this scheme can be employed for a full-color image, the cover image cannot be completely restored. In 2009, Chang *et al.* presented a (2, 2) verifiable secret sharing (VSS) scheme [10] to protect the secret image wherein the restored secret image can be verified by certified users. Due to the high computational complexity, however, their method is not suitable for real-time applications. Later, Wu and San [24] proposed a secret sharing scheme based on random mesh. In their method, the pixel expansion problem is inhibited, and the image distortions of the generated shadows are reduced.

Recently, the reversibility of secret image sharing has aroused great attention [1,4,6,8,11,12,14,16]. The first reversible secret image sharing scheme using two shadows was proposed by Chang *et al.* [4] in 2007. In their method, two 5-base digits were embedded into an identical cover pixel pair along the main diagonal and the second diagonal direction in the exploiting modification direction (EMD) magic matrix. The embedding ratio (ER) is about 1 bit per pixel (bpp). Using the characteristics of

the Sudoku reference matrix, Chang *et al.* [6] introduced a dual image reversible data hiding scheme in 2013. An ER of 1.55 bpp is achieved when the embedded image quality is around 40 dB in terms of peak signal-to-noise ratio (PSNR). In 2013, Lee and Huang [11] developed a novel reversible data hiding scheme. The reversibility of their method is achieved by the combination of the pixel orientations located at two generated shadows. Though the image quality of the stego-image was up to 49 dB, the ER of the method was just 1.07 bpp.

Inspired by the TS-based reference matrix introduced by Chang *et al.* [5] in 2014, Liu and Chang [14] proposed a novel visual secret sharing scheme innovatively employing the turtle shell reference matrix. Though the achieved the image quality of the shadows ranged from 45 dB to 51 dB, the ER of their scheme is nearly 1 bpp. To improve the ER, an efficient turtle shell based lossless secret sharing scheme using dual images is proposed in this paper. We utilize the characteristic of the TS reference matrix in a 3×3 block and conceal three secret bits into a pixel which picked up from the cover image to generate two high quality shadows. In the data extraction phase, the embedded secret data and the cover image can be recovered losslessly with the help of the pixel orientational relationship located at two shadows. In our experiments, an ER gain of 1.5 bpp is achieved and high image quality of 46 dB for the generated shadows is obtained. Under the static attack of the pixel-value difference histogram (PDH), our scheme can maintain a similar shape with the cover image for the generated shadows.

The rest of this paper is organized as follows. In Section 2, Chang *et al.*'s method for the TS-based data hiding scheme is briefly overviewed. The proposed scheme is descripted in Section 3, and the experimental results are analyzed in Section 4. Section 5 presents our conclusions.

2 Review of Chang *et al.*'s Method

In 2014, Chang *et al.* are the pioneers who first introduced the concept of the turtle shell matrix in the field of data hiding. In their method, the reference matrix T, as shown in Figure 1, which consists of adjacent turtle shells, is utilized both in the data embedding phase and the data extracting phase. In a turtle shell, there are two elements on the back; and six elements on the edge for a total of eight elements. The values of these eight elements range from 0 to 7. The reference matrix T is constructed by the following rules:

1) Every two consecutive elements in the same row should satisfy:

$$T(P_i + 1, P_{i+1}) = (T(P_i, P_{i+1}) + 1) \bmod 8 .$$

Here, $(P_i,\ P_{i+1})$ is a pair of grayscale pixel values selected from the cover image, and $P_i,\ P_{i+1}$ are arranged from 0 to 255. The notation $T(P_i,\ P_{i+1})$ is the element where this pixel pair is located at T.

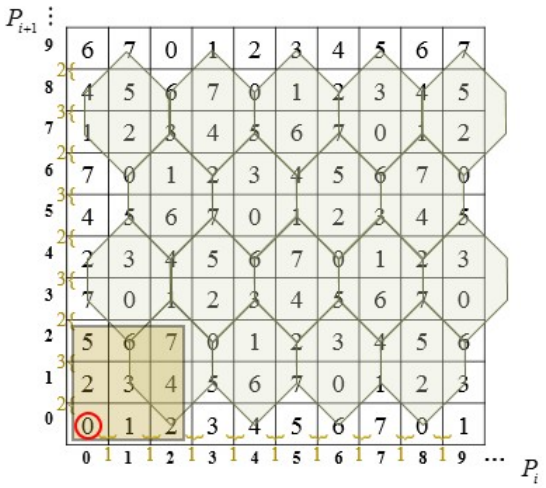


Figure 1: The reference matrix T

2) Every two consecutive elements in the same column should satisfy:

$$T(P_i, P_{i+1}+1) = \begin{cases} (T(P_i, P_{i+1})+2) \bmod 8 & P_{i+1} \text{ is even,} \\ (T(P_i, P_{i+1})+3) \bmod 8 & P_{i+1} \text{ is odd.} \end{cases}$$

After all the elements of T are acquired by the above rules, the reference matrix can be created. In order to generate the identical reference matrix T in the data embedding phase and data extraction phase, $T(0,\ 0)$ is set to 0, as shown in Figure 1.

In Chang *et al.*'s method, the stego-image is constructed as follows. Sequentially select the pixel pair $(P_i,\ P_{i+1})$ from the cover image; and locate it at reference matrix T, where the location is denoted as $T(P_i,\ P_{i+1})$.

1) If $T(P_i,\ P_{i+1})$ belongs to a turtle shell, an 8-base secret digit D can be concealed. The stego pixel pair (P'_i, P'_{i+1}) is gained where $T\,(P'_i, P'_{i+1})$ is equal to D.

2) If $T(P_i,\ P_{i+1})$ is located at the border, a 3×3 block is generated to embed an 8-base secret digit D. The original pixel pair is modified to (P'_i, P'_{i+1}) where $T\,(P'_i, P'_{i+1})$ is equal to D. Figure 1 depicts an example of the pixel pair (0, 0), which is located at the border.

After each pixel pair is sequentially processed, the stego image is generated. In the data extraction phase, each stego pixel pair will be used to extract an 8-base secret digit D. By locating the pixel pair at the reference matrix T, the value of the element in the turtle shell is extracted and stored as the secret digit D.

The ER of Chang *et al.*'s method can reach up to 1.5 bpp, but the cover image can't be recovered. Examine the characteristic of the reference matrix T carefully, there are two and only two types that the location $T(P_i,\ P_{i+1})$ belongs to: the upper type, which is denoted as upper back (or edge) elements, and the lower type, which is denoted as lower back (or edge) elements, as shown in Figure 2. Thus, two rules should be noticed:

RULE-1: Given an upper (or lower) location in T, a 3×3 block that considers this location as the right-bottom (or left-top) corner can be determined. We can see that the elements in this block must contain 0 to 7.

RULE-2: In the determined 3×3 block, the only duplicate elements will appear in the original location and its left-top (or right-bottom) neighbor which are denoted as $T(P_i, P_{i+1}) = T(P_i\text{-}2, P_{i+1}+1)$ (or $T(P_i, P_{i+1}) = T(P_i+2, P_{i+1}\text{-}1)$).

In order to prove the above rules, we simply set the value of the location as t. Taking the location belonging to the upper type into account, the other elements in the 3×3 block revealed in Figure 3(a) can be uniquely determined by value t. In Figure 3(a) for any given value $t\in[0, 7]$, the elements in the block must contain 0 to 7. The same conclusion can be found when the location belongs to the lower type, as shown in Figure 3(b). Thus, *RULE*-1 is completely proved. At the same time *RULE*-2 is revealed obviously according to Figure 3.

From the above observation, we propose a novel reversible (2, 2) secret sharing scheme based on TS reference matrix with dual images. The embedding capacity of the proposed scheme can reach up to 1.5 bpp and maintain a high image quality of the stego images.

3 Propose Scheme

We will present our method in this section which is organized as follows: (1) shadows generation phase, (2) data extraction and image recovery phase, and (3) example of the proposed scheme.

3.1 Shadows Generation Phase

In the beginning, the dealer establishes the reference matrix T. Then, a pixel P_i is selected from the cover image, which is replicated to gain a pixel pair (P_i, P_i) and located at T. The location is denoted as $T(P_i, P_i)$. Obviously, the location $T(P_i, P_i)$ can only be locate at the primary diagonal of T. According to this characteristic, the location $T(P_i, P_i)$ can be further classified into the following three categories.

Category 1: $T(P_i, P_i)$ belongs to the border, i.e., $P_i\in[0, 1]$ or $P_i\in[254, 255]$, as shown by the red triangle-marked values in Figure 4;

Category 2: $T(P_i, P_i)$ belongs to the upper type, as shown by the red circle-marked values in Figure 4;

Category 3: $T(P_i, P_i)$ belongs to the lower type, as shown by the blue circle-marked values in Figure 4.

According to the categories location $T(P_i, P_i)$ belongs to, we can embed an 8-base secret data into Categories 2 and 3. After data embedding, the cover pixel pair (P_i, P_i) is modified into a stego pixel pair (P_{i1}, P_{i2}), where pixel P_{i1} is kept by shadow 1 and P_{i2} is kept by shadow 2. A more detailed description of the data embedding phase is given as follows:

1) If the location $T(P_i, P_i)$ belongs to Category 1, then the cover pixel pair (P_i, P_i) is not utilized for embedding and is kept intact to generate the stego pixel pair (P_{i1}, P_{i2}), i.e., $P_{i1}=P_i$, and $P_{i2}=P_i$.

2) If the location $T(P_i, P_i)$ belongs to Category 2 (or 3), then we construct an upper (or a lower) 3×3 block that considers this location $T(P_i, P_i)$ as the right-bottom (or left-top) corner and utilize the 8 elements in the block, except $T(P_i, P_i)$, to conceal an 8-base secret data D. The cover pixel pair (P_i, P_i) is modified into a stego pixel pair (P_{i1}, P_{i2}), where $T(P_{i1}, P_{i2})$ is equal to D, as shown in Figure 5.

It should be noted that, the **location collision problem** will occur with the same stego pixel pairs gained from different cover pixel pairs when they attempt to occupy an identical location in the reference matrix T. For example, a collision may happen when both locations of $T(2, 2)$ and $T(4, 4)$ are embedding secret data 4, and they will be both modified to $T(2, 4)$.

To avoid these collisions to ensure the reversibility of our proposed scheme, we shift the location to the cover pixel pair to overcome the collision problem. For the location, $T(P_i, P_i)$ belongs to Category 2 (or 3) if the stego pixel pair (P_{i1}, P_{i2}) is located at the collision position, i.e., $P_{i1}= P_i$ and $P_{i2}= P_i+2$ (or $P_{i1}= P_i$ and $P_{i2}= P_i\text{-}2$). We shift the stego pixel pair to the cover pixel pair, i.e., $P_{i1}= P_i$ and $P_{i2}= P_i$. Finally, the **location collision problem** is well-surmounted. After the whole pixel in the cover image is processed, two shadows, which are marked as $S1$ and $S2$ are generated.

3.2 Data Extraction and Image Recovery Procedure

This phase can be implemented only if two participants release their shadows simultaneously. We select a pixel P_{i1} from $S1$ and P_{i2} from $S2$ at the same location to construct the stego pixel pair (P_{i1}, P_{i2}). The difference of stego pixel value d is calculated by:

$$d = P_{i1} - P_{i2}.$$

The secret data D can be obtained in the following way:

Case 1: If d equals 0 and the stego pixel pair (P_{i1}, P_{i2}) belongs to the border, which is mentioned in Section 3.1 means that no secret data is hidden; otherwise, the secret data D can be retrieved by:

$$D = \begin{cases} T(P_{i1}, P_{i2}+2) & (P_{i1}, P_{i2}) \textit{ locate at the upper,} \\ T(P_{i1}, P_{i2}-2) & \text{Otherwise.} \end{cases}$$

Case 2: If d is not equal to 0, then the secret data D is retrieved by:

$$D = T(P_{i1}, P_{i2}).$$

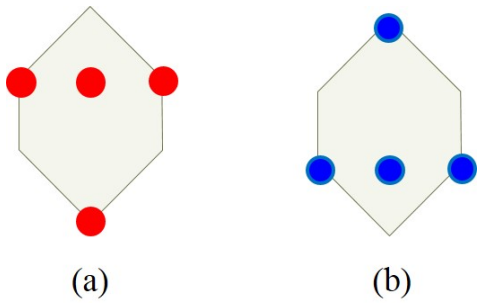


Figure 2: Two element types in a turtle shell. (a) The upper type (b) The lower type.

t+3	t+4	t+5
t	t+1	t+2
t-2	t-1	t

(a)

t	t+1	t+2
t-2	t-1	t
t-5	t-4	t-3

(b)

Figure 3: Two types of the 3×3 block determined by the location $T(P_i, P_{i+1})$. The elements with same back ground color are belong to the same type. (a) the location belongs to the upper type (b) the location belongs to the lower type.

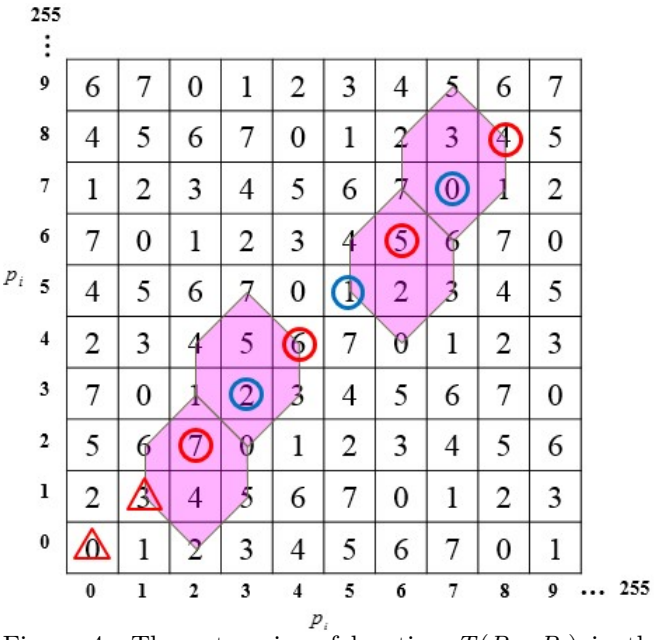


Figure 4: The categories of location $T(P_i, P_i)$ in the reference matrix T

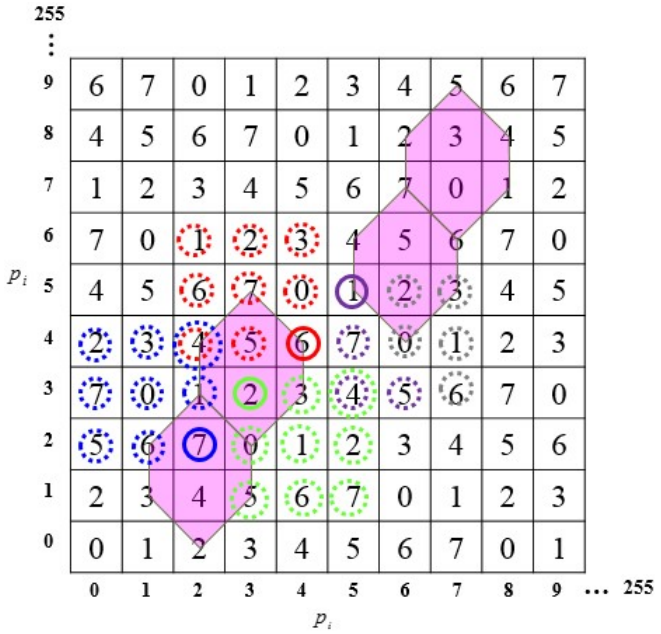


Figure 5: The embedding phase of location $T(P_i, P_i)$; The solid circles denote the cover location and the dashed circles denote the elements utilized for data embedding

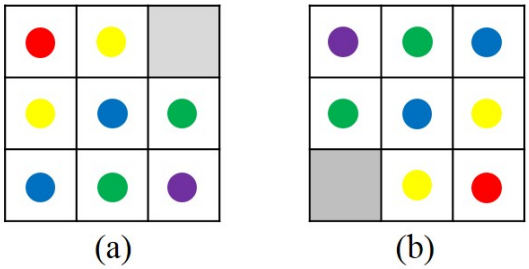


Figure 6: The locations $T(P_i, P_i)$. Pink circle marked denoted the original location. The location with gray background color demonstrated the unused location. (a) The original location belongs to the upper type (b) The original location belongs to the lower type.

Meanwhile, the cover pixel value P_i can be retrieved in the following way:

Case 1: If d equals 0, set the cover pixel value $P_i = P_{i1}$.

Case 2: If d equals -4 (or +4), the location is shown by the red circle-marked in Figure 6, and the cover pixel value P_i is obtained by $P_i = P_{i1}+2$ (or $P_i = P_{i1}-2$).

Case 3: If d equals -3 (or +3), the location is shown by the yellow circle-marked in Figure 6. We assign t=+1 or +2 (-1 or -2 when d equals +3). Examine the type what $T(P_{i1}+t, P_{i1}+t)$ belongs to, and if it belongs to the upper type (lower type when d equals to +3), set $P_i = P_{i1}+t$.

Case 4: If d equals to -2 (or +2), the location is shown by the blue circle-marked in Figure 6. We assign t=+1 or +2 (-1 or -2 when d equals to +2). Examine the type that $T(P_{i1}+t, P_{i1}+t)$ belongs to, if it belongs to the upper type (lower type when d equals to +2), set $P_i = P_{i1}+t$.

Case 5: If d equals -1 (or +1), the location is shown by the green circle-marked in Figure 6. We assign t=+1 or 0 (-1 or 0 when d equals to +1). Examine the type that $T(P_{i1}+t, P_{i1}+t)$ belongs to, and if it belongs to the upper type (lower type when d equals to +1), set $P_i = P_{i1}+t$.

Obviously, after processing the whole pixels in these two generated shadows, not only the embedded secret data but also the cover image can be recovered without errors.

3.3 Example of the Proposed Scheme

In this section, an example is utilized to interpret the principle of the proposed scheme. Suppose the cover pixel values are 1, 2, 4, 5 and the embedded 8-base secret data are D=4, 6, 3. We first show how to embed the secret data into the cover pixel P_i to construct the stego pixel pair (P_{i1}, P_{i2}):

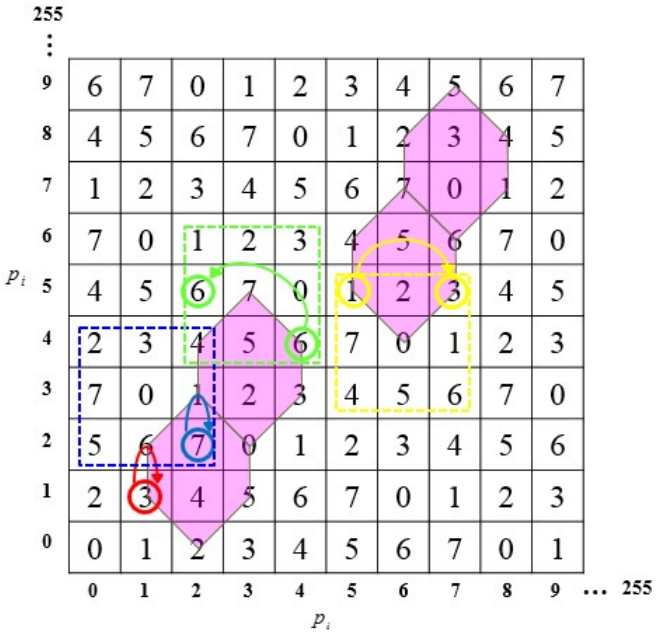


Figure 7: Example of data embedding of the proposed scheme

1) For P_i=1: Because P_i belongs to the border, no secret data is concealed. The stego pixels are all set to P_i, i.e., $P_{i1} = P_i$ and $P_{i2} = P_i$, as shown in Figure 7 with the red color.

2) For P_i=2: $T(2, 2)$ belongs to the upper type, so we utilized the upper 3×3 block for data embedding, as shown in Figure 7 with the blue color. Select the 8-base secret data from D that is 4 for embedding. As the location $T(2, 4)$ is equal to secret data 4 and it is a collision location, the stego pixel pair is kept unchange to embed secret data 4. Hence the pixels in shadow $S1$ are 1, 2 and 1, 2 in shadow $S2$.

3) For P_i=4: $T(4, 4)$ belongs to the upper type either, so we also utilized the upper 3×3 block for data embedding, as shown in Figure 7 with the green color. Select the 8-base secret data 6 from D. The original location is modified to $T(2, 5)$, which is equal to secret data 6 for embedding secret data 6. Hence, the pixels in shadow $S1$ are 1, 2, 2 and 1, 2, 5 in shadow $S2$.

4) For P_i=5: $T(5, 5)$ belongs to the lower type, so we also utilized the lower 3×3 block for data embedding, as shown in Figure 7 with the yellow color. Select the 8-base secret data 3 from D. The original location is modified to $T(7, 5)$, which is equal to secret data 3 for embedding secret data 3. Hence the pixels in shadow $S1$ are 1, 2, 2, 7 and 1, 2, 5, 5 in shadow $S2$.

Let us continue this example to illustrate the extracting phase.

1) Select pixel 1 from $S1$ and 1 from $S2$, since the pixels

are all belong to the border, no secret data is concealed in this situation, and the original pixel is 1.

2) Pick next pixel 2 from $S1$ and 2 from $S2$. Their distance d is calculated by Equation (3) as 0. Since the location $T(2, 2)$ belongs to the upper type, according to Equation (4), the secret data can be retrieved as $T(2, 4)$=4 and the original pixel value is 2. Thus, the set of original pixel values are 1, 2 and the set of secret data D is 4.

3) Continue picking the next pixel 2 from $S1$ and 5 from $S2$. Their distance d is calculated by Equation (3) as -3. In this situation, this belongs to Case 3 in Section 3.2. The embedded secret data can be retrieved directly as $T(2, 5)$=6. We assign t=+1 or +2. Examine the types to which $T(3, 3)$ and T(4, 4) belong, we can see that $T(4, 4)$ belongs to the upper type, which reveals that the original pixel value is equal to 4. Hence, the set of original pixel values become 1, 2, 4 and the set of secret data D is 4, 6.

4) Continue picking the next pixel 7 from $S1$ and 5 from $S2$. Their distance d is calculated by Equation (3) as 2. In this situation, this belongs to Case 4 in Section 3.2. The embedded secret data can be retrieved directly as $T(7, 5)$=3. We assign t=-1 or -2. Examine the types to which $T(5, 5)$ and $T(6, 6)$ belong, we can see that $T(5, 5)$ belongs to the lower type, which reveals that the original pixel value is equal to 5. Hence, the set of original pixel values become 1, 2, 4, 5 and the set of secret data D is 4, 6, 3.

Obviously, the embedding secret data and the original pixel values are all recovered losslessly.

4 Experimental Results

In this section, we first validate the performance of the proposed scheme with some existing schemes using two criteria: the ER and PSNR value, followed by an examination of the resisting PDH analysis of the proposed scheme. The experiments were running on a personal computer with a Windows 7 operating system and an Intel Xeon E3-1225 v5, 3.3GHz CPU, and 8GB memory. All the experiments were implemented by Matlab 2012R. All test images are 8-bit grayscale images of size 512×512 as shown in Figure 8.

4.1 The Performance of the Proposed Scheme

The ER is commonly used to estimate the performance of a data hiding scheme measured by the embedding capacity (bpp), which is calculated as:

$$ER = N/\left(num \times W \times H\right),$$

where N represents the total number of secret bits concealed in the shadows. Notation num denotes the number of shadows used. In our method, the value of num is 2. The notations W and H are denoted as the width and height of the cover image, respectively.

Meanwhile, the PSNR is utilized to evaluate the image quality (dB) of a generated shadow which is defined as:

$$PSNR = 10\log_{10}\left(\frac{255^2 \times W \times H}{\sum_{i=1}^{W}\sum_{j=1}^{H}\left(I_{ij} - S_{ij}\right)^2}\right),$$

where I_{ij} and S_{ij} refer to the pixel values in the corresponding location of the cover image I and the shadow S, respectively.

Table 1 show the comparison results between the proposed scheme and the methods in [10–12, 14, 16] in terms of maximum ER. From Table 1, the highest ER of 1.5 bpp is achieved by our method. While comparing to the four related methods in [10–12, 14], the gains of the ER are 0.5 bpp, 0.76 bpp, 0.43 bpp and 0.5 bpp, respectively.

Table 1: Results of the embedding ratio of the comparative schemes

Images	[10]	[12]	[11]	[14]	**Proposed**
Barbara	1	0.74	1.07	1	1.5
Baboon	1	0.74	1.07	1	1.5
Goldhill	1	0.74	1.07	1	1.5
Boat	1	0.74	1.07	1	1.5
Lena	0.99	0.75	1.07	1	1.5
Peppers	1	0.75	1.07	1	1.5
Zelda	0.99	0.74	1.07	1	1.5
Airplane	1	0.74	1.07	0.99	1.5
Average	**1**	**0.74**	**1.07**	**1**	**1.5**

Table 2 show the comparison results of shadow image quality between the proposed scheme and the methods in [10–12, 14] in terms of PSNR under the maximum ER. As can be seen in Table 2, the proposed scheme not only has a higher ER than the method in [10], but also gains neatly 6.13 dB in $S1$ and 6.12 dB in $S2$, respectively. Though our method has no superiority in terms of PSNR compared to the methods in [12] and [11], their method yields a lower ER. The method in [14] achieved a better PSNR of 5.69 dB in $S1$ than our scheme did, yet the proposed scheme performed slightly superior at 0.31 dB in $S2$.

The proposed method explores more locations around the cover pixel, which reduces the quality of the generated shadows. The comparison of average PSNR between the proposed method and the related methods [10–12, 14] is shown in Figure 9, from which we can see that although the proposed scheme is inferior than schemes in [11, 12] in terms of PSNR under different ER for both $S1$ and $S2$, it achieves a highest ER.

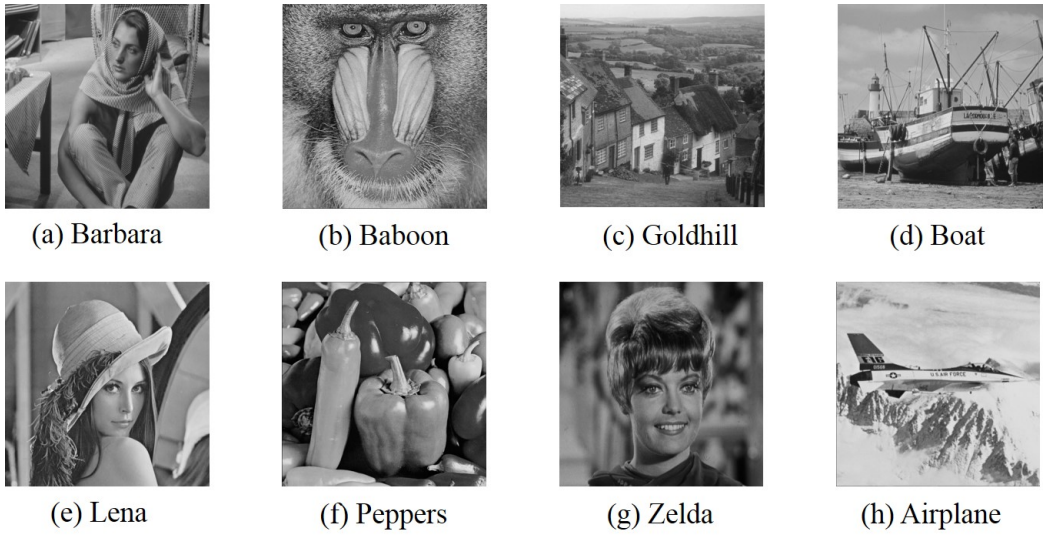


Figure 8: The eight test grayscale images

Table 2: Comparison of shadow image quality under the maximum ER in different schemes

Images	[10]		[12]		[11]		[14]		**Proposed**	
	$S1$	$S2$	$S1$	$S2$	$S1$	$S2$	$S1$	$S2$	$S1$	$S2$
Barbara	39.89	39.89	52.39	52.39	49.62	49.63	51.73	45.70	46.02	46.01
Baboon	39.91	39.91	52.39	52.39	49.61	49.63	51.72	45.71	46.02	46.05
Goldhill	39.90	39.90	52.39	52.39	49.62	49.63	51.72	45.71	46.03	46.02
Boat	39.89	39.89	52.39	52.39	49.62	49.63	51.68	45.73	46.04	46.00
Lena	39.89	39.89	52.39	52.39	49.63	49.63	51.69	45.70	46.02	46.03
Peppers	39.94	39.94	52.38	52.39	49.64	49.63	51.73	45.70	46.03	46.00
Zelda	39.89	39.89	52.39	52.39	49.62	49.63	51.71	45.70	46.04	46.03
Airplane	39.88	39.87	52.39	52.39	49.61	49.63	51.72	45.71	46.03	46.01
Average	**39.90**	**39.90**	**52.39**	**52.39**	**49.62**	**49.63**	**51.72**	**45.71**	**46.03**	**46.02**

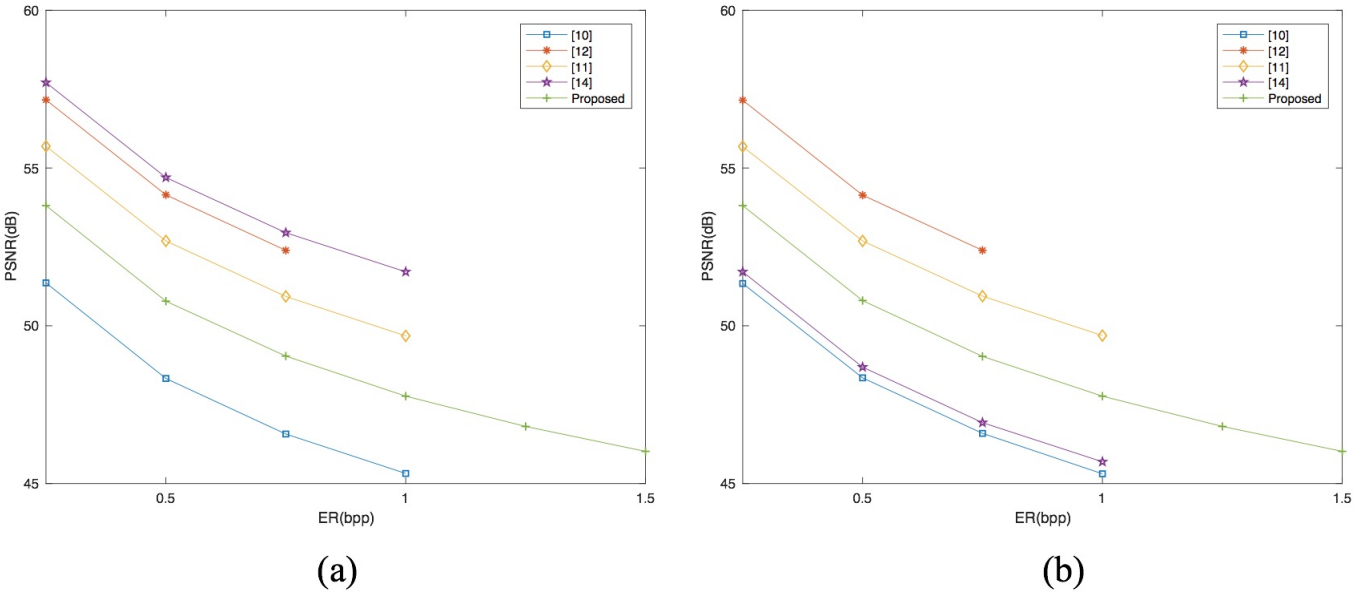


Figure 9: The results of the average PSNR of (a) $S1$ and (b) $S2$ for all test images under different ER

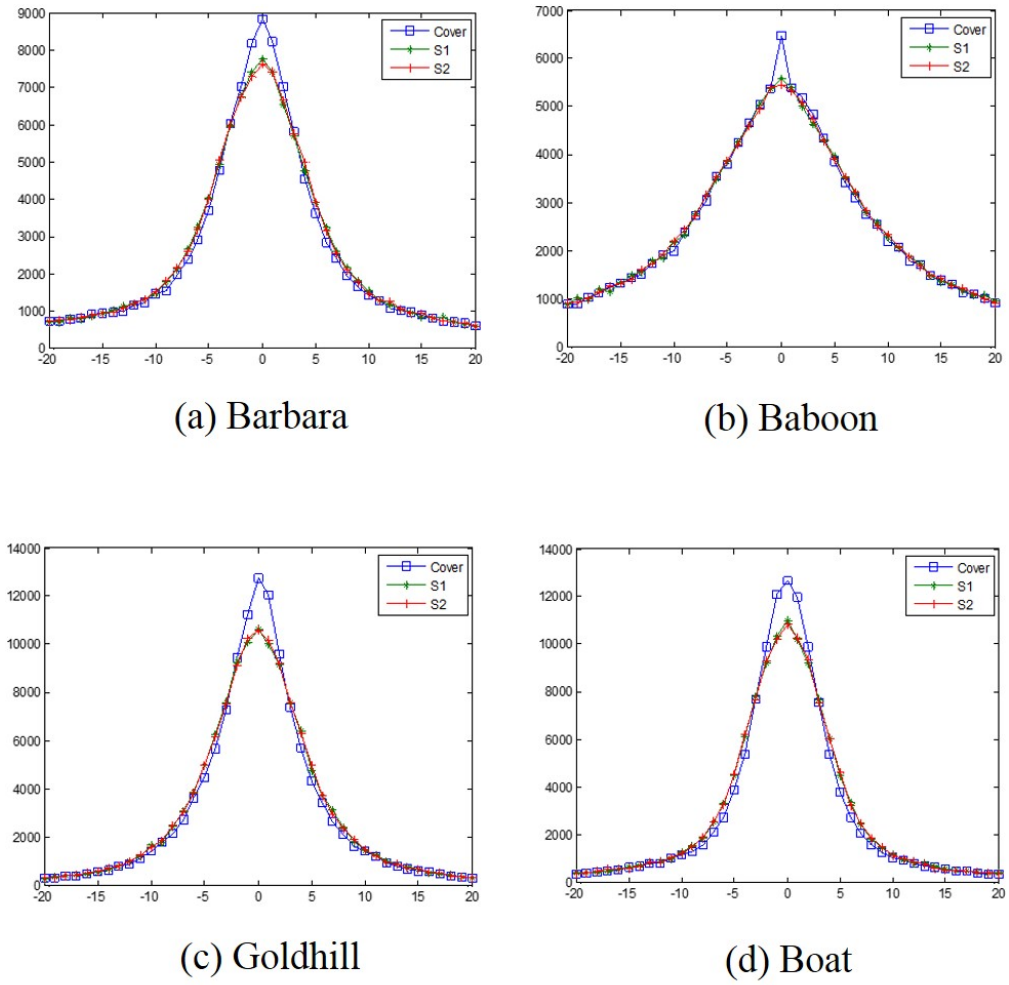


Figure 10: The PDH for the cover image and their corresponding shadows

4.2 PDH Analysis

The PDH is a measure to determine whether the image has been modified. It is constructed by the difference of two continuous pixels. After the PDH of the cover image and the shadows are constructed, the more the shapes between them are similar, the better a reversible secret sharing scheme is. The PDH of the four cover images and their corresponding shadows are shown in Figure 10. As can be seen from Figure 10, the shape of the PDH of the shadows generated by our scheme is very close to the shape of the PDH of the cover image.

5 Conclusions

According to the characteristic of the numbers located in a 3×3 block in the TS reference matrix, a novel reversible secret sharing scheme with dual images is proposed. The proposed scheme has the following features:

1) It is quite simple;
2) Reversibility is completely achieved;
3) It provides a higher ER;
4) It maintains the good visual quality of the shadows.

Simulation results show that the proposed scheme outperforms some previously published schemes. Moreover, the PDH reveal that the smoothness shape of the two shadows can be well preserved by the proposed scheme.

Acknowledgments

This work is partially supported by the Science and Technology Project of Xiamen (grant number: JT180439).

References

[1] K. Bharanitharan, N. T. Huynh and C. C. Chang, "Quadri-directional searching algorithm for secret image sharing using meaningful shadows," *Journal of Visual Communication and Image Representation*, vol. 28, pp. 105–112, 2015.

[2] C. C. Chang, Y. C. Chou and T. D. Kieu, "An information hiding scheme using sudoku," in *Proceedings of Third International Conference on Innovative Computing, Information and Control*, pp. 17–22, June 2008.

[3] I. C. Chang, Y. C. Hu, W. L. Chen, and C. C. Lo, "High capacity reversible data hiding scheme based on residual histogram shifting for block truncation coding," *Signal Processing*, vol. 108, pp. 376–388, 2015.

[4] C. C. Chang, T. D. Kieu, and Y. C. Chou, "Reversible data hiding scheme using two steganographic images," in *Proceedings of IEEE Region 10 International Conference (TENCON'07)*, pp. 1–4, Nov. 2007.

[5] C. C. Chang, Y. J. Liu, and T. S. Nguyen, "A novel turtle shell based scheme for data hiding," in *Proceedings of Tenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 89–93, Aug. 2014.

[6] C. C. Chang, T. C. Lu, G. Horng, Y. H. Huang, and Y. M. Hsu, "A high payload data embedding scheme using dual stego-images with reversibility," in *Proceedings of the Third International Conference on Information, Communications and Signal Processing*, pp. 1–5, Dec. 2013.

[7] Y. C. Hu, "High capacity image hiding scheme based on vector quantization," *Pattern Recognition*, vol. 39, no. 9, pp. 1715–1724, 2006.

[8] B. Jana, "Dual image based reversible data hiding scheme using weighted matrix," *International Journal of Electronics and Information Engineering*, vol. 5, pp. 6–19, Sep. 2016.

[9] H. J. Kim, C. Kim, Y. Choi, S. Wang, and X. Zhang, "Improved modification direction schemes," *Computer and Mathematics with Applications*, vol. 60, no. 2, pp. 319–325, 2010.

[10] T. H. N. Le, C. C. Chang, C. C. Lin and H. B. Le, "Sharing a verifiable secret image using two shadows," *Pattern Recognition*, vol. 42, no. 11, pp. 3097–3114, 2009.

[11] C. F. Lee and Y. L. Huang, "Reversible data hiding scheme based on dual stegano-images using orientation combinations," *Telecommunication Systems*, vol. 52, no. 4, pp. 2237–2247, 2011.

[12] C. F. Lee, K. H. Wang, C. C. Chang, and Y. L. Huang, "A reversible data hiding scheme based on dual steganographic images," in *Proceedings of the Third International Conference on Ubiquitous Information Management and Communication*, pp. 228–237, Jan. 2009.

[13] C. C. Lin and W. H. Tsai, "Secret image sharing with steganography and authentication," *Journal of Systems and Software*, vol. 73, no. 3, pp. 405–414, 2004.

[14] Y. J. Liu and C. C. Chang, "A turtle shell-based visual secret sharing scheme with reversibility and authentication," *Multimedia Tools and Applications*, vol. 77, no. 19, pp. 25295–25310, 2018.

[15] Y. J. Liu, C. C. Chang, and T. S. Nguyen, "High capacity turtle shell-based data hiding," *IET Image Processing*, vol. 10, no. 2, pp. 130–137, 2016.

[16] Y. J. Liu, J. Y. Lin and C. C. Chang, "A real-time dual-image-based reversible data hiding scheme using turtle shells," *Journal of Real-Time Image Processing*, vol. 16, pp. 673–684, 2019.

[17] M. Naor and A. Shamir, "Visual cryptography," *Lecture Notes in Computer Science*, vol. 950, pp. 1–12, 1995.

[18] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 3, pp. 354–362, 2006.

[19] C. Qin and Y. C. Hu, "Reversible data hiding in VQ index table with lossless coding and adaptive switching mechanism," *Signal Processing*, vol. 129, pp. 48–55, 2016.
[20] A. Shamir, "How to share a secret," *Communications of the Association for Computing Machinery*, vol. 22, no. 11, pp. 612–613, 1979.
[21] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890–896, 2003.
[22] P. Y. Tsai, Y. C. Hu, and H. L. Yeh, "Reversible image hiding scheme using predictive coding and histogram shifting," *Signal Processing*, vol. 89, no. 6, pp. 1129–1143, 2009.
[23] W. C. Wu, C. Chang and Y. C. Hu, "Lossless recovery of a VQ index table with embedded secret data," *Journal of Visual Communication and Image Representation*, vol. 18, no. 3, pp. 207–216, 2007.
[24] X. T. Wu and W. Sun, "Generalized random grid and its applications in visual cryptography," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 9, pp. 1541–1553, 2013.
[25] Y. H. Yu, C. C. Chang and Y. C. Hu, "Hiding secret data in images via predictive coding," *Pattern Recognition*, vol. 38, no. 5, pp. 691–705, 2005.

Biography

Jiang-Yi Lin. Jiang-Yi Lin received the B.S. and M.S. degrees in Computer science and Technology from FuZhou Uniersity, FuJian, China, in 2005 and 2008, respectively. He is currently pursuing the Ph. D. degree with the Multimedia and Secure Networking Laboratory (MSN lab), the Department of Information Engineering and Computer Science of Feng Chia University, Taichung, Taiwan. His research interests include image processing, secret sharing and steganography.

Yu Chen. Yu Chen received the B. S. degree in Computer and Application from Hunan University, Hunan, China in 1993, and M.S. degree in Software Engineering from Fuzhou University, Fujian, China, in 2006. Currently, he is an associate professor in the School of Information Science and Engineering, Fujian University of Technology (FJUT), China. His current research interests include information retrieval, data mining, and digital image processing.

Chin-Chen Chang. Chin-Chen Chang received his Ph.D. degree in computer engineering from National Chiao Tung University. His current title is Chair Professor in Department of Information Engineering and Computer Science, Feng Chia University, from February 2005. He is currently a Fellow of IEEE and a Fellow of IEE, UK. And, since his early years of career development, he consecutively won Outstanding Talent in Information Sciences of the R. O. C., AceR Dragon Award of the Ten Most Outstanding Talents, Outstanding Scholar Award of the R. O. C., Outstanding Engineering Professor Award of the R. O. C., Distinguished Research Awards of National Science Council of the R. O. C., Top Fifteen Scholars in Systems and Software Engineering of the Journal of Systems and Software, and so on. On numerous occasions, he was invited to serve as Visiting Professor, Chair Professor, Honorary Professor, Honorary Director, Honorary Chairman, Distinguished Alumnus, Distinguished Researcher, Research Fellow by universities and research institutes. His current research interests include database design, computer cryptography, image compression, and data structures.

Yu-Chen Hu. Yu-Chen Hu received his Ph.D. degree in computer science and information engineering from the Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan in 1999. Currently, Dr. Hu is a professor in the Department of Computer Science and Information Management, Providence University, Sha-Lu, Taiwan. His research interests include digital forensics, information hiding, image and signal processing, data compression, information security, and data engineering.

Revocable and Searchable Attribute-based Encryption Scheme with Multi-keyword and Verifiability for Internet of Things

Zhenhua Liu[1], Fangfang Yin[1], Jiaqi Ji[1], and Baocang Wang[2]
(Corresponding author: Fangfang Yin)
School of Mathematics and Statistics, Xidian University[1]
Xi'an 710071, P. R. China
State Key Laboratory of Integrated Services Networks, Xidian University[2]
Xi'an 710071, P. R. China
(Email: yinff21@163.com)
(Received July 26, 2020; Revised and Accepted Nov. 21, 2020; First Online Feb. 1, 2021)

Abstract

Internet of things (IoT) is a popular information and communication technologies paradigm, which can realize the interconnection of all things. Considering IoT's security and privacy requirements, we propose a novel attribute-based encryption scheme for IoT, which can provide user revocation, multi-keyword search, and data integrity verification. The proposed scheme utilizes the node selection algorithm KUNodes to achieve efficient user revocation. Simultaneously, the proposed scheme can support multi-keyword search and effectively avoid the return of a large number of irrelevant documents. Furthermore, the proposed scheme calculates the hash value of the concatenation of random key hash value and symmetric ciphertext, and they can quickly verify the integrity of the results. Finally, under the general group model, the proposed scheme is resistant to selective plaintext attacks and selective keyword attacks. The performance analysis shows that the proposed scheme has reasonable practicability in IoT.

Keywords: Attributed-based Encryption; Internet of Things; Multi-keyword Search; User Revocation; Verifiability

1 Introduction

The development of IoT technology has promoted the interconnection between different devices, such as sensors, mobile devices and RFID, to realize data collection, transmission, storage and management [14]. Along with convenience, the IoT data privacy and security has become the biggest obstacle to its prevalence. When data are outsourced to cloud service provider (CSP), the owner has lost the right to manage data privacy. To solve this problem, it is one of the most feasible approaches to encrypt-then-upload the data. However, traditional encryption schemes could hinder the encrypted data sharing and computing, such as secure search [8,15] or functional evaluation [11,32].

For this reason, Waters and Sahai [25] first introduced a novel cryptographic primitive called as attributed-based encryption (ABE) in 2005, which has the advantage of fine-grained access control over ciphertext. Specially, for a ciphertext-policy attribute-based encryption (CP-ABE) [17], a trusted authority center (AC) can issue a private or secret key associated with the user's attributes for each registered user via a secret channel, a data owner (DO) can embed an access policy to encrypt the data and upload the ciphertext to CSP, and then any data user (DU) can decrypt the ciphertext only if her or his attributes satisfy the embedded access policy. Subsequently, there are many attribute-based encryption schemes and their variants [3,18] for Internet of things.

1.1 Motivations

Although great progresses have been made in ABE for IoT [18,32], there are some fundamental and major challenges to be solved, which may discourage its widespread use. In this paper, we mainly discuss the following issues.

1) Effective revocation: In some ABE systems, taking CP-ABE as an example, the users' secret keys are associated with her/his attributes. When a third party obtains the users' secret keys, it is easy to cause the disclosure of confidential information inside the system. In order to solve the above problem, some literatures [1,4,8] have studied revocable attribute-based encryption (RABE). However, some users still have access for a short time after their access rights were

revoked. Therefore, it is important to research on ABE with efficient and timely revocation for IoT.

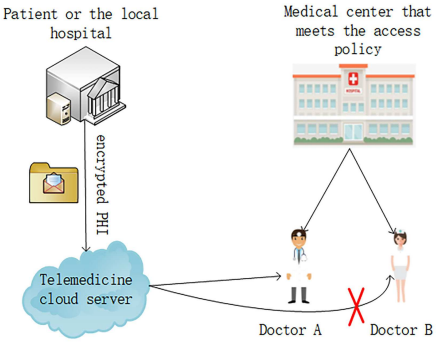


Figure 1: The telemedicine system in IoT

Given an example as shown in Figure 1, when the patient cannot get a good treatment at a local hospital with limited resources, he/she may seek a better treatment through telemedicine. In the telemedi-cine system, the patient and the local hospital manage personal health information (PHI), encrypt PHI under the access policy (Province X, dermatology, telemedicine experience $>$ 5 years), and send to the telemedicine cloud server (TCS). The medical staff in the national or regional medical center are DU who may have the attributes, such as province, department, telemedicine experience, and so on. DU sends the trapdoor of "Province X, dermatology, telemedicine experience $>$ 5 years" to TCS. When TCS receives the trapdoor, the server can test whether the attributes of the medical staff satisfy the access policy of PHI ciphertext. Then, TCS can send the encrypted PHI to the medical staff if the test result is valid. However, after doctor B is terminated and has his/her access right revoked, he/she still can decrypt the PHI ciphertext stored in TCS. The reason is that the doctor B had the insider knowledge and his/her old key. Therefore, it is necessary to prevent the doctor B from continuing to access the PHI ciphertext after he/she was revoked.

2) Accurate search: A traditional method is to download all the ciphertext, decrypt and then search for data of interest. Obviously, the method not only requires a lot of storage space, but also increases the computational costs. As far as we know, there are many literatures [23, 27, 30] that have focused on searchable attribute-based encryption (SABE). However, most of these schemes only supports single-keyword search, and thus returns many redundant search results.

   Considering the telemedicine system that we mentioned above, TCS stored numerous encrypted PHI documents. When the doctor A utilizes a single keyword search, he/she may receive a large number of PHI files, and then spend much time looking for files of interest. If a searchable data sharing scheme can achieve multi-keyword search, then the doctor can obtain the accurate PHI files that contain multi-keyword. Consequently, the patient can receive the treatment from the medical practitioner remotely on the telemedicine platform. Therefore, it is interesting for SABE with multi-keyword in IoT to improve the search accuracy.

3) Integrity verification: Searchable encryption permits users to retrieve IoT data from the encrypted data without decrypting. When performing search operations on the encrypted IoT data or performing predecryption operations on search results, an untrusted CSP may return partial searching results for some reason. In order to solve the above issue, it is necessary to verify the integrity of the searching results. Many existing schemes [13, 16, 34] focused on the integrity verification of ciphertext data, but their efficiency remains to be improved. Therefore, fast and effective verification of IoT data integrity becomes more and more important in SABE for IoT.

1.2 Our Contributions

In this paper, we address searchable and privacy-preserving IoT data sharing by proposing a multi-functional CP-ABE scheme for IoT, which supports user revocation, multi-keyword searchability and data integrity verification. The main contributions can be summarized as follows:

1) We utilized a server-aided to revoke malicious or key exposure users. For each registered attribute user, the authority center generates a transforming key related with attribute set and a secret key, where the first key is sent to CSP for revoking a potential user and pre-processing a ciphertext, and the second key is transmitted to DU for computing a search token and decrypting the ciphertext.

2) We used the multi-keyword searchable encryption technology, which can not only help DU retrieve the interested ciphertext stored in the cloud, but also improve the searching accuracy of the cloud server.

3) Furthermore, only two hash operations are used to verify the integrity of the results returned by the cloud server provider, and thus the verification efficiency is improved.

4) The security of the proposed scheme is proved that the ciphertext is selectively secure against chosen-plaintext attacks in the random oracle model, and the keyword index is indistinguishable under the chosen keyword attacks. The performance analysis shows that the proposed scheme has good practicability in IoT.

1.3 Related Works

Attribute-based encryption (ABE): Rouselakis and Waters [24] introduced the concept of ABE, which was viewed as an expansion of identity-based encryption (IBE) by treating identity as a set of attributes. Up to now, most of CP-ABE schemes can offer secure data access control, but these proposals are still not feasible when used in reality.

In 2013, Jin *et al.* [12] firstly introduced a fine-grained access control scheme based on outsourcing ABE, where two CSPs are used to perform key distribution and decryption, respectively. Since CSP is considered to be honest and curious, it is important to ensure that CSP preprocesses the ciphertext correctly. Lai *et al.* [13] presented a verifiable outsourcing decryption ABE scheme, which can effectively verify the correctness of ciphertext returned by CSP. Furthermore, Li *et al.* [16] put forward an ABE scheme with full verifiability for outsourced decryption, and guaranteed the correctness of outsourcing decryption for all users.

Revocable attribute-based encryption (RABE): In recent years, revocation mechanism has aroused extensive attention. In 2008, Boldyreva *et al.* [4] designed the first scalable and efficient revocable identity-based encryption (RIBE) scheme by using a binary tree data structure. In 2013, Hur *et al.* [11] gave a CP-ABE data sharing scheme that enhanced the security and efficiency, adopted tree access structure to achieve user revocation, and solved the insufficient data integrity verification problem. Then, Attrapadung and Imai [1] introduced a hybrid RABE system that supported direct and indirect revocation. In 2014, Lv *et al.* [19] proposed an ABE scheme that realized effective user revocation by outsourcing key generation and decryption to the mobile cloud environment. In 2015, Qin *et al.* [21] gave a server-aided revocable ABE (SR-ABE) scheme. In 2016, Cui *et al.* [7] presented a SR-ABE scheme, which involved an untrusted server who can help a non-revoked user transfer ciphertext. Recently, Cui *et al.* [8] designed a RABE scheme with keyword search, but did not consider the realistic threat of decryption key exposure on a user's decryption keys. Besides these works, Qin *et al.* [22] gave two SR-ABE schemes, which can resist decryption key exposure attacks. However, the above schemes [7, 19, 21] can not support keyword search.

Searchable attribute-based encryption (SABE): Searchable encryption (SE) was proposed to solve the problem of retrieving ciphertext without revealing plaintext information. SE has two types: symmetric searchable encryption (SSE) and asymmetric searchable encryption (ASE). Song *et al.* [26] first introduced the concept of SSE in 2000. In 2004, Boneh *et al.* [5] proposed the first public-key encryption with keyword search (PEKS) scheme. Qiu *et al.* [23] constructed a SABE scheme with hidden policy, which was secure against keyword attacks in the general group model. Since CSP is semi-trusted, it may return the error search results. Therefore, it is important to ensure the correctness of the returned results. To this end, Chai and Gong [6] proposed the first keyword search scheme that can provide verifiable search. And then, Sun *et al.* [27] proposed a SABE scheme to achieve fine-grained access control and verify encrypted data integrity. However, most schemes [23, 32] only supported single-keyword search. To remedy this problem, a lot of multi-keyword search schemes have been proposed [9, 10, 15]. Furthermore, Li *et al.* [15] constructed a fine-grained multi-keyword SABE, which enhanced user practice and search accuracy, and Huang *et al.* [10] presented a multi-keyword multi-sever search scheme over encrypted cloud storage data, which was proven to be resistant to selection keyword security.

Attribute-based encryption for IoT: The IoT is an important part of the new generation of information technology. The IoT involves users' daily life and work, and thus data sharing, privacy and security have become key issues. In 2015, Lee *et al.* [14] reviewed and analyzed the challenges and opportunities for IoT. And then Yang *et al.* [33] proposed a survey on security and privacy issues in IoT. Furthermore, Wang and Yao [29] combined the notions of SABE and introduced a keyword searchable ABE scheme with equality test for IoT. However, these schemes for IoT cannot support both revocation and search capabilities. In IoT, most of devices that collect and manage data are resource-constrained. To solve this problem, IoT devices may outsource heavy computing tasks to cloud or peripheral devices.

In 2018, Belguith *et al.* [3] proposed a ABE scheme that supports secure outsourced decryption in the IoT constrained environment. Xu *et al.* [32] presented a practical attribute-based access control system for IoT cloud by introducing an efficient RABE that allows DO to efficiently manage the credentials of data users.

1.4 Outlines

In Section 2 some necessary background information will be given. The formal definition of the proposed scheme and security models are developed in Section 3. Then the specific construction is clearly described in Section 4 and the security proof is shown in Section 5. In Section 6, the comparisons of theoretical performance and functionalities with some related works are provided. Finally, we make a conclusion in Section 7.

2 Preliminaries

In this section, some basic cryptographic definitions that will be used in this paper are recalled.

2.1 Bilinear Pairings

Definition 1. *Let $\mathbb{G}$ and $\mathbb{G}_T$ be two cyclic multiplicative groups of prime order p, and g be a generator of $\mathbb{G}$. A bilinear pairing [17] is a map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, which has the following properties:*

1) *Bilinearity:* $\forall \tau, \upsilon \in \mathbb{G}$, *and* $x, y \in \mathbb{Z}_p^*$, $e(\tau^x, \upsilon^y) = e(\tau, \upsilon)^{xy}$.

2) *Non-degeneracy:* $e(g, g) \neq 1$.

3) *Computability:* $\forall \tau, \upsilon \in \mathbb{G}$, *there exists an efficient algorithm to calculate* $e(\tau, \upsilon)$.

2.2 Generic Bilinear Group

Definition 2. *Let* $\psi_1, \psi_2 : \mathbb{Z}_p^+ \to \{0,1\}^k$, *where* $\mathbb{Z}_p^+$ *is an addition group,* $k > 3\log(p)$. *The group* $\mathbb{G} = \{\psi_1(\mathcal{X}) | \mathcal{X} \in \mathbb{Z}_p\}$ *denotes a generic bilinear group [20] and group* $\mathbb{G}_T = \{\psi_2(\mathcal{X}) | \mathcal{X} \in \mathbb{Z}_p\}$, *where* $\psi_1(1) = g$, $\psi_1(\mathcal{X}) = g^{\mathcal{X}}$, $\psi_2(1) = e(g,g)$, *and* $\psi_2(\mathcal{X}) = e(g,g)^{\mathcal{X}}$.

2.3 Complexity Assumption

Rouselakis and Waters [24] introduced a q-type assumption q-1 on prime order bilinear groups, which is similar to the decisional parallel bilinear Diffie-Hellman exponent assumption [31]. Specifically, the q-1 assumption is defined as the following game between a challenger and an attacker:

1) The challenger inputs some security parameters to run the group generation algorithm. Pick an element $g \in \mathbb{G}$ and $q+2$ exponents $a, \mu, b_1, b_2, \cdots, b_q \in \mathbb{Z}_p$ at random. Send the group description $\mathcal{G}(1^\lambda) \to (p, \mathbb{G}, \mathbb{G}_T, e)$ and all of the following terms to the attacker.

$$
\begin{array}{ll}
g, g^{\mu}, & \\
g^{a^i}, g^{b_j}, g^{\mu b_j}, g^{a^i b_j}, g^{a^i b_j^2}, & \forall (i,j) \in [q,q] \\
g^{a^i b_j / b_{j'}^2}, & \forall (i,j,j') \in [2q,q,q] \text{ with } j \neq j' \\
g^{a^i / b_j}, & \forall (i,j) \in [2q,q] \text{ with } i \neq q+1 \\
g^{\mu a^i b_j / b_{j'}}, g^{\mu a^i b_j / b_{j'}^2}, & \forall (i,j,j') \in [q,q,q] \text{ with } j \neq j'
\end{array}
$$

Then flip a random coin $b \leftarrow \{0,1\}$. If $b = 0$, the challenger gives the term $e(g,g)^{\mu a^{q+1}}$ to the attacker. Otherwise, give a random term $R \in \mathbb{G}_T$.

2) The attacker makes a guess $b' \in \{0,1\}$.

Definition 3. *The* q-1 *assumption holds, if all probabilistic polynomial-time (PPT) attackers have at most a negligible advantage in* λ *in the above game, where the advantage is defined as* $Adv = |\Pr[b' = b] - 1/2|$.

2.4 Access Structure and Linear Secret Sharing Scheme

Definition 4. ***(Access Structure)*** *[2] Let* $\mathcal{U}$ *be an attribute universe. An access structure on* $\mathcal{U}$ *is a collection* $\mathbb{A} \subseteq 2^{\mathcal{U}} \setminus \{\emptyset\}$ *of non-empty attribute set. The sets in* $\mathbb{A}$ *are called the authorized sets, and the sets not in* $\mathbb{A}$ *are called the unauthorized sets. An access structure is monotone for* $\forall B, C$, *if* $B \in \mathbb{A}$ *and* $B \subseteq C$, *then* $C \in \mathbb{A}$.

Definition 5. ***(Linear Secret Sharing Scheme (LSSS))*** *[2] Given a prime* p *and an attribute universe* $\mathcal{U}$, *a secret sharing scheme* Π *realizing access structures on* $\mathcal{U}$ *is a linear secret sharing scheme on* $\mathbb{Z}_p$ *if the followings hold.*

1) *The shares of a secret* $\mu \in \mathbb{Z}_p$ *for each attribute constitute a vector on* $\mathbb{Z}_p$.

2) *For every access structure* $\mathbb{A}$ *on* $\mathcal{U}$, *there exists a share-generating matrix* $M \in \mathbb{Z}_p^{l \times n}$ *and a function* $\rho : \{1, \cdots, l\} \to \mathcal{U}$ *that maps each row of* M *to an associate attribute* $\rho(i)$. *Consider a column vector* $\vec{v} = (\mu, r_2, \cdots, r_n)^\top$, *where* $\mu \in \mathbb{Z}_p$ *is a sharing secret among the random constants* $r_2, \cdots, r_n \in \mathbb{Z}_p$. *Then,* $(M \cdot \vec{v})$ *is a vector of* l *shares of* μ *in the light of* Π, *where for* $i \in [l]$ *the share* $\mu_i = (M \cdot \vec{v})_i$ *is corresponding to the attribute* $\rho(i)$. *At the same time,* (M, ρ) *is called a policy of the access structure* $\mathbb{A}$.

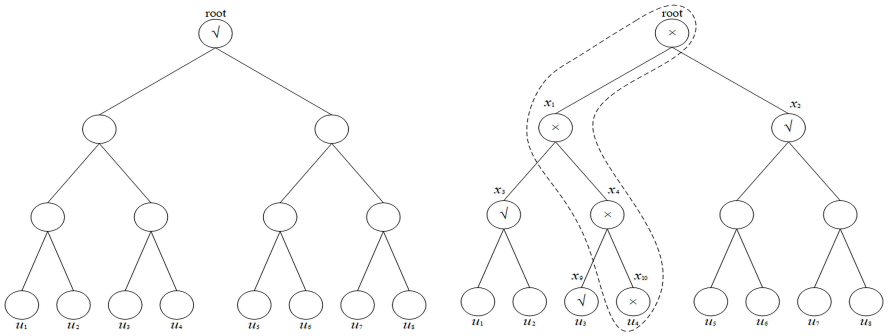
Furthermore, every LSSS satisfies the linear reconstruction property [2, 24]. In this paper, let Π be a LSSS for an access structure $\mathbb{A}$ encoded by a policy (M, ρ), $S \in \mathbb{A}$ be any authorized set, and L be a set of rows defined as $L = \{i \mid i \in [l] \wedge \rho(i) \in S\}$. There exists coefficients $\{\omega_i \in \mathbb{Z}_p\}_{i \in L}$ such that, if $\{\mu_i\}_{i \in L}$ are valid shares of any secret μ according to Π, then $\sum_{i \in L} \omega_i \mu_i = \mu$ holds. But for any unauthorized sets S', no such coefficients $\{\omega_i\}$ exist.

2.5 KUNodes Algorithm

In 2008, Boldyreva *et al.* [4] proposed a KUNodes algorithm, which reduces key update costs and improves revocation efficiency. In this section, we review a binary tree data structure and the KUNodes algorithm.

Let BT be a binary tree with N leaves for N users, and *root* be the root node of BT. If θ is a leaf node, $Path(\theta)$ represents a set of nodes on the path from θ to *root*, including θ and *root*. For a non-leaf node θ, we denote the left and the right children of θ by θ_l and θ_r, respectively. Every user is distributed to a leaf node. Suppose that the nodes in BT are uniquely encoded as strings, and BT is defined by all of its node descriptions. Every node of BT stores a secret value to calculate the key update information.

The algorithm KUNodes is used to calculate a minimum set Y of nodes that need to publish key updates, so that only users who are not revoked at time t can generate time-based transformation keys. The operation process of the algorithm is as follows. The algorithm takes as input a binary tree BT, a revocation list RL, and a time period t, and outputs the minimum set Y of nodes in BT such that the nodes in RL with corresponding time at or before t (users revoked at or before t) have no ancestor (or themselves) in the set, and all other leaf nodes (corresponding to non-revoked users) have only one ancestor (or themselves) in the set. Table 1 describes its formal definition.



(a) No user is revoked: $Y = \{root\}$ (b) User u_4 is revoked: $Y = \{x_2, x_3, x_9\}$

Figure 2: An example of the node selection algorithm KUNodes, where the revoked nodes are marked $\times$ and the unrevoked nodes are marked $\surd$

Table 1: KUNodes(BT, RL, t)

KUNodes(BT, RL, t)
$X, Y \leftarrow \emptyset$.
$\forall(\theta_i, t_i) \in RL$, if $t_i \leq t$, then add $Path(\theta_i)$ to X.
$\forall x \in X$, if $x_l \notin X$, then add x_l to Y;
if $x_r \notin X$, then add x_r to Y.
If $Y = \emptyset$, then add $root$ to Y.
Return Y.

Figure 2 gives an example of the algorithm that marks the ancestors of all revoked nodes as revoked, and then outputs the non-revoked children of all revoked nodes. In the example, let a user u_4 be revoked. Then $X = Path(x_{10}) = \{x_{10}, x_4, x_1, root\}$ and $Y = \{x_2, x_3, x_9\}$.

3 System and Security Model

3.1 System Model

The system model is shown as Figure 3, which includes four entities: Authority Center (AC), Cloud Server Provider (CSP), Data Owner (DO), and Data User (DU).

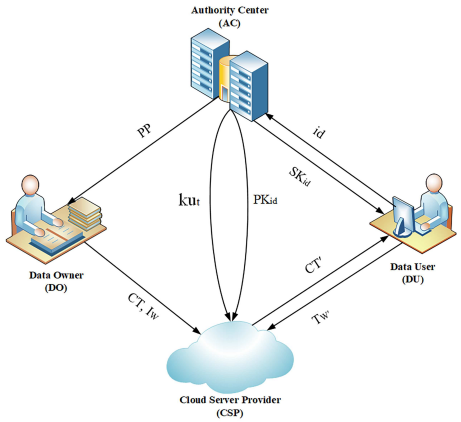


Figure 3: System construction of the proposed scheme.

- **Authority center:** AC is fully trusted by the other three entities, and takes charge of system setup and user management, including registration and revocation.
- **Cloud servers provider:** CSP takes charge of data storage, search, and pre-decryption, but is not fully trusted. After receiving the user's access request, CSP uses the received search token to perform the search task. Once the search token matches the keyword index of some data file, CSP will pre-decrypt the ciphertext, When the user is not revoked and her or his attributes meet the access policy in the ciphertext, the pre-decryption is successful, and then the corresponding ciphertext will be sent to the user. In addition, CSP assisted AC to revoke some users during the user revocation phase.
- **Data owner:** DO can encrypt a data file, extract keyword set, generate an index, get a verification key, and upload ciphertext, keywords index, and verification key to CSP.
- **Data user:** DU can release a search query. If a user is non-revoked, CSP can return the corresponding search results, and DU can verify and decrypt the results. Otherwise, CSP returns nothing.

3.2 Formal Definition

In this section, the formal definition will be described, including a tuple of twelve algorithms, named *Setup*, *KeyGen*, *IndexGen*, *Encrypt*, *Token*, *Search*, *KeyUp*, *TransKG*, *PreDecrypt*, *DecKG*, *Decrypt*, and *Revoke* as follows:

- $Setup(1^\lambda) \rightarrow (PP, MSK, RL, ST)$: AC executes this algorithm, inputs a security parameter λ, and then generates public parameters PP, a master secret key MSK, an empty revocation list RL, and a state ST. Then AC publishes PP.
- $KeyGen(PP, MSK, id, S, ST) \rightarrow (TK_{id}, SK_{id})$: AC runs this algorithm, takes the public parameters PP, the master secret key MSK, a user's identity id, a

set of attributes S, and a state ST as input, and generates a transformation key TK_{id} and a secret key SK_{id}. Then TK_{id} is sent to CSP and SK_{id} is sent to DU.

- $IndexGen(PP, W) \rightarrow I_W$: DO executes this algorithm, inputs the public parameters PP and the keyword set W, and then gets the index I_W of the keyword set.
- $Encrypt(PP, \mathbb{A}, F, t) \rightarrow CT$: DO runs this algorithm, takes the public parameters PP, the access policy $\mathbb{A}$, a data file F, and a time period t as input, and then generates a ciphertext CT. DO sends CT and I_W to CSP.
- $Token(PP, W', SK_{id}) \rightarrow T_{W'}$: DU performs this algorithm, inputs the public parameters PP, a keyword set W' to be searched, and the secret key SK_{id}, and then generates a search token $T_{W'}$, which is sent to CSP.
- $Search(T_{W'}, I_W) \rightarrow (0, 1)$: CSP executes this algorithm, takes the search token $T_{W'}$ submitted by DU, and the keywords index I_W uploaded by DO as input, and then outputs 1 if they match successfully and 0 otherwise.
- $KeyUp(PP, MSK, RL, ST, t) \rightarrow (ku_t, ST)$: AC runs this algorithm, inputs the public parameters PP, the master secret key MSK, an empty revocation list RL, a state ST, and a time period t, and then outputs a key update material ku_t, which is sent to CSP, and an update state ST.
- $TransKG(PP, id, TK_{id}, ku_t) \rightarrow TK_{id,t}$: CSP runs this algorithm, takes the public parameters PP, the user's identity id, a transformation key TK_{id}, and a key update information ku_t as input, and then gets a time-based transformation key $TK_{id,t}$.
- $PreDecrypt(PP, CT, id, S, TK_{id,t}) \rightarrow CT'$: CSP performs this algorithm, inputs the public parameters PP, the ciphertext CT, the user's identity id, a set of attributes S, and a time-based transformation key $TK_{id,t}$, and then outputs a partially decrypted ciphertext CT', which is sent to DU.
- $DecKG(PP, id, SK_{id}, t) \rightarrow DK_{id,t}$: DU runs this algorithm, takes the public parameters PP, the user's identity id, the secret key SK_{id}, and a time period t as input, and then generates a decryption key $DK_{id,t}$.
- $Decrypt(PP, id, DK_{id,t}, CT') \rightarrow F$: DU executes this algorithm, inputs the public parameters PP, the user's identity id, a decryption key $DK_{id,t}$, and a partially decrypted ciphertext CT', and then recovers the data file F.
- $Revoke(id, t, RL, ST) \rightarrow RL$: AC performs this algorithm, takes the user's identity id, a time period t, a revocation list RL, and a state ST as input, and then outputs an updated revocation list RL.

3.3 Security Definitions

The security of the proposed scheme depends on the general bilinear group model and cryptographic assumption. In order to evaluate the security of the proposed scheme, we established three security models by using of three games between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ based on Qin [22], Miao [20], and Qin [21], which are indistinguishability against selective ciphertext-policy and chosen plaintext attack (IND-sCP-CPA) model, indistinguishability against chosen keyword attack (IND-CKA) model, and verifiability model.

3.3.1 IND-sCP-CPA Model

Initialization. The adversary $\mathcal{A}$ declares an access structure $\mathbb{A}^*$ and a time period t^* in advance, which he wishes to challenge upon.

Setup. The challenger $\mathcal{C}$ executes the *Setup* algorithm to generate the public parameters PP, the master secret key MSK, an empty revocation list RL and a state ST. Then it sends PP to the adversary $\mathcal{A}$, and holds MSK, RL and ST privately.

Phase 1. $\mathcal{A}$ would adaptively query the following oracles.

- ***Create(id, S).*** $\mathcal{C}$ runs the *KeyGen* algorithm on (id, S) to obtain (TK_{id}, SK_{id}), adds $(id, S, TK_{id}, SK_{id})$ to a list T, and returns TK_{id} to $\mathcal{A}$.
- ***Corrupt(id).*** If an entry with index id exists in T, $\mathcal{C}$ gets the entry $(id, S, TK_{id}, SK_{id})$ and sets $D = D \cup (id, S)$. Then $\mathcal{C}$ returns SK_{id} to $\mathcal{A}$. Otherwise, it returns $\perp$.
- ***KeyUp(t).*** If $\mathcal{A}$ issues a key update oracles on a time period t, then $\mathcal{C}$ executes the *KeyUp* algorithm and returns ku_t to $\mathcal{A}$.
- ***DecKG(id, t).*** If $\mathcal{A}$ issues a decryption key oracles on (id, t) and T contains an entry indexed by id, $\mathcal{C}$ gains the tuple $(id, S, TK_{id}, SK_{id})$. Then $\mathcal{C}$ performs $DecKG(PP, id, SK_{id,2}, t)$ and returns $DK_{id,t}$ to $\mathcal{A}$. Otherwise, it returns $\perp$.
- ***Revoke(id, t).*** If $\mathcal{A}$ issues a revocation query on (id, t), then $\mathcal{C}$ executes $Revoke(id, t, RL, ST)$ and returns RL to $\mathcal{A}$.

Challenge. $\mathcal{A}$ submits two messages R_1, R_2 with the same size, $\mathbb{A}^*$ and t^* satisfying the following restrictions (Suppose S^* is the set of attribute associated with id^*).

1) If there is $(id^*, S^*) \in D$ and $S^* \in \mathbb{A}^*$, $\mathcal{A}$ must query the revocation oracle on (id^*, t), where $t \leq t^*$.
2) If there is $(id^*, S^*, TK^*_{id}, SK^*_{id}) \in T, S^* \in \mathbb{A}^*$, and id^* is non-revoked at or before time period t^*, then $\mathcal{A}$ cannot query the decryption oracle on (id^*, t^*).

$\mathcal{C}$ selects a bit $\beta \in (0,1)$ at random and sends a challenge ciphertext $CT^* \leftarrow Encrypt(PP, \mathbb{A}^*, R_\beta, t^*)$ to $\mathcal{A}$.

Phase 2. Same as ***Phase 1***.

Guess. The adversary outputs a guess β' of β and wins the game if $\beta' = \beta$.

The advantage of $\mathcal{A}$ in the above game is defined as

$$Adv_{\mathcal{A}}^{IND-sCP-CPA}(\lambda) = \left| \Pr[\beta' = \beta] - \frac{1}{2} \right|.$$

Definition 6. *The proposed scheme is IND-sCP-CPA secure if all PPT adversaries have at most a negligible advantage in λ in the above game.*

3.3.2 IND-CKA Model

Setup. The challenger $\mathcal{C}$ executes the *Setup* algorithm to get public parameters PP, a master secret key MSK, then sends PP to the adversary $\mathcal{A}$, and holds MSK privately.

Phase 1. $\mathcal{A}$ would repeatedly ask the following oracles. $\mathcal{C}$ maintains a keyword set list $\mathcal{L}_W$, whose initial value is empty.

- ***KeyGen***(PP, MSK, id). $\mathcal{C}$ runs the *KeyUp* algorithm, and returns the secret key $SK_{id,1}$ to $\mathcal{A}$.
- ***Token***$(PP, W^*, SK_{id,1})$. For an interested keyword set W^* and the secret key $SK_{id,1}$, $\mathcal{C}$ gets a search token before forwarding it to $\mathcal{A}$, then $\mathcal{C}$ adds W^* to $\mathcal{L}_W$.

Challenge. $\mathcal{A}$ submits two keyword set W_1, W_2 with the same size, where $W_1, W_2 \notin \mathcal{L}_W$. $\mathcal{C}$ picks a bit $\beta \in (0,1)$ at random, gets an I_{W_β} of the keyword set W_β, and then sends I_{W_β} to $\mathcal{A}$. The constraint $W_1, W_2 \notin \mathcal{L}_W$ is that $\mathcal{A}$ can't guess bits β from *Token*.

Phase 2. Same as *Phase 1*, but the keyword set W_1, W_2 cannot be inquired to *Token* oracle.

Guess. The adversary outputs a guess β' of β and wins the game if $\beta' = \beta$.

The advantage of $\mathcal{A}$ in this game is defined as

$$Adv_{\mathcal{A}}^{IND-CKA}(\lambda) = \left| \Pr[\beta' = \beta] - \frac{1}{2} \right|.$$

Definition 7. *The proposed scheme is IND-CKA secure if all PPT adversaries have at most a negligible advantage in λ in the above game.*

3.3.3 Verifiability Model

Setup. The challenger $\mathcal{C}$ executes the *Setup* algorithm to get public parameters PP, a master secret key MSK, then sends PP to the adversary $\mathcal{A}$, and holds MSK privately.

Phase 1. $\mathcal{A}$ would adaptively query the following oracles.

- ***Create(id, S).*** $\mathcal{C}$ runs the *KeyGen* algorithm on (id, S) to obtain (TK_{id}, SK_{id}), adds $(id, S, TK_{id}, SK_{id})$ to a list T, and then returns TK_{id} to $\mathcal{A}$.
- ***Corrupt(id).*** If $\mathcal{C}$ obtains the entry $(id, S, TK_{id}, SK_{id})$ of the id index in list T, sets $D = D \cup (id, S)$, and then returns SK_{id} to $\mathcal{A}$. Otherwise, it returns $\perp$.
- ***Decrypt(id, CT').*** If $\mathcal{C}$ obtains the entry $(id, S, TK_{id}, SK_{id})$ of the id index in list T and returns the output of *Decrypt* algorithm to $\mathcal{A}$. Otherwise, it returns $\perp$.

Challenge. $\mathcal{A}$ submits a file F^* and a challenging access structure $\mathbb{A}^*$. $\mathcal{C}$ computes a challenging ciphertext CT^*=$Encrypt(PP, \mathbb{A}^*, F^*, t)$, and sends it to $\mathcal{A}$.

Phase 2. Same as ***Phase 1***.

Guess. The adversary $\mathcal{A}$ returns the attributes S^* that matches $\mathbb{A}^*$, and CT'.

The advantage of $\mathcal{A}$ in this game is defined as

$$Adv_{\mathcal{A}}^{verif}(\lambda) = \Pr[\mathcal{A} \ succeeds].$$

Definition 8. *The proposed scheme is verifiably secure if all PPT adversaries have at most a negligible advantage in λ in the above game.*

4 The Proposed Scheme

4.1 Specific Construction

Inspired by Qin *et al.*'s server-aided revocable attribute-based encryption (SR-ABE) scheme [22] and Miao *et al.*'s attribute-based keyword search (ABKS) scheme [20], we gives the description of the proposed scheme.

Setup$(1^\lambda) \rightarrow (PP, MSK, RL, ST)$**:** AC chooses two bilinear groups $\mathbb{G}$ and $\mathbb{G}_\mathbb{T}$ of prime order p, where g is a generator of $\mathbb{G}$, gets a bilinear map description $\mathcal{G} = (p, \mathbb{G}, \mathbb{G}_\mathbb{T}, e)$, and selects a symmetric encryption algorithm $(Enc_{SE}(\cdot), Dec_{SE}(\cdot))$. Let $\mathcal{U} = \mathbb{Z}_p$ be an attribute space and $\mathcal{T} = \mathbb{Z}_p$ be a time space. Pick randomly $\alpha, \bar{\alpha}, a \in \mathbb{Z}_p$, $u, h, u_0, h_0 \in \mathbb{G}$, and compute $v_1 = g^a, v_2 = g^{\bar{\alpha}}$. Let RL be an empty list storing the revoked users and BT be a binary tree with at least N leaf nodes. Define $H_0(y) = u^y h, H_1(y) = u_0^y h_0$ as two functions that map any element number y in $\mathbb{Z}_p$ to an element in $\mathbb{G}$. Let $H : \mathbb{G}_T \rightarrow \{0,1\}^{\ell_{SE}}$ be an extractor that is used to obtain a symmetric key, where ℓ_{SE} is the length of the symmetric key, Let $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ and $H_3 : \mathbb{G}_T \rightarrow \{0,1\}^*$ be two hash functions. Then, AC outputs the revocation list RL, the

initial state $ST = BT$, and public parameters $PP = (\mathcal{G}, g, u, h, u_0, h_0, v_1, v_2, Enc_{SE}(\cdot), Dec_{SE}(\cdot), e(g,g)^{\alpha})$, $MSK = (\alpha, \overline{\alpha})$. Finally, AC only publishes PP.

KeyGen$(PP, MSK, id, S, ST) \rightarrow (TK_{id}, SK_{id})$**:** AC runs the key generation algorithm. Pick an exponent $\beta_{id} \in \mathbb{Z}_p$ at random, and set $SK_{id} = (v_1^{\bar{\alpha}}, g^{\beta_{id}})$. Then, AC selects an undefined leaf node θ in the binary tree BT and embeds id on the node. Next, for every node $x \in Path(\theta)$, AC executes as follows:

1) Pick $g_x \in \mathbb{G}$, store g_x on the undefined node x, and calculate $g'_x = g^{\alpha-\beta_{id}}/g_x$.
2) Select $k+1$ random exponents $r_x, r_{x,1}, r_{x,2}, \cdots, r_{x,k} \in \mathbb{Z}_p$, and compute $T_{x,0} = g'_x v_1^{r_x}, T_{x,1} = g^{r_x}$, $T^i_{x,2} = g^{r_{x,i}}$, and $T^i_{x,3} = H_0(A_i)^{r_{x,i}} v_2^{-r_x}$, $i \in [k]$, where $S = \{A_1, A_2, \cdots, A_k\}$.
3) Output the corresponding transformation key and secret key below:

$$TK_{id} = (x, T_{x,0}, T_{x,1}, \{T^i_{x,2}, T^i_{x,3}\}_{i\in[k]})_{x\in Path(\theta)},$$
$$SK_{id} = (SK_{id,1}, SK_{id,2}) = (v_1^{\bar{\alpha}}, g^{\beta_{id}}).$$

Finally, AC sends TK_{id} to CSP and SK_{id} to DU.

IndexGen$(PP, W) \rightarrow I_W$**:** Given a set of files $\mathcal{F} = \{F_1, F_2, \cdots, F_d\}$, and a keyword set $W = \{w_1, w_2, \cdots, w_m\}$ extracted from the data file $F_s (1 \leq s \leq d)$, where m indicates the number of keywords in the keyword set, for each keyword $w_j \in W$, DO randomly selects $\sigma, \tau_s \in \mathbb{Z}_p^*$ for F_s, then computes

$$I = v_2^{\sigma}, \ I_{s,j} = v_1^{(\sigma+\tau_s)} \cdot g^{\sigma \cdot H_2(w_j)}, \ I_s = g^{\tau_s}.$$

Finally, DO generates an index as

$$I_W = \{I_{w_j}\}_{1\leq j\leq m} = \{I, I_{s,j}, I_s\}_{1\leq j\leq m}.$$

Encrypt$(PP, \mathbb{A}, F_s, t) \rightarrow CT$**:** DO picks a random key $R \in \mathbb{G}_{\mathbb{T}}$ and generates an access structure $\mathbb{A} = (M, \rho)$. DO runs as follows:

1) Choose a vector $\vec{v} = (\mu, y_2, \cdots, y_n)^{\top} \in \mathbb{Z}_p^n$ at random and compute $\vec{\lambda} = (\lambda_1, \cdots, \lambda_l)^{\top} = M \cdot \vec{v}$. Then pick l random exponents $\mu_1, \mu_2, \cdots, \mu_l \in \mathbb{Z}_p$, and compute $CT_{\mathbb{A}} = (C, C_0, \{C_{i,1}, C_{i,2}, C_{i,3}\}_{i\in[l]}, C_4)$, where $C = R \cdot e(g,g)^{\alpha\cdot\mu}$, $C_0 = g^{\mu}$, $C_{i,1} = v_1^{\lambda_i} v_2^{\mu_i}$, $C_{i,2} = H_0(\rho(i))^{-\mu_i}$, $C_{i,3} = g^{\mu_i}$, $C_4 = H_1(t)^{\mu}, \forall i \in [l]$.
2) Calculate a symmetric key $K_{SE} = H(R)$ and a label $Tag = H_3(R)$, and compute $C_{SE} = Enc_{SE}(K_{SE}, F_s)$, and $VK_F = H_2(Tag \parallel C_{SE})$.

Finally, DO sends a ciphertext

$$CT = (CT_{\mathbb{A}}, C_{SE}, VK_F)$$

to CSP, where $CT_{\mathbb{A}}$ is a traditional ABE ciphertext, C_{SE} is a symmetric ciphertext, and VK_F is a verifiable key.

Token$(PP, W', SK_{id}) \rightarrow T_{W'}$**:** DU releases a search query for a keyword set $W' = \{w'_1, w'_2, \cdots, w'_{m'}\}$, where m' indicates the number of keyword queried. Choose $\delta \in \mathbb{Z}_p^*$, and calculate

$$t_1 = v_1^{\delta} \cdot \prod_{j=1}^{m'} g^{\delta \cdot H_2(w'_j)}, t_2 = v_2^{\delta}, t_3 = (SK_{id,1})^{\delta} = v_1^{\bar{\alpha}\cdot\delta}.$$

Output a search token as

$$T_{W'} = (t_1, t_2, t_3).$$

Finally, DU sends $T_{W'}$ to CSP.

Search$(T_{W'}, I_W) \rightarrow (0, 1)$**:** When CSP obtained a search token from a data user, CSP can verify whether the following equation holds:

$$e(\prod_{j=1}^{m'} I_{s,j}, t_2) = e(I, t_1) \cdot e(I_s, t_3).$$

If the above equation holds, the keywords in the token matches the keywords in the index. So, CSP returns 1, otherwise 0.

KeyUp$(PP, MSK, RL, ST, t) \rightarrow (ku_t, ST)$**:** Firstly, AC calls the $KUNodes$ algorithm based on the revocation list RL and the state ST in time period t to generate the minimum set of non-revoked user. Then, AC extracts g_x from each node $x \in KUNodes(BT, RL, t)$, picks an exponent $s_x \in \mathbb{Z}_p$, and calculates

$$ku_t = \{x, K_{x,0}, K_{x,1}\}_{x\in KUNodes(BT,RL,t)}$$
$$= \{x, g_x \cdot H_1(t)^{s_x}, g^{s_x}\}_{x\in KUNodes(BT,RL,t)}.$$

Finally, AC sends ku_t to CSP.

TransKG$(PP, id, TK_{id}, ku_t) \rightarrow TK_{id,t}$**:** Assume that an identity id of a data user who submits a search token to CSP is stored in the leaf node θ. Denote $I = \{x | x \in Path(\theta)\}$ and $J = \{x | x \in KUNodes(BT, RL, t)\}$. If $I \cap J = \emptyset$, CSP returns $\perp$. Otherwise, for any node $x \in I \cap J$, CSP computes

$$tk_0 = T_{x,0} \cdot K_{x,0}, tk_1 = T_{x,1}, tk_{2,i} = T^i_{x,2},$$
$$tk_{3,i} = T^i_{x,3}, tk_4 = K_{x,1}, \text{ for } i \in [k].$$

Then CSP outputs the time-based transformation key $TK_{id,t} = (tk_0, tk_1, \{tk_{2,i}, tk_{3,i}\}_{i\in[k]}, tk_4)$.

PreDecrypt$(PP, CT, id, S, TK_{id,t}) \rightarrow CT'$**:** If the *Search* algorithm outputs 1, CSP can pre-decrypt the corresponding ciphertext $CT_{\mathbb{A}}$. If the user is revoked at time period t or the attribute set S does not satisfy the access policy (M, ρ) embedded in $CT_{\mathbb{A}}$, the algorithm outputs $\perp$. Otherwise, CSP runs as follows:

1) Denote $L = \{i | \rho(i) \in S\}$ and $\{\omega_i \in \mathbb{Z}_p\}_{i\in L}$ as a set of constants such that $\sum_{i\in L} \omega_i \lambda_i = (1, 0, \cdots, 0)$ if $\{\lambda_i\}$ are valid shares of μ according to M.

2) Compute

$$B = \frac{e(C_0, tk_0) \cdot e(C_4, tk_4)^{-1}}{\prod_{i \in L}(e(C_{i,1}, tk_1) \cdot e(C_{i,2}, tk_{2,\tau}) \cdot e(C_{i,3}, tk_{3,\tau}))^{\omega_i}},$$

where $\mathbb{A}_\tau = \rho(i)$.

3) Get the pre-decrypted ciphertext

$$CT'_{\mathbb{A}} = (C', C'_0, C'_4) = (\frac{C}{B}, C_0, C_4).$$

Finally, CSP sends $CT' = (CT'_{\mathbb{A}}, C_{SE}, VK_F)$ to DU.

DecKG$(PP, id, SK_{id}, t) \rightarrow DK_{id,t}$**:** Choose an exponent r_t and compute the decryption key

$$\begin{aligned} DK_{id,t} &= (D_0, D_1) = (SK_{id,2} \cdot H_1(t)^{r_t}, g^{r_t}) \\ &= (g^{\beta_{id}} \cdot H_1(t)^{r_t}, g^{r_t}). \end{aligned}$$

Decrypt$(PP, id, DK_{id,t}, CT') \rightarrow F$**:** DU runs the algorithm as follows:

1) Recover the key $R = C' \cdot \frac{e(C'_4, D_1)}{e(C'_0, D_0)}$, and compute $Tag = H_3(R)$.
2) If $H_2(Tag \parallel C_{SE}) \neq VK_F$, DU returns $\perp$ and aborts immediately.
3) Otherwise, DU calculates

$$K_{SE} = H(R),$$

and outputs

$$F_s = Dec_{SE}(K_{SE}, C_{SE}).$$

Revoke$(id, t, RL, ST) \rightarrow RL$**:** If a data user id is revoked at time period t, then adds (x, t) to RL, where x is all nodes associated with the identity id.

4.2 Correctness

1) The correctness of keyword search : $e(\prod_{j=1}^{m'} I_{s,j}, t_2) = e(I, t_1) \cdot e(I_s, t_3)$. First compute the left side as:

$$\begin{aligned} & e(\prod_{j=1}^{m'} I_{s,j}, t_2) \\ &= e(\prod_{j=1}^{m'} v_1^{(\sigma+\tau_s)} g^{\sigma \cdot H_2(w_j)}, v_2^{\delta}) \\ &= e(\prod_{j=1}^{m'} g^{a \cdot (\sigma+\tau_s)} g^{\sigma \cdot H_2(w'_j)}, g^{\bar{\alpha} \cdot \delta}) \\ &= e(g,g)^{a \cdot \bar{\alpha} \cdot \delta \cdot (\sigma+\tau_s)} \cdot e(g,g)^{\sigma \cdot \bar{\alpha} \cdot \delta \cdot \sum_{j=1}^{m'} H_2(w'_j)}. \end{aligned}$$

Then compute the right side as:

$$\begin{aligned} & e(I, t_1) \cdot e(I_s, t_3) \\ &= e(v_2^{\sigma}, v_1^{\delta}) \cdot e(g^{\tau_s}, v_1^{\bar{\alpha} \cdot \delta}) \\ &= e(g^{\bar{\alpha} \cdot \sigma}, g^{a \cdot \delta} \cdot \prod_{j=1}^{m'} g^{\delta \cdot H_2(w'_j)}) \cdot e(g^{\tau_s}, g^{a \cdot \bar{\alpha} \cdot \delta}) \\ &= e(g,g)^{\bar{\alpha} \cdot \sigma \cdot a \cdot \delta} \cdot e(g,g)^{\bar{\alpha} \cdot \sigma \cdot \delta \cdot \sum_{j=1}^{m'} H_2(w'_j)} \\ &\quad \cdot e(g,g)^{\tau_s \cdot a \cdot \bar{\alpha} \cdot \delta} \\ &= e(g,g)^{a \cdot \bar{\alpha} \cdot \delta \cdot (\sigma+\tau_s)} \cdot e(g,g)^{\sigma \cdot \bar{\alpha} \cdot \delta \cdot \sum_{j=1}^{m'} H_2(w'_j)}. \end{aligned}$$

2) The correctness of the file decryption:

$$\begin{aligned} B &= \frac{e(C_0, tk_0) \cdot e(C_4, tk_4)^{-1}}{\prod_{i \in L}(e(C_{i,1}, tk_1) \cdot e(C_{i,2}, tk_{2,\tau}) \cdot e(C_{i,3}, tk_{3,\tau}))^{\omega_i}} \\ &= \frac{e(C_0, tk_0) \cdot e(C_4, tk_4)^{-1}}{\prod_{i \in L}(e(v_1^{\lambda_i} v_2^{\mu_i}, g^{r_x}) \cdot e(H_0(\rho(i))^{-\mu_i}, g^{r_{x,\tau}}))^{\omega_i}} \\ &\quad \cdot \frac{e(C_0, tk_0) \cdot e(C_4, tk_4)^{-1}}{\prod_{i \in L}(e(g^{\mu_i}, H_0(A_\tau)^{r_{x,\tau}} v_2^{-r_x}))^{\omega_i}} \\ &= \frac{e(g^{\mu}, g'_x v_1^{r_x} \cdot g_x H_1(t)^{s_x}) \cdot e(H_1(t)^{\mu}, g^{s_x})^{-1}}{\prod_{i \in L}(e(v_1^{\lambda_i}, g^{r_x}))^{\omega_i}} \\ &= \frac{e(g^{\mu}, g^{\alpha - \beta_{id}} v_1^{r_x})}{e(v_1, g^{r_x})^{\sum_{i \in L} \lambda_i \omega_i}} \\ &= \frac{e(g^{\mu}, g^{\alpha - \beta_{id}} v_1^{r_x})}{e(v_1, g^{r_x})^{\mu}} \\ &= e(g^{\mu}, g^{\alpha - \beta_{id}}), \end{aligned}$$

$$\frac{C}{B} = \frac{R \cdot e(g,g)^{\alpha\mu}}{e(g^{\mu}, g^{\alpha - \beta_{id}})} = R \cdot e(g^{\mu}, g^{\beta_{id}}),$$

$$C' \cdot \frac{e(C'_4, D_1)}{e(C'_0, D_0)} = R \cdot e(g^{\mu}, g^{\beta_{id}}) \cdot \frac{e(H_1(t)^{\mu}, g^{r_t})}{e(g^{\mu}, g^{\beta_{id}} H_1(t)^{r_t})} = R.$$

5 Security Analysis

5.1 IND-sCP-CPA Security

Inspired by Qin *et al.* [22], we prove the proposed scheme to be IND-sCP-CPA secure. Furthermore, the IND-sCP-CPA security of the proposed scheme will be reduced to the q-1 assumption.

Theorem 1. *The proposed scheme can be selectively attacked with a negligible advantage by all PPT adversaries with a challenge matrix of size $l^* \times n^*$, where $l^*, n^* \leq q$, assuming that the q-1 assumption holds in $\mathbb{G}$ and $\mathbb{G}_T$.*

Proof. To prove the theorem, we will assume if there exists a PPT adversary $\mathcal{A}$, which can attack the security of the proposed scheme with a non-negligible advantage $Adv_{\mathcal{A}}$ under the selective security model, then a PPT simulator $\mathcal{B}$ can be constructed to solve the q-1 assumption with a non-negligible advantage.

Initialization. $\mathcal{A}$ sends the challenging access structure $\mathbb{A}^* = (M^*, \rho^*)$ and time period t^* to $\mathcal{B}$, where M^* is an $l^* \times n^*$ matrix with $l^*, n^* \leq q$.

Setup. $\mathcal{B}$ has to furnish $\mathcal{A}$ the public parameters PP of the system. $\mathcal{B}$ picks $\tilde{\alpha} \in \mathbb{Z}_p$ randomly, and calculates $e(g,g)^{\alpha} = e(g^a, g^{a^q}) \cdot e(g,g)^{\tilde{\alpha}}$, which implicitly sets MSK to be $\alpha = a^{q+1} + \tilde{\alpha}$.

$\mathcal{B}$ selects $a, b, \tilde{u}, \tilde{h}, \tilde{v_2} \in \mathbb{Z}_p$, computes $u_0 = g^a v_1, h_0 = g^b v_1^{-t^*}$, and gives $PP = (D, g, u, h, u_0, h_0,$

$v_1, v_2, e(g,g)^{\alpha})$ to $\mathcal{A}$:

$$g = g, \quad u = g^{\tilde{u}} \cdot \prod_{\tilde{j},\tilde{k} \in [l^*, n^*]} (g^{a^{\tilde{k}}/b_{\tilde{j}}^2})^{M^*_{\tilde{j},\tilde{k}}},$$
$$h = g^{\tilde{h}} \cdot \prod_{\tilde{j},\tilde{k} \in [l^*, n^*]} (g^{a^{\tilde{k}}/b_{\tilde{j}}^2})^{-\rho^*_{\tilde{j}} \cdot M^*_{\tilde{j},\tilde{k}}},$$
$$v_1 = g^{a}, \quad v_2 = g^{\tilde{u}} \cdot \prod_{\tilde{j},\tilde{k} \in [l^*, n^*]} (g^{a^{\tilde{k}}/b_{\tilde{j}}})^{M^*_{\tilde{j},\tilde{k}}}.$$

Phase 1. $\mathcal{B}$ chooses an undefined leaf node θ^*, which stores the target user id^*. For any node $x \in Path(\theta^*)$, $\mathcal{B}$ chooses a random exponent $\alpha_x \in \mathbb{Z}_P$ and sets $g_x = g^{\alpha+\alpha_x}$. Then $\mathcal{A}$ would adaptively query the following oracles.

Create(id, S). $\mathcal{A}$ queries key generation oracle on (id, S), where $S = (A_1, A_2, \cdots, A_k)$. $\mathcal{B}$ creates a list T with elements $(id, S, TK_{id}, SK_{id})$ and sends TK_{id} to $\mathcal{A}$ as following:

Case 1: If $S \in \mathbb{A}^*$, $\mathcal{B}$ sets $id^* := id$, stores id^* in the node θ^*, which is pre-assigned, and computes $SK_{id^*,2} = g^{\beta_{id^*}}$ (the exponent β_{id^*} is chosen randomly). For any $x \in Path(\theta^*)$, $\mathcal{B}$ extracts $g_x = g^{\alpha+\alpha_x}$ and computes $g'_x = g^{\alpha-\beta_{id}}/g_x = g^{-\beta_{id^*}-\alpha_x}$. Then $\mathcal{B}$ chooses $k+1$ exponents $r_x, r_{x,1}, r_{x,2}, \cdots, r_{x,k} \in \mathbb{Z}_p$, and computes $TK_{id^*} = \{x, T_{x,0}, T_{x,1}, T^i_{x,2}, T^i_{x,3}\}_{x \in Path(\theta^*), i \in [k]}$ as follows:

$$T_{x,0} = g'_x v_1^{r_x}, \ T_{x,1} = g^{r_x}, \ T^i_{x,2} = g^{r_{x,i}}, \ T^i_{x,3} = H_0(A_i)^{r_{x,i}} v_2^{-r_x}, \ \text{for } i \in [k].$$

Case 2: If $S \notin \mathbb{A}^*$, $\mathcal{B}$ stores id in a node θ, which is an undefined, sets $SK_{id,2} = g^{\beta_{id}}$ (the exponent β_{id} is chosen randomly). So $\mathcal{B}$ implicitly sets $r = \tilde{r} + \omega_1 a^q + \omega_2 a^{q-1} + \cdots + \omega_n a^{q+1-n}$, where $\tilde{r} \in \mathbb{Z}_p$ is a random number and r is correctly distributed. Then $\mathcal{B}$ selects the suitable terms from the challenging problem to calculate:

$$Q_0 = g^{\alpha} v_2^r, \ Q_1 = g^r, \ Q_{i,2} = g^{r_i}, \ Q_{i,3} = H_0(A_i)^{r_i} v_2^{-r}, \ \text{for } i \in [k].$$

For any $x \in Path(\theta)$, $\mathcal{B}$ extracts g_x from the node x. If x is undefined, it picks $g_x \in \mathbb{G}$. $\mathcal{B}$ chooses $k+1$ exponents $r_x, r_{x,1}, r_{x,2}, \cdots, r_{x,k} \in \mathbb{Z}_p$, and computes $TK_{id} = \{x, T_{x,0}, T_{x,1}, T^i_{x,2}, T^i_{x,3}\}_{x \in Path(\theta), i \in [k]}$ as follows:

1) For $x \in Path(\theta) \cap Path(\theta^*)$, $\mathcal{B}$ has g_x and g'_x, and computes TK_{id} as ***Case 1.***
2) For $x \notin Path(\theta) \cap Path(\theta^*)$, $\mathcal{B}$ has g_x, chooses randomly $r \in \mathbb{Z}_p$, and computes

$$T_{x,0} = g_x^{-1} g^{-\beta_{id}} \cdot Q_0 \cdot v_1^{r_x} = \frac{g^{\alpha-\beta_{id}}}{g_x} \cdot v_1^{r+r_x} = g'_x v_1^{r+r_x},$$
$$T_{x,1} = Q_1 \cdot g^{r_x} = g^{r+r_x},$$
$$T^i_{x,2} = Q_{i,2} \cdot g^{r_{x,i}} = g^{r_i + r_{x,i}},$$
$$T^i_{x,3} = Q_{i,3} \cdot H_0(A_i)^{r_{x,i}} v_2^{-r_x} = H_0(A_i)^{r_i + r_{x,i}} v_2^{-(r+r_x)}, \text{for } i \in [k].$$

Corrupt(id). If $\mathcal{B}$ obtains the entry $(id, S, TK_{id}, SK_{id})$ of the list T, $\mathcal{B}$ sets $D = D \cup (id, S)$ and returns SK_{id} to $\mathcal{A}$. Otherwise, $\mathcal{B}$ returns $\perp$. (Note that $\mathcal{A}$ is allowed to corrupt (id^*, SK_{id^*}).)

KeyUp(t). If $\mathcal{A}$ issues a key update query at a time period t, $\mathcal{B}$ can compute ku_t as below. For all $x \in KUNodes(BT, RL, t)$, $\mathcal{B}$ extracts g_x from the node x. If x is undefined, $\mathcal{B}$ picks $g_x \in \mathbb{G}$ at random and puts it on x.

1) If $x \notin Path(\theta^*)$, $\mathcal{B}$ has g_x. Then $\mathcal{B}$ picks an exponent $s_x \in \mathbb{Z}_p$ at random, and compute $K_{x,0} = g_x \cdot H_1(t)^{s_x}, K_{x,1} = g^{s_x}$.
2) If $x \in Path(\theta^*)$, $\mathcal{B}$ does not have g_x (α is unknown). Recall that $\mathcal{B}$ has α_x so can compute ku_t as follows. $\mathcal{B}$ chooses $S \notin \mathbb{A}^*$, compute $(Q_0, Q_1) = (g^{\alpha} v_2^r, g^r)$ ($r \in \mathbb{Z}_p$ is unknown), and sets implicitly $s_x = \frac{r}{t-t^*}$ (Note that $t \neq t^*$. Otherwise, $x \notin Path(\theta^*)$ and $(id, t) \in RL$). Thus $\mathcal{B}$ can compute $K_{x,0} = g^{\alpha_x} \cdot g^{\alpha} v_1^r \cdot (g^r)^{\frac{a \cdot t + b}{t-t^*}} = g_x \cdot H_1(t)^{s_x}$, $K_{x,1} = (g^r)^{\frac{1}{t-t^*}} = g^{s_x}$.

Finally, $\mathcal{B}$ returns ku_t to $\mathcal{A}$.

DecKG(id, t). If $\mathcal{B}$ obtains the corresponding $DK_{id,t}$ for the entry $(id, S, TK_{id}, SK_{id})$ of the list T, $\mathcal{B}$ can answer $\mathcal{A}$'s decryption key queries directly. Otherwise, $\mathcal{B}$ returns $\perp$.

Revoke(id, t). If $\mathcal{A}$ issues a revocation query on (id, t), then $\mathcal{B}$ inserts (id, t) to RL.

Challenge. The adversary submits two messages R_0, R_1 with the same size, for unknown $\mu \in \mathbb{Z}_p$, $\mathcal{B}$ picks at random bit $b \in \{0, 1\}$, computes $C = R_b \cdot \Upsilon \cdot e(g,g)^{\alpha \cdot \mu}$, $C_0 = g^{\mu}$, where Υ is the challenging term and g^{μ} is the relevant term in the q-1 assumption. $\mathcal{B}$ implicitly sets $\vec{y} = (\mu, \mu a + \tilde{y}_2, \mu a^2 + \tilde{y}_3, \cdots, \mu a^{n-1} + \tilde{y}_n)^{\top}$, where $\tilde{y}_2, \tilde{y}_3, \cdots, \tilde{y}_n \in \mathbb{Z}_p$ are randomly chosen. Note that μ and $\vec{y}$ are correctly distributed, and μ is information theoretically hidden from $\mathcal{A}$. Since $\vec{\lambda} = M^* \cdot \vec{y}$, for $i \in [l]$, we can get $\lambda_i = \sum_{\tau \in [n]} M^*_{i,\tau} \mu a^{\tau-1} + \sum_{\tau=2}^{n} M^*_{i,\tau} \tilde{y}_{\tau} = \sum_{\tau \in [n]} M^*_{i,\tau} \mu a^{\tau-1} + \tilde{\lambda}_i$. So, $\mathcal{B}$ calculates

$$C_{i,1} = v_1^{\lambda_i} v_2^{\mu_i}, \ C_{i,2} = H_0(\rho(i))^{-\mu_i}, \ C_{i,3} = g^{\mu_i}, \ C_4 = H_1(t^*)^{\mu} = (g^{\mu})^{a \cdot t^* + b} = (C_0)^{a \cdot t^* + b}, \ \text{for } i \in [l].$$

Finally, $\mathcal{B}$ returns the challenging ciphertext $CT^*_{\mathbb{A}} = (C, C_0, \{C_{i,1}, C_{i,2}, C_{i,3}\}_{i\in[l]}, C_4)$ to $\mathcal{A}$.

Phase 2. Same as ***Phase 1***.

Output. $\mathcal{A}$ outputs a guess b' of b. If $b' = b$, $\mathcal{B}$ outputs 0, which declares that the challenging term $\Upsilon = e(g,g)^{\mu a^{q+1}}$. Otherwise, $\mathcal{B}$ outputs 1. This is to say that Υ is a random term R of $\mathbb{G}_T$. If the challenging term is a random term of $\mathbb{G}_T$, then all the information about the R_b is lost in the challenging ciphertext. That is to say $Adv_{\mathcal{A}} = 0$. Therefore, if $\mathcal{A}$ can break the security game with a non-negligible advantage, $\mathcal{B}$ can break the q-1 assumption with a non-negligible advantage.

□

5.2 IND-CKA Security

Inspired by Miao *et al.* [20], we prove the proposed scheme to be IND-CKA secure.

Theorem 2. *The proposed scheme is keyword ciphertext indistinguishability under chosen-keyword attacks in the generic bilinear group model, assuming that H_2 is a one-way hash function.*

Proof. In the IND-CKA game, the goal of adversary $\mathcal{A}$ is to distinguish $v_1^{\sigma+\tau_s} g^{\sigma\cdot H_2(w_0)}$ from $v_1^{\sigma+\tau_s} g^{\sigma\cdot H_2(w_1)}$, that is, $g^{a\cdot(\sigma+\tau_s)} g^{\sigma\cdot H_2(w_0)}$ from $g^{a\cdot(\sigma+\tau_s)} g^{\sigma\cdot H_2(w_1)}$. For the input a number $f \in \mathbb{Z}_p$, $\mathcal{A}$ has the same advantage in distinguishing g^f from $g^{a\cdot(\sigma+\tau_s)} g^{\sigma\cdot H_2(w_0)}$ as it distinguishes g^f from $g^{a\cdot(\sigma+\tau_s)} g^{\sigma\cdot H_2(w_1)}$. In simple terms, we assume that $\mathcal{A}$ can distinguish g^f from $g^{a\cdot(\sigma+\tau_s)}$ in a modified IND-CKA game between $\mathcal{A}$ and $\mathcal{C}$ as shown below.

Setup. The challenger $\mathcal{C}$ picks $a, \bar{\alpha} \in \mathbb{Z}_p^*$, runs *Setup* algorithm to generate the public parameters $PP = (g, g^a, g^{\bar{\alpha}})$ and the master secret key $MSK = \bar{\alpha}$, sends PP to the adversary $\mathcal{A}$, and then holds MSK privately.

Phase 1. $\mathcal{A}$ would repeatedly query the following oracles.

- ***KeyGen***(PP, MSK, id). $\mathcal{C}$ computes $SK_{id,1} = v_1^{\bar{\alpha}} = g^{a\cdot\bar{\alpha}}$ and $PP = (g, g^a, g^{\bar{\alpha}})$, then sends $SK_{id,1}$ to $\mathcal{A}$.
- ***Token***$(PP, W^*, SK_{id,1})$. For PP, W^* and $SK_{id,1}$, $\mathcal{C}$ picks $\delta \in \mathbb{Z}_p^*$ at random, outputs a search token $T_{W^*} = (t_1, t_2, t_3)$ of the keyword set W^*, where $t_1 = g^{a\cdot\delta} \cdot \prod_{j=1}^{m'} g^{\delta\cdot H_2(w'_j)}$, $t_2 = g^{\bar{\alpha}\cdot\delta}$, $t_3 = (SK_{id,1})^{\delta} = g^{a\cdot\bar{\alpha}\cdot\delta}$, and adds W^* to $\mathcal{L}_W$.

Challenge. The adversary submits two keyword sets W_1, W_2 with the same size, where $W_1, W_2 \notin \mathcal{L}_W$. $\mathcal{C}$ picks $\sigma, \tau_s \in \mathbb{Z}_P^*$, and selects a bit $\beta \in \{0,1\}$ at random. If $\beta = 1$, $\mathcal{C}$ returns $I = g^{\bar{\alpha}\cdot\sigma}$, $I_{s,j} = g^{a\cdot(\sigma+\tau_s)}$, $I_s = g^{\tau_s}$. Otherwise, output $I = g^{\bar{\alpha}\cdot\sigma}$, $I_{s,j} = g^f$, $I_s = g^{\tau_s}$.

Phase 2. Same as ***Phase 1***, but the keyword sets W_1, W_2 cannot be inquired to the *Token* oracle.

Guess. If $\mathcal{A}$ could construct $e(g,g)^{h'\cdot a\cdot(\sigma+\tau_s)}$ for the term $g^{h'}$ returned by the query, then $\mathcal{A}$ can distinguish g^f from $g^{a\cdot(\sigma+\tau_s)}$. However, we still have to prove that $\mathcal{A}$ can get $e(g,g)^{h'\cdot a\cdot(\sigma+\tau_s)}$ for $g^{h'}$ with a negligible advantage in the IND-CKA game.

We utilize two injective functions ψ_1, ψ_2, where $\mathbb{G} = \{\psi_1(\mathcal{X}) \mid \mathcal{X} \in \mathbb{Z}_p\}$, $\mathbb{G}_T = \{\psi_2(\mathcal{X}) \mid \mathcal{X} \in \mathbb{Z}_p\}$, based on the generic group model, to obtain that the advantage of distinguishing elements between maps ψ_1 and ψ_2 be negligible.

Then, we probe into $\mathcal{A}$'s advantage in constructing $e(g,g)^{h'\cdot a\cdot(\sigma+\tau_s)}$ from $g^{h'}$. Since σ can only be obtained from $\bar{\alpha}\cdot\sigma$, h' should contain $\bar{\alpha}$ in order to get $e(g,g)^{h'\cdot a\cdot(\sigma+\tau_s)}$. That is, given $h' = h''\cdot\bar{\alpha}$, $\mathcal{A}$ will try to build $e(g,g)^{h''\cdot\bar{\alpha}\cdot a\cdot(\sigma+\tau_s)}$. However, $\bar{\alpha}$ is the master key for $\mathcal{C}$, thus $\mathcal{A}$ cannot get $e(g,g)^{h''\cdot\bar{\alpha}\cdot a\cdot(\sigma+\tau_s)}$ in any way. Furthermore, $\mathcal{A}$ cannot get $e(g,g)^{h'\cdot a\cdot(\sigma+\tau_s)}$ from $g^{h'}$ in any way.

At last, we draw the conclusion that $\mathcal{A}$ cannot distinguish g^f from $g^{a\cdot(\sigma+\tau_s)}$, so $\mathcal{A}$ is unlikely to distinguish $g^{a\cdot(\sigma+\tau_s)} g^{\sigma\cdot H_2(w_0)}$ from $g^{a\cdot(\sigma+\tau_s)} g^{\sigma\cdot H_2(w_1)}$. As a result, $\mathcal{A}$ cannot break IND-CKA games with a non-negligible advantage. □

5.3 Verifiability Security

Inspired by Qin *et al.* [21], we prove the security of ciphertext integrity verification.

Theorem 3. *The proposed scheme can verify the integrity of ciphertext, assuming that two hash functions H_0, H_1 are collision-resistant.*

Proof. Given that an adversary $\mathcal{A}$ break the verifiability, we construct a simulator $\mathcal{B}$ to attack the collision-resistance of the hash function H_0 or H_1.

Setup. $\mathcal{B}$ runs the *Setup* algorithm, then gives $PP = (D, g, u, h, u_0, h_0, v_1, v_2, e(g,g)^{\alpha})$ to $\mathcal{A}$, and holds the master key MSK privately.

Phase 1. $\mathcal{A}$ would adaptively query the following oracles.

- ***Create****(id, S)*. Same as in ***Phase 1*** in IND-sCP-CPA security.
- ***Corrupt****(id)*. Same as in ***Phase 1*** in IND-sCP-CPA security.
- ***Decrypt****(id, CT′)*. If $\mathcal{B}$ obtains the entry $(id, S, TK_{id}, SK_{id})$ of the list T, and returns the output of $Decrypt(id, DK_{id,t}, CT')$ to $\mathcal{A}$. Otherwise, $\mathcal{B}$ returns $\perp$.

Challenge. The adversary submits a message F^*. The simulator computes a challenging ciphertext CT^* by

running $Encrypt(PP, \mathbb{A}^*, R^*, t)$, where $R^* \in F$, and sends CT^* to $\mathcal{A}$. Then $\mathcal{B}$ sets

$$Tag_0^* = H_0(R^*), K_{SE}^* = H(R^*),$$

and also gets

$$C_{SE}^* = Enc_{SE}(K_{SE}^*, F^*), Tag^* = H_1(Tag_0^* \parallel C_{SE}^*).$$

Finally, $\mathcal{B}$ returns the challenging ciphertext $CT^* = (CT_{\mathbb{A}^*}^*, C_{SE}^*, VK_F^* = Tag^*)$ to $\mathcal{A}$.

Phase 2. Same as ***Phase 1***.

Guess. The adversary outputs the attributes S^* that satisfies $\mathbb{A}^*$, and CT'. If $\mathcal{A}$ can be against the verifiability, $\mathcal{B}$ will recover a file $F \notin \{F^*, \perp\}$ by running *Decrypt* algorithm. Observe that the decryption algorithm outputs $\perp$ if $H_1(Tag_0 \parallel C_{SE}) \neq Tag^*$, where $Tag_0 = H_0(R)$ and $R = Decrypt(DK_{S^*}, CT'_{\mathbb{A}})$. So, we consider the following two situations:

- ***Type 1.*** Since $\mathcal{B}$ has (Tag_0^*, C_{SE}^*), if $(Tag_0, C_{SE}) \neq (Tag_0^*, C_{SE}^*)$, $\mathcal{B}$ directly gets a collision of the hash function H_1.
- ***Type 2.*** If $(Tag_0, C_{SE}) = (Tag_0^*, C_{SE}^*)$, but $R \neq R^*$. Note that $H_0(R) = Tag_0 = Tag_0^* = H_0(R^*)$. So, $\mathcal{B}$ breaks the collision-resistance of H_0.

Therefore, we conclude that $\mathcal{A}$ cannot verify the ciphertext integrity, if two hash functions (H_0, H_1) are not collision resistant. Namely, when H_0 or H_1 is a collision-resistant hash function, $\mathcal{A}$ cannot break ciphertext integrity verification games with a non-negligible advantage. □

6 Performance Evaluations

The scheme contains many mathematical symbols, in order to improve the readability of the paper. We define the following notations to denote the operation or value (as shown in Table 2).

Table 2: Notations in the scheme

Notations	Descriptions
E	Time cost of an exponential operation
P	Time cost of a pairing operation
H	Time cost of a hash operation
n_a	Number of all attributes
$n_{a,u}$	Number of attributes owned by a user
$\|\mathbb{G}\|$	Bit length of the element in $\mathbb{G}$
$\|\mathbb{Z}_p\|$	Bit length of the element in $\mathbb{Z}_p$

6.1 Comparisons of Functionality

In Table 3, we compared the proposed scheme with some existing schemes in terms of access control, revocation, multiple keyword search, result verification, decryption outsourcing, and resist key exposure, etc. As shown in Table 3, these schemes [8, 23, 27, 28] can achieve multi-keyword search, but cannot resist key exposure. Furthermore, Xu *et al.*'s scheme [32] can resist key exposure, Lai *et al.*'s scheme [13] can verify ciphertext integrity, and Wang *et al.*'s scheme [29] can realize the revocation function, but these schemes cannot achieve multi-keyword search. The proposed scheme can not only realize the above functions, but also have high efficiency in computation cost and storage cost as shown in Table 4 and Table 5. In the setup phase, the computational complexity of the proposed scheme is independent of the attributes, since the proposed scheme can support large universe. In addition, since only two hash functions are needed in the ciphertext integrity verification, the proposed scheme is more efficient and practical.

6.2 Experimental Results

Furthermore, we give the experimental results of the proposed scheme and the existing schemes [27–29,32]. We implement the above schemes in software based on Pairing-based Cryptography (PBC) Library with the Type A elliptic curve ($y^2 = x^3 + x$ with 512 bits based filed size and the group order p is 160 bits). The hardware platform for execution is 2.90 GHz Intel(R) Core(TM) i5-9400 CPU with 8GB RAM running 64 bit Windows 10.

Figure 4 shows the computation cost of the *Setup* algorithm, *KeyGen* algorithm, *Encrypt* algorithm, *Token* algorithm, *Search* algorithm, *PreDecrypt* algorithm, and *Decrypt* algorithm. As shown in Sub-figure (a), (d)-(g), the experimental results show that the computation time does not increase with the number of attributes. From Sub-figure (b), (c), the experimental results show that the time required for calculation changes little with the increase of attributes. In summary, the proposed scheme has the advantage of performance as shown in Figure 4.

7 Conclusions

In this paper, we have proposed an attribute-based encryption scheme for Internet of Things, which can realize the functionality of user revocation and local decryption key exposure resistance, implement multi-keyword search without reducing the search efficiency, and verify the integrity of the search results. The proposed scheme is proven to be selective plaintext security, selective keyword attack security and data verifiability. Furthermore, compared with the existing schemes, the experiment results show that the proposed solution is more suitable for applying to IoT in view of the diversity of functionalities.

Acknowledgments

This work is supported by the National Key R&D Program of China under Grant No.2017YFB0802000, the National Natural Science Foundation of China under Grants

Table 3: The comparisons of functionality

Schemes	Access Control	Revocation	Multiple Keyword Search	Result Verification	Decryption Outsourcing	Resist Key Exposure
Cui *et al.* [8]	Tree	User	Yes	No	-	No
Lai *et al.* [13]	LSSS	-	No	Yes	Yes	No
Qiu *et al.* [23]	AND-gate	-	Yes	No	No	No
Sun *et al.* [27]	AND-gate	User	Yes	Yes	-	No
Wang et al [28]	LSSS	Attribute	Yes	Yes	Yes	No
Wang et al [29]	LSSS	Attribute	No	No	Yes	No
Xu *et al.* [32]	LSSS	User	No	No	No	Yes
Ours	LSSS	User	Yes	Yes	Yes	Yes

Table 4: The comparisons of computation

Operation	Sun *et al.* [27]	Wang *et al.* [28]	Wang *et al.* [29]	Xu *et al.* [32]	Ours
Setup	$(1+3n_a)E+P$	$(3+n_a)E+P$	$4E+2P$	P	$3E+P$
KeyGen	$(3+2n_a)E$	$(6+n_{a,u})E$	$(10+10n_{a,u})E$	$(3+2n_{a,u})E$	$(4+n_{a,u})E$
Encryption	$(2+n_{a,u})E+P$	$(6+3n_{a,u})E$	$(7+6n_{a,u})E$	$(4+3n_{a,u})E$	$(2+3n_{a,u})E$
Token	$(1+2n_{a,u})E$	$4E$	$4E$	-	$3E$
Search	$E+(n_a+1)P$	$4P$	$4P$	-	$3P$
Verification	-	$3P$	-	-	$2H$
Decryption by CSP	-	$2E+2P$	$2E+2P$	-	$E+P$
Decryption by user	-	$2E+P$	$2E+2P$	$3P$	$2P$

Table 5: The comparisons of storage

Operation	Sun *et al.* [27]	Wang *et al.* [28]	Wang *et al.* [29]	Xu *et al.* [32]	Ours
Setup	$(2+3n_a)\|\mathbb{G}\|+(1+3n_a)\|\mathbb{Z}_p\|$	$4\|\mathbb{G}\|+2\|\mathbb{Z}_p\|$	$11\|\mathbb{G}\|+6\|\mathbb{Z}_p\|$	$(5+n_a)\|\mathbb{G}\|+2\|\mathbb{Z}_p\|$	$7\|\mathbb{G}\|+2\|\mathbb{Z}_p\|$
KeyGen	$(1+2n_{a,u})\|\mathbb{G}\|+\|\mathbb{Z}_p\|$	$(5+n_a)\|\mathbb{G}\|+\|\mathbb{Z}_p\|$	$(5+9n_{a,u})\|\mathbb{G}\|$	$(2+2n_{a,u})\|\mathbb{G}\|$	$(5+n_{a,u})\|\mathbb{G}\|$
IndexGen	$(2+n_{a,u})\|\mathbb{G}\|$	$4\|\mathbb{G}\|$	$4\|\mathbb{G}\|$	-	$3\|\mathbb{G}\|$
Encryption	-	$(2+2n_{a,u})\|\mathbb{G}\|$	$(3+8n_{a,u})\|\mathbb{G}\|$	$(3+4n_{a,u})\|\mathbb{G}\|$	$(4+n_{a,u})\|\mathbb{G}\|$
Token	$(1+2n_{a,u})\|\mathbb{G}\|+\|\mathbb{Z}_p\|$	$4\|\mathbb{G}\|$	$4\|\mathbb{G}\|$	-	$3\|\mathbb{G}\|$

No.61807026, the Natural Science Basic Research Plan in Shaanxi Province of China under Grant No.2019JM-198, the Plan For Scientific Innovation Talent of Henan Province under Grant 184100510012, and in part by the Program for Science and Technology Innovation Talents in the Universities of Henan Province under Grant 18HASTIT022.

References

[1] N. Attrapadung and H. Imai, "Attribute-based encryption supporting direct/indirect revocation modes," in *IMA International Conference on Cryptography and Coding*, pp. 278–300, 2009.

[2] A. Beimel, "Secure schemes for secret sharing and key distribution," *Research Thesis*, 1996. (`https://dphu.org/uploads/attachements/books/books_1542_0.pdf`)

[3] S. Belguith, N. Kaaniche, M. Laurent, A. Jemai, and R. Attia, "PHOABE: Securely outsourcing multi-authority attribute based encryption with policy hidden for cloud assisted IoT," *Computer Networks*, vol. 133, pp. 141–156, 2018.

[4] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-based encryption with efficient revocation," in *Proceedings of the 15th ACM Conference on Computer and Communications Security*, pp. 417–426, 2008.

[5] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 506–522, May 2004.

[6] Q. Chai and G. Gong, "Verifiable symmetric searchable encryption for semi-honest-but-curious cloud servers," in *IEEE International Conference on Communications*, pp. 917–922, 2012.

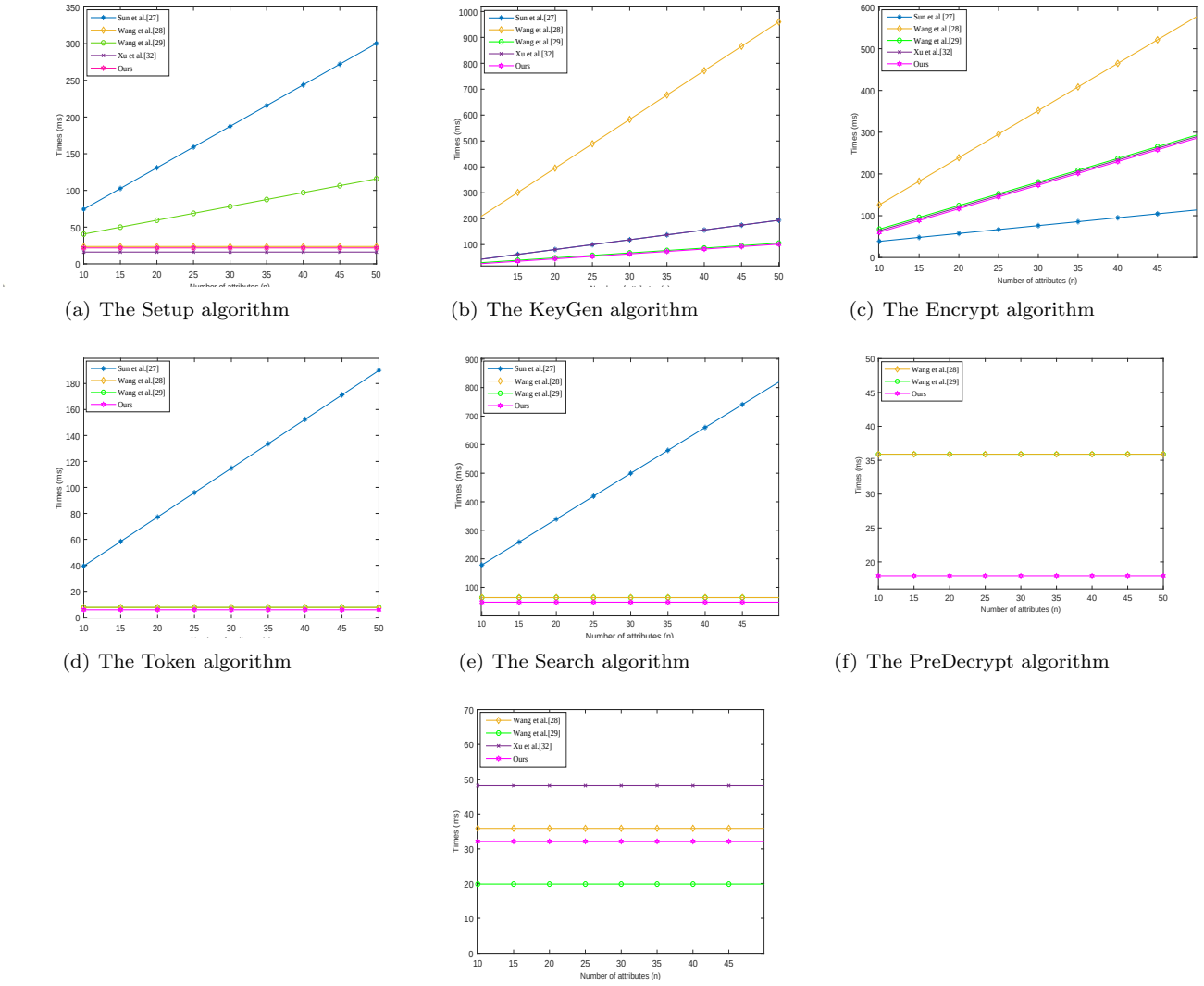
[7] H. Cui, R. H. Deng, Y. Li, and B. Qin, "Server-aided revocable attribute-based encryption," in *The 21st European Symposium on Research in Computer Security*, pp. 570–587, 2016.

[8] J. Cui, H. Zhou, H. Zhong, and Y. Xu, "Akser: Attribute-based keyword search with efficient revocation in cloud computing," *Information Sciences*, vol. 423, pp. 343–352, 2018.

[9] M. Hu, H. Gao, and T. Gao, "Secure and efficient ranked keyword search over outsourced cloud data by chaos based arithmetic coding and confusion," *International Journal of Network Security*, vol. 21, no. 1, pp. 105–114, 2019.

[10] H. Huang, J. Du, H. Dai, and R. Wang, "Multi-sever multi-keyword searchable encryption scheme based on cloud storage," *Journal of Electronics and Information Technology*, vol. 39, no. 2, pp. 389–396, 2017.

[11] J. Hur, "Improving security and efficiency in attribute-based data sharing," *IEEE Transactions*



(a) The Setup algorithm

(b) The KeyGen algorithm

(c) The Encrypt algorithm

(d) The Token algorithm

(e) The Search algorithm

(f) The PreDecrypt algorithm

(g) The Decrypt algorithm

Figure 4: Experimental results for the proposed scheme

on Knowledge and Data Engineering, vol. 25, no. 10, pp. 2271–2282, 2013.

[12] L. Jin, X. Chen, J. Li, C. Jia, J. Ma, and W. Lou, "Fine-grained access control system based on outsourced attribute-based encryption," in *European Symposium on Research in Computer Security*, vol. 8134, pp. 592–609, 2013.

[13] J. Lai, R. H. Deng, C. Guan, and J. Weng, "Attribute-based encryption with verifiable outsourced decryption," *IEEE Transactions on Information Forensics and Security*, vol. 8, pp. 1343–1354, Aug. 2013.

[14] I. Lee and K. Lee, "The internet of things (IoT): Applications, investments, and challenges for enterprises," *Business Horizons*, vol. 58, no. 4, pp. 431–440, 2015.

[15] H. Li, Y. Yang, T. H. Luan, X. Liang, L. Zhou, and X. S. Shen, "Enabling fine-grained multi-keyword search supporting classified sub-dictionaries over encrypted cloud data," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, pp. 312–325, May/June 2016.

[16] J. Li, Y. Wang, Y. Zhang, and J. Han, "Full verifiability for outsourced decryption in attribute based encryption," *IEEE Transactions on Services Computing*, vol. 13, pp. 478–487, May 2020.

[17] J. Li, W. Yao, J. Han, Y. Zhang, and J. Shen, "User collusion avoidance cp-abe with efficient attribute revocation for cloud storage," *IEEE Systems Journal*, vol. 12, pp. 1767–1777, June 2018.

[18] Q. Li, H. Zhu, J. Xiong, R. Mo, and H. Wang, "Fine-grained multi-authority access control in IoT-enabled mhealth," *Annals of Telecommunications*, vol. 74, no. 4, pp. 389–400, 2019.

[19] Z. Lv, J. Chi, M. Zhang, and D. Feng, "Efficiently attribute-based access control for mobile cloud storage system," in *IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications*, pp. 292–299, Sep. 2014.

[20] Y. Miao, J. Ma, Q Jiang, X. Li, and A. K. Sangaiah, "Verifiable keyword search over encrypted cloud data in smart city," *Computers and Electrical Engineering*, vol. 65, pp. 90–101, 2018.

[21] B. Qin, R.H. Deng, S. Liu, and S. Ma, "Attribute-based encryption with efficient verifiable outsourced decryption," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 7, pp. 1384–1393, 2015.

[22] B. Qin, Q. Zhao, D. Zheng, and H. Cui, "(Dual) server-aided revocable attribute-based encryption with decryption key exposure resistance," *Information Sciences*, vol. 490, pp. 74–92, 2019.

[23] S. Qiu, J. Liu, Y. Shi, and R. Zhang, "Hidden policy ciphertext-policy attribute-based encryption with keyword search against keyword guessing attack," *Science China Information Sciences*, vol. 60, no. 5, pp. 052105, 2017.

[24] Y. Rouselakis and B. Waters, "Practical constructions and new proof methods for large universe attribute-based encryption," in *Proceedings of ACM SIGSAC Conference on Computer & Communications Security*, pp. 463–474, 2013.

[25] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Proceedings of The 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 457–473, May 2005.

[26] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Proceeding of IEEE Symposium on Security and Privacy*, pp. 44–55, May. 2002.

[27] W. Sun, S. Yu, W. Lou, Y. T. Hou, and H. Li, "Protecting your right: Verifiable attribute-based keyword search with fine-grained owner-enforced search authorization in the cloud," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, pp. 1187–1198, Apr. 2016.

[28] S. Wang, S. Jia, and Y. Zhang, "Verifiable and multi-keyword searchable attribute-based encryption scheme for cloud storage," *IEEE Access*, vol. 7, pp. 50136–50147, 2019.

[29] S. Wang, L. Yao, J. Chen, and Y. Zhang, "Ksabeswet: A keyword searchable attribute-based encryption scheme with equality test in the internet of things," *IEEE Access*, vol. 7, pp. 80675–80696, 2019.

[30] S. Wang, D. Zhang, Y. Zhang, and L. Liu, "Efficiently revocable and searchable attribute-based encryption scheme for mobile cloud storage," *IEEE Access*, vol. 6, pp. 30444–30457, 2018.

[31] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *International Workshop on Public Key Cryptography*, pp. 53–70, 2011.

[32] S. Xu, G. Yang, Y. Mu, and X. Liu, "A secure IoT cloud storage system with fine-grained access control and decryption key exposure resistance," *Future Generation Computer Systems*, vol. 97, pp. 284–294, 2019.

[33] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, "A survey on security and privacy issues in internet-of-things," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1250–1258, 2017.

[34] K. Zhang, J. Gong, S. Tang, J. Chen, and Z. Cao, "Practical and efficient attribute-based encryption with constant-size ciphertexts in outsourced verifiable computation," in *Acm on Asia Conference on Computer and Communications Security*, pp. 269–279, May 2016.

Biography

Zhenhua Liu received the B.S. degree from Henan Normal University, in 2000, and the M.S. and Ph.D. degrees from Xidian University, China, in 2003 and 2009,respectively, where he is currently a Professor. His research interests include cryptography and information security.

Fangfang Yin received the B.S. degree from Henan Normal University, in 2018. She is currently pursuing the M.S. degree in applied mathematics with Xidian University, China. Her research interests include cryptography and cloud security.

Jiaqi Ji received the B.S. degree from Taiyuan Normal University, in 2017. She is currently pursuing the M.S. degree in applied mathematics with Xidian University, China. Her research focuses on network and information security.

Baocang Wang received the B.S. degree in computational mathematics and the M.S. and Ph.D. degrees in cryptology from Xidian University, China, in 2001, 2004, and 2006, respectively, where he is currently a Professor and a Ph.D. Supervisor. His research interests include post-quantum cryptography, fully homomorphic cryptography, number theoretic algorithms, and cloud security.

A Pilot Study on Survivability of Networking Based on the Mobile Communication Agents

Awais Akram[1], Ren Jiadong[1], Tahir Rizwan[2], Muhammad Irshad[1], Sohail M. Noman[1,3], Jehangir Arshad[4], and Sana Ullah Badar[4]

(Corresponding author: Sohail M. Noman)

Department of Computer science and Technology, Yanshan University, PR China[1]
38 Hebei Street West Section, Haigang District, Qinhuangdao, Hebei, China
Department of Automation, Shanghai Jiao Tong University, PR China[2]
Department of Cell Biology and Genetics, Shantou University, Shantou, Guangdong, PR China[3]
Department of Electrical and Computer Engineering, COMSATS University, Pakistan[4]
(Email: mn.sohail@hotmail.com)

(Received Jan. 19, 2020; revised and accepted June 8, 2020)

Abstract

The word coping is characterized as a work-state or a network's ability to provide essential services inside deterministic values. The bulk of networks is unbounded, so they ignore a formal administrative control and have a single security policy framework. Survival training can require such infinite systems to provide essential services while retaining key features, such as privacy, integrity, and efficiency, during failure. The network will adapt and adjust to changes within the network system with self-aware management. This paper outlines the solutions and survival strategies in a network and illustrates how the self-aware architecture handles IPQoS? This significantly discusses the problems of a functioning wireless network and the self-healing methods used in wireless systems.

Keywords: Ad-hoc Routing; Asymmetric Channel; Connectivity and Stability; Low Probability of Detection (LPD); Satellite; Survivability; Wireless Network

1 Introduction

This paper focuses on the survivability of the network infrastructure by considering the detailed revision of various articles on a single platform. In addition, this paper allows the automation of a key network survivability features. The major revised articles includes the theories from Sundeep Selvaraj [33], James Sterbenz *et al.* [37] , Suk Yu Hui *et al.* [15], and Xianghui Liu *et al.* [25]. This paper describes boundaries in a survivable network by modelling of self-healing devices, self-configuration, self-supply, and self-monitoring facilities in the network survivability. The relevant descriptions identify the key characteristics which is accompanied by self-aware management, a policy based QoS management, and strategies to implement future network resources. An agent approach allows building a complex, sophisticated system using modular components and self-aware system can manage the processes itself, which defines different levels, including mediators of access, computer, resource, and network elements. In addition, a survivable wireless network has different challenges as wireless communication travels through an unknown channel, unlike the error free cable distribution. Furthermore, security is essential for survivable networks. This paper also discusses how protection and reliability of a network are preserved.

Network systems have gained drastic significance over the past two decades on different segments of daily life like health, education, travelling, and so on. All these sectors operates and works on the network systems to fulfil their scope globally. Since the network came into reality, people stated realizing about the consequences of network failure which reflect the working habits [10, 38, 39]. Hence, the active precautions came to existence in order to overcome the consequences of critical failure of systems and networks. These overcomes depend on the accurate findings of services in the network on time. Automating these network systems monitoring is critical due to factors like users demand for quality of service, and expense involved in hiring professionals. Therefore, human intervention in network management and process automation is critical which is often called a control plan and management plan. When implementing these strategies, assessing network system operational objectives is very significant which should be allowed with tracking and alteration strategies and techniques [7, 22]. In addition, the network survivability maintenance becomes difficult since the time of global internet services involved. Since central administration is absent, and defense is complicated in an unbounded network. Although such networks lack central administration, but the independent services are

more reliable if goes with the proper techniques [2,3,6,38].

2 Designing

The networking environment can be divided into two categories of network infrastructure named bounded and unbounded in creating a survivable framework. Both system parts are regulated and entirely controlled by a single administrative entity in a bounded system, while the unbounded network does not have a centralized control of the system parts [30]. There, the administrative body implies the authority to carry out certain activities in the network rather than a delegate that proposes various solutions.

In addition, the framework is said to have an eternal lifespan in an environment with multiple administrative domains. Online, for instance, can be seen as a boundless environment. The Internet is a collection of many device and network applications. For a public web server, customers in many administrative domains can be available on the Internet. All customers are not regulated fairly by any central authority [21,28]. A web server can therefore never rely on a certain client. In this case, the system is the web server and client. The number of website domains are multiple administrative domains. Most domains have legitimate users, and for anonymous interference, different platforms are used. Such sites cannot be differentiated by their administrative domain but by customer behaviour that is specified by a hypertext transport protocol, a relationship between server and user [2,18,22,27].

Furthermore, the system of web servers and clients is widely distributed across the globe. Legitimate users and attackers are both members of the same community, so it is difficult to isolate those legitimate users from attackers. In other terms, it is quite difficult to attach an area to these legitimate users in a common administrative procedure. Therefore, security is an essential element in the survivable network today.

2.1 Survivable Features of the Network

One of the dominant features of the surviving network is to ensure its survival and to provide essential services, even in case of failure, while preserving other essential properties, such as integrity, secrecy, efficiency and other essential qualities that play an important role in maintaining a balance of multi-quality attributes, such as perforation. The ability of a system to provide essential services while retaining its essential properties continues even if a major portion of the system is not functional. In fact, the next important aspect of their existence is to identify essential services and properties within a specific operating system [5,18,19].

The surviving financial sector maintains integrity, confidentiality and availability, even if an attack/accident causes a certain node or communication connection, of key information such as account and loan information and financial services such as transaction validation and processing. It must be able to retrieve this information and services leaked promptly. The key functionality of the system is to adapt to the environment and provide essential services. The ultimate idea is to carry out the system's task without always creating a functional system component.

2.2 Management Based on Policy

Policy-based administration (PBM) distinguishes knowledge of resource management and information related to the state. This enables an operator to develop coverage objectives and policies that will be followed by potential network infrastructure. Judgment on the resource allocation and configuration can therefore be made locally autonomously. The policy based administration of internet engineering task force (IETF) provides an infrastructure for the management of IP networks with service guarantees. In this context, the technology introduced operates IP networks providing guarantees of service [3, 8, 23]. In addition, this infrastructure also allows for flexible network conduct.

Further, this reacts differently to different network events based on the defined policy. Such protocols are but a set of rules governing access to network resources and regulating them. It allows network managers or service providers to control their network behaviour based on criteria such as user identity or type of application. Practices at different levels can also be defined. The IETF and DMTF (distributed management task force) develops an alternative significant model called the Policy Core Information Model (PCIM) to see the Network as a state machine using state transition control policies [12, 29]. It can identify and monitor states. This model also defines priority roles and the order of performance.

2.3 Agencies Approach

Agency approach is one of the promising features of the survivable network that enables the installation of a complex or sophisticated device with modular components. Intelligent components are often called agents, which are considered to be the core of the multi-agent system. A simple and responsible network process execution software can be used by an agent. It can also have some automatic functional knowledge. Smart agents generally cooperate between user interfaces and smart processes to perform certain common tasks. Agents are therefore accountable for the detection, resolution and infrastructure development as expected. These properties are independent, but also responsible for adapting and distributing networks [20, 35]. The agent-based approach also aims to introduce mobile and responsible agents to deal with the dynamic nature of the network system. The key part of this approach is coordination with other systems and the transfer of research to other intelligent agents to reduce the network connectivity load.

2.4 Self-conscious Infrastructure for Management

The ability to maintain management processes and the associated network infrastructure without some external assistance can be described as self-aware management. The self-conscious management structure includes basic elements such as configuration, optimization, healing, and protection. In fact, self-confident control architecture is built using the concepts of PBM and multi-agent applications [5, 19, 28, 29]. This architecture enables complex service management efficiency within the context. It is also consistent with the IST CADENUS project architecture (creation and deployment of Premium User Services), which consists of access mediators, service mediators, and resource mediators. This standard has established a service level agreement (SLA) on the basis of a three-way framework which includes suitable end-user services [4, 13].

The Access Mediator shall be primarily responsible for the cooperation among end - users and different service providers and shall also have awareness of and conduct end - users, access links and terminal sort in order to access the service provider. In addition, it offers the user a larger choice of services at the lowest cost, simplifying the selection process and informing the user immediately if a new service is convenient.

All new service offers will be notified by the Service Mediator. It is also responsible for maintaining visual access to the resources through a relevant underlying network using the relevant resource mediator. There is no direct contact with the SLA end users by the service mediators. It covers their composition with other service providers and the support of their services with network providers.

The resource mediator manages the network performance according to demand of service providers. The Policy Decision Point (PDP) also plays a role in the policy-based management environment. In addition, the policy of rules to be applied in the network components that are met by the service mediators is identified. The primary role of PDP in this architecture is to send policies to the network level that cannot be implemented directly by network elements. Political rules consist usually of kinds, for, on and on.

2.5 Wireless Networking Should Endure

The environment via which wireless communication travels is unpredictable, unlike error - free transmission. To mention a few environmental radio frequencies, wireless communication may be unreliable due to the noise generated by powerful engines, other wireless devices, microwaves and air moisture content. Wireless networks are manually configurable and follow traditional wired models. This means that it must be programmed to connect the node or transceptors to a specific node, which is usually a central base station. The main challenge is to stop the communication if the node loses contact with its peer. These nodes have been placed in optimal space to compensate for this drawback. But even this decision could not guarantee reliability because the environment today can change [16].

In addition, the most important advances in cellular self - healing was ad hoc networks. They are autonomous, self-organizing and instantly reconfigured without human interference when contact between transceivers fail or break down. These networks can have links or interfaces to other networks such as Ethernet or 802.11 [24]. The key strength of such design is that a base station or central control point is not necessary. Growing node is an endpoint and router for other nodes in the decentralized network. This naturally increases reliability and scalability of the network. Automated analysis by link, road exploration and evaluation of network self - healing algorithms remains the most prominent features. By way of discovery, networks create one or more routes between the sender and the recipient of the message. Throw away route failures, trigger renewed discovery and select the best route for the message through the Evaluation Networks.

Moreover, the wireless network of healing itself is typically pro-active or on-demand, has unique paths and a dynamic routing framework. Such features affect speed, delivery, resource and electricity consumption in different quantities. Continuously updating and reconfiguring positive research networks [11]. They believe that frequent link breaks and changes in performance occur and are structured to continuously explore and strengthen optimum connections. Proactive exploration takes place when nodes assume that each route is viable and try to find it. On request however, the discovery only defines routes that higher-level software requires, allowing the nodes to save bandwidth and resources and preserve the traffic-free network.

Further, sometimes the Dynamic Routing is used to predetermine the end-to-end route and messages are forwarded to all neighbours and transmitted in accordance with a cost scheme. While this routing system has the advantage of several complementary routes from source to end point, it generates much network trade. A gradient routing of Ad - hoc networks means that wireless networks provide full dynamic routing [13]. The routing of GRAd stresses the possibility of redundant routes to maximize the lowest latencies between originators and destinations. GRAd deletes message loops by returning a response to that network traffic when the request reaches the destination. Maintaining multiple routes increases memory costs and network traffic, but flexibility and efficiency in the delivery of messages increase the return.

3 Evaluation

The survival of the network is an important aspect of reliable communication services. Survival consists not only of robustness against natural failures, accidents or unintended operational errors, but also of failures due to malice, especially in the context of military networks. Cell wireless networks provide ubiquitous computing and unthread Internet access, but they pose a significant challenge to survival both because users are cell and communications accessible to everyone [34]. This segment also discusses the problems, obstacles and research proposals in cellular sustaining networks as a consequence of our participation in a study program of DARPA's Mobile Wireless Information Networks.

3.1 Resilience, Regeneration, Appreciation, and Reconstruction

Survival focuses on the delivery and maintenance of essential services. Essential services and equipment are the device features necessary to the accomplishment of mission objectives. Resistance, recognition and rehabilitation depend on three key abilities. Resistance is the ability of a system to repel attacks. Recognition is the ability to detect attacks, assess damage and compromise and recovery characteristics, the ability to provide essential services and assets for attacks, to limit damage levels and to restore full post - attack services. We extend this definition further to require survivable systems to quickly incorporate lessons from failures, develop and adapt to emerging threats [4, 12]. We call this a refinement of survival. We may categorize survivable wireless networking specifications into four categories, including opposition, identification, recovery and enhancement requirements.

Moreover, the survival criterion can be defined as a specification technique based on software requirement definition processes. This includes identifying program and security criteria, permissible and invasive applications, development needs, operational requirements and evolution criteria. Essential services and resistance, identification and recovery requirements must be established for entry, exploration and exploitation phases of the assault. Both methods have driven the research and are proposed for further study of mobile wireless networks in the future [13]. Finally, two distinct aspects of sustainability span all networking levels. One is access to information and the second is contact from end-to-end.

The customer would accept obtaining information or services required to complete the task in the event of failure or assault in the event of access requirements. For example, when the network is partitioned, will services or information be replicated and distributed locally? End-to-end interactions should not be expected in these circumstances. In addition, interactive applications and interpersonal communications such as voice calls or dynamically generated information are available in the context of end-to-end communication requirements. Are current sessions surviving? Does the user should create new sessions to reach the desired point of contact even with crashes and attacks? It requires protection of the communication endpoints, and the adversary cannot divide the network indefinitely. In addition, the opponent cannot permanently disable access to necessary services, such as routing, authentication, discovery of resources or naming.

3.2 Military and Cellular Network Survivability

In order to support military operations, use of Wireless Networking technologies imposes strict security and operational obligation on technology such as Transmission Safety (TRANSEC, COMSEC), Authorization and Access Controls, Network Infrastructure Safety, Robustness and Performance. Furthermore, the current work on cell phone network survival focuses mainly on infrastructure survival and does not take into account adversarial attacks. They provide insight into the survival quantification and function of network management tools. Networks, especially software, are vulnerable during upgrades [9,17,31]. Therefore, rapid evolution leads to learning curve problems and over-concentration of traffic or services in single failure points.

Moreover, deficits of operating and maintaining increasingly complex systems in network management tools exacerbate this problem. Deployment failures (for example, fiber backup circuits) will deactivate fault tolerance designs. The use of reliable networks (e.g. SONET-rings), multi-mode systems, and overlay networks to improve survival requires cell technological improvements [1, 24]. Historically, fixed and wireless providers were various administrative bodies, and the reliability of radio links was poor and low expectations increased. Reliability and survival issues will become increasingly important in future cellular networks.

4 Wireless AD-HOC Network Safety

4.1 Connectivity Protections

The first big survival goal is to establish and sustain a network as shown in the Figure 1, where possible. This allows traditional routing and end-to-end protocols to be carried out. The goal is to stay steady [26, 40, 41].

4.2 Foundation Assumptions

There are two separate forms of thought about the area of all-round wireless networking. One approach (for example, Mobile IP) often depends on pre-configured networks. The other (ad-hoc networks) solution suggests that all nodes operate a common ad hoc routing protocol and that there are no networks. There is a small mix of heterogeneous wired and wireless networks. The practice

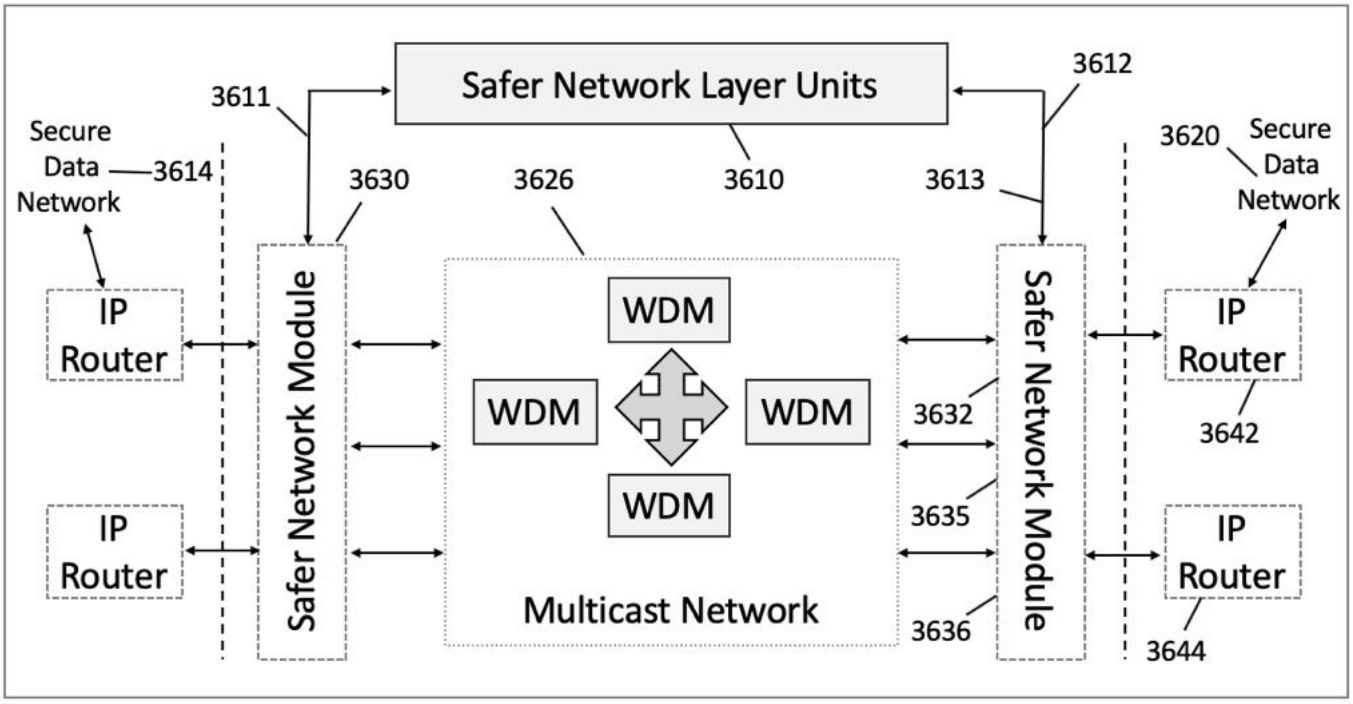


Figure 1: Survivable connectivity flow structure

of quasi-static naming of nodes and subnetworks on IP networks is one justification [32, 36].

In addition, the exploration and self-configuration of current networks is not feasible with new technologies. In any research system, mechanical fullback modes that enable ad-hoc networking of the nodes of Internet terminals do not exist. Such a multimodal service is essential for survivable mobile networking, which has an efficient and smooth transition between basic ad-hoc and infrastructure modes [41]. By seamlessly transportation and application sessions must survive switching between infrastructure and ad-hoc modes.

4.3 Auto-configuration Network Layer

Most self-configuration research is about naming and device creation in heterogeneous networks. They presume that each network node has both an address and a routing scheme in advance. This affects overlays at application level rather than network bootstrap. Wired network self-configuration is usually limited to a DHCP, Zeroconf, or existing unique host identification system such as IPv6 [14]. Such approaches and strategies involve unique resources or network identifiers. Surviving nodes must address the problem of auto-configuring mission-based naming, routing and signalling in secure network layer. Safe wireless network automated configuration remains complex because no suitable approaches except for address-less routing approaches such as diffusion routing for specific applications and single shared, probabilistic or gossip-based protocols are established. There is need of establishing further studies in this scope.

4.4 Private Sensor Network

Many ad-hoc networks that already exist do not require private nodes. For each node it assumes unique identifiers such as the Ethernet MAC address or IPv6 EUI-64 identification. Common identities present other concerns about health and confidentiality. Knowing an identifier of a node does not necessarily reveal the identity of the user or owner, but may provide hints that pose unacceptable risks for topology or traffic analysis. An anonymous network cannot determistically assign global unique IDs. The only way to avoid this is to specify an initiator or to allow random ties. There may be some clustering strategies for certain methodologies, such as amorphic calculation, anonymous address networks and spare wireless networks [34].

4.5 Statistical Odds of Detection

The low probability of detection, interception and misuse of (LPD/LPI/LPE), that is a capability of an enemy to monitor and manipulate radio energy, is paramount for most ad - hoc strategic networks. Few techniques can be used to obscure the radio signature of a node, including hidden waveforms, space diffusions and lower transmitter capacity. Survival increases as the network becomes rugged to future opponents. But it makes legal inter-

actions more difficult; lower transmission power generally increases the probability of detection of both enemies and valid nodes [13]. In fact, strategic networks must be able to deny topology information to opponents.

5 Discussion and Potential Direction

This section describes two technologies, include centered drivers for the adaptation and connection of adaptive and satellite networks with a view to dynamic environments. This eliminates the need for standardization and determination of the full range of algorithms, protocols and hard code in nodes prior to deployment. Only a structure for the exploration of nodes and protocol agreements must be established beforehand; radio software is a key technology.

Despite the standardization and pre-known program, application and task conditions, mobile wireless networks are inherently complex and allow for volatile channel conditions. So, network nodes and protocols that learn and adapt to their environment are required for surviving networks. The next step is semantic networking, allowing potential nodes and networks to know about their environment and to take measures to improve sustainability. In addition, the effects of weakly linked channels and node mobility can be minimized by satellites and UAVs. Satellites and other airborne nodes offer unique features similar to ground-based nodes. The high altitude of a satellite allows a very large terrestrial footprint where every ground node can communicate and communicate optionally to the satellite.

Furthermore, this benefit, together with the so called intrinsic transmit power of the satellite, allows the satellite to connect to a large number of earth nodes, providing a broader spectrum than earth nodes. A satellite occurs at a predetermined space point at a mobile node. Satellites are geostationary (GEO), whereas small satellites (MEO) and low-earth satellites (LEO) have computerized trajectories. UAVs may have predictable pathways (e.g. shape of tracks). Where satellite footprint represents cluster or cell size, handovers between node and satellite are more uncommon than between ground-based node and base station, while gaps in retrieval and registration are minimised. The altitude that protects the satellite from overrunning (physical attack) also restricts node mobility.

In particular, survivable networks require more than conventional reliability and fear to loss. While considerable progress has been made on the creation and maintenance of connected networks, further research is required to understand trade-offs towards stealth criteria (LPI/LPD/LPE). In addition, surviving mobile wireless networks require that asymmetric, weakly connected and episodically disconnected links are not defects, but first class citizens. Agility is also essential and used to improve survival. We suggest a significant improvement in how routing algorithms manage communication, encouraging potential networking in areas where this is not currently possible.

In addition, research has begun to scratch the surface because it is not possible or practical to a priori to anticipate the contact environment. It is important that network nodes and protocols respond to their communication scenario or tasks. Dynamically flexible protocols, algorithms and parameters using active networking and software radio technology are key enablers for this functionality. Aerial nodes like satellites and UAVs also provide innovative networks in order to alleviate the impacts of isolated and asymmetric connections and mobility. Moreover, the problems of 5G network testing are identified and defined. Problems as the machine, telecommunications and services are grouped into three regions. Some challenges, including multimode terminals, wireless system discovery, application compatibility and QoS support, are well explored. Nevertheless, some are less known. These include selection of systems, stability, breakdown and life. Research is also required in 5G networks to incorporate personal independence, billing and accounting structures. The discussion in this article not only demonstrates that much work needs to be done of terms of transitioning to 5G systems but also illustrates the need to incorporate current systems so that 5G technology can be implemented smoothly. 5G networks will not start quickly without these infrastructures.

6 Conclusions

To sum up, the results from recent studies, including modelling various self-healing devices to assess serviceability, QoS guarantees for self (configuration, supply, and monitoring facilities), are reviewed in this paper. The theory of agents described in this paper allows automation a key element of survivable networks. Notwithstanding efforts to maintain safety in unlimited networks, the preservation ethos contributes to tightening security in unlimited networks. Survivable Network also has interesting areas for further research.

Acknowledgments

The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] S. E. Abkari, A. Jilbab, and J. E. Mhamdi, "Wireless indoor localization using fingerprinting and trilateration," *IJCSNS International Journal of Computer Science and Network Security*, vol. 20, no. 5, p. 131, 2020.

[2] R. S. Alonso, I. Sittón-Candanedo, S. Rodríguez-González, Ó. García, and J. Prieto, "A survey on software-defined networks and edge computing over

iot," in *International Conference on Practical Applications of Agents and Multi-Agent Systems*, pp. 289–301, 2019.

[3] J. Arshad, S. Salim, T. Younas, M. D. Amentie, G. Farid, A. U. Rehman, and A. Khokhar, "A study on device automation: An integration of internet protocols and embedded system," in *International Conference on Engineering and Emerging Technologies (ICEET'20)*, pp. 1–6, 2020.

[4] O. Awotayo, "Information systems strategies for small and medium size enterprise sustainability," *Walden University*, 2020. (https://search.proquest.com/openview/9f61763613d800f87a620cdc68d0933b/1?pq-origsite=gscholar&cbl=18750&diss=y)

[5] P. Baumann and M. M. Keupp, "Assessing the reliability of street networks: A case study based on the swiss street network," in *The Security of Critical Infrastructures*, pp. 111–129, 2020.

[6] L. Deng, D. Li, X. Yao, D. Cox, and H. Wang, "Mobile network intrusion detection for iot system based on transfer learning algorithm," *Cluster Computing*, vol. 22, no. 4, pp. 9889–9904, 2019.

[7] J. Dong, G. Wu, T. Yang, and Z. Jiang, "Battlefield situation awareness and networking based on agent distributed computing," *Physical Communication*, vol. 33, pp. 178–186, 2019.

[8] A. M. F. Durrani, A. U. Rehman, A. Farooq, J. A. Meo, and M. T. Sadiq, "An automated waste control management system (AWCMS) by using arduino," in *International Conference on Engineering and Emerging Technologies (ICEET'19)*, pp. 1–6, 2019.

[9] E. Felemban, "Passively inferring wireless network signal quality from different data resources," *IJCSNS International Journal of Computer Science and Network Security*, vol. 20, no. 5, p. 138, 2020.

[10] R. Fotohi, E. Nazemi, and F. S. Aliee, "An agent-based self-protective method to secure communication between UAVs in unmanned aerial vehicle networks," *Vehicular Communications*, p. 100267, 2020.

[11] E. N. Ganesh, "Study of voip network delay using neural networks," *International Journal of Electronics Engineering*, vol. 12, no. 2, pp. 83–91, 2020.

[12] W. Han and C. Lei, "A survey on policy languages in network and security management," *Computer Networks*, vol. 56, no. 1, pp. 477–489, 2012.

[13] B. Harris, "Survival and sustainability for small businesses within the timber industry," *Trident University International, ProQuest Dissertations Publishing*, 2020. (https://search.proquest.com/openview/c8f076325516acb18a76c444a180d9f7/1?pq-origsite=gscholar&cbl=18750&diss=y)

[14] G. Howser, "The network layer," in *Computer Networks and the Internet*, pp. 55–87, 2020.

[15] S. Y. Hui and K. H. Yeung, "Challenges in the migration to 4G mobile systems," *IEEE Communications Magazine*, vol. 41, no. 12, pp. 54–59, 2003.

[16] S. Islam, H. Ali, A. Habib, N. Nobi, M. Alam, and D. Hossain, "Threat minimization by design and deployment of secured networking model," *International Journal of Electronics and Information Engineering*, vol. 8, no. 2, pp. 135–144, 2018.

[17] J. Jabbar *et al.*, "Socialize the behavior of iot on human to devices interaction and internet marketing," *IJCSNS International Journal of Computer Science and Network Security*, vol. 20, no. 5, pp. 158–164, 2020.

[18] M. A. Kochte, R. Baranowski, M. Sauer, B. Becker, and H. J. Wunderlich, "Formal verification of secure reconfigurable scan network infrastructure," in *The 21th IEEE European Test Symposium (ETS'16)*, pp. 1–6, 2016.

[19] M. Kountouris *et al.*, "Performance limits of network densification," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1294–1308, 2017.

[20] D. Kumar, A. Sharma, R. Kumar, and N. Sharma, "A holistic survey on disaster and disruption in optical communication network," *Recent Advances in Electrical & Electronic Engineering (Formerly Recent Patents on Electrical & Electronic Engineering)*, vol. 13, no. 2, pp. 130–135, 2020.

[21] M. Lecuyer, V. Atlidakis, R. Geambasu, D. Hsu, and S. Jana, "Certified robustness to adversarial examples with differential privacy," in *IEEE Symposium on Security and Privacy (SP'19)*, pp. 656–672, 2019.

[22] G. Leu and J. Tang, "Comparison of infrastructure and adhoc modes in survivable networks enabled by evolutionary swarms," in *International Conference on Swarm Intelligence*, pp. 80–89, 2019.

[23] D. Li, R. Zhang, S. Jia, D. Liu, Y. Jin, and J. Li, "Improved dynamic frequency-scaling approach for energy-saving-based radial basis function neural network," *IET Cyber-Physical Systems: Theory & Applications*, vol. 5, no. 2, 2020.

[24] L. Liu, L. Wang, and Z. Cao, "A note on one adaptive indexing structure for realtime search on microblogs," *International Journal of Electronics Engineering*, vol. 12, no. 1, pp. 1–6, 2020.

[25] X. Liu, J. Ning, J. Li, J. Yin, and M. Li, "Model for survivability of wireless sensor network," in *International Conference on Mobile Ad-Hoc and Sensor Networks*, pp. 705–714, 2007.

[26] Y. Liu, J. Bi, and J. Yang, "Research on vehicular ad hoc networks," in *Chinese Control and Decision Conference*, pp. 4430–4435, 2009.

[27] S. H. Mohamed, T. E. H. El-Gorashi, and J. M. H. Elmirghani, "A survey of big data machine learning applications optimization in cloud data centers and networks," *Networking and Internet Architecture*, 2019. (https://arxiv.org/abs/1910.00731)

[28] D. Mox, M. Calvo-Fullana, J. Fink, V. Kumar, and A. Ribeiro, "Mobile wireless network infrastructure on demand," *Robotics*, 2020. (https://arxiv.org/abs/2002.03026)

[29] R. Nabhen, E. Jamhour, and C. Maziero, "RBPIM: A PCIM-based framework for RBAC," in *Proccedings of the 28th Annual IEEE International Conference on Local Computer Networks (LCN '03)*, pp. 52–61, 2003.

[30] S. O. Obute, M. R. Dogar, and J. H. Boyle, "Chemotaxis based virtual fence for swarm robots in unbounded environments," in *Conference on Biomimetic and Biohybrid Systems*, pp. 216–227, 2019.

[31] K. Odagiri, S. Shimizu, and N. Ishii, "Implementation of user authentication processes for the cloud type virtual policy based network management scheme for the specific domain," *IJCSNS International Journal of Computer Science and Network Security*, vol. 20, no. 5, pp. 197–204, 2020.

[32] H. A. Omar, N. Lu, and W. Zhuang, "Wireless access technologies for vehicular network safety applications," *IEEE Network*, vol. 30, no. 4, pp. 22–26, 2016.

[33] S. S. Pundamale, "Survivable networks," *Department of Computer Science, University of Helsinki*, vol. 2, 2007. (`https://www.cs.helsinki.fi/u/niklande/opetus/SemK07/paper/pundamale.pdf`)

[34] M. Sakib and J. Singh, "Simulation based performance analysis of IPSec VPN over IPv6 networks," *International Journal of Electronics Engineering*, vol. 12, no. 2, pp. 92–104, 2020.

[35] L. B. L. Santos, L. R. Londe, T. J. D. Carvalho, D. S. Menasché, and D. A. Vega-Oliveros, "About interfaces between machine learning, complex networks, survivability analysis, and disaster risk reduction," in *Towards Mathematics, Computers and Environment: A Disasters Perspective*, pp. 185–215, 2019.

[36] A. Sharif, J. P. Li, M. A. Saleem, T. Saba, and R. Kumar, "Efficient hybrid clustering scheme for data delivery using internet of things enabled vehicular ad hoc networks in smart city traffic congestion," *Journal of Internet Technology*, vol. 21, no. 1, pp. 149–157, 2020.

[37] J. P. G. Sterbenz, R. Krishnan, R. R. Hain, A. W. Jackson, D. Levin, R. Ramanathan, and J. Zao, "Survivable mobile wireless networks: Issues, challenges, and research directions," in *Proceedings of the 1st ACM Workshop on Wireless Security*, pp. 31–40, 2002.

[38] J. Tang and G. Leu, "Survivable networks via online real-time evolution of dual air-ground swarm," *Swarm and Evolutionary Computation*, vol. 53, pp. 100642, 2020.

[39] V. T. Venkateswarlu, P. V. Naganjaneyulu, and D. N. Rao, "Rendezvous agents-based routing protocol for delay sensitive data transmission over wireless sensor networks with mobile sink," *International Journal of Intelligent Enterprise*, vol. 7, no. 1-3, pp. 338–355, 2020.

[40] Y. Ye, S. Feng, M. Liu, X. Sun, T. Xu, and X. Tong, "A safe proactive routing protocol sdsdv for ad hoc network," *International Journal of Wireless Information Networks*, vol. 25, no. 3, pp. 348–357, 2018.

[41] S. Yousefi, M. S. Mousavi, and M. Fathy, "Vehicular ad hoc networks (vanets): Challenges and perspectives," in *The 6th International Conference on ITS Telecommunications*, pp. 761–766, 2006.

Biography

Awais Akram was born in Pakistan and holds a Bachelor's degree (B.Sc) and Master's degree (M.Sc) in Computer Science from the Islamic University of Bahawalpur, Pakistan. Currently, he pursuing a Ph.D. research at "Yanshan University, China" with projects related to "Network Security and Complex Networks."

Ren Jiadong received the B.S. and M.S. degrees from the Northeast Heavy Machinery Institute, in 1989 and 1994, respectively, and the Ph.D. degree from the Harbin Institute of Technology, in 1999. He is currently a Professor with the School of Information Science and Engineering, Yanshan University, China. His research interests include data mining, complex networks, and software security.

Tahir Rizwan was born in Pakistan in 1987 and earned his B.Sc. degree in "Electrical and Electronic Engineering" from the University of Sunderland, UK in 2012. In 2015, author completed his Master's Degree at "Xidain University, China" in the framework of various projects. The author is currently pursuing a Ph.D. research in "Control Science and Engineering" from "Shanghai Jiao Tong University, China" with the participation of "Artificial Intelligence, Control Theory, Swarm Robots, Machine Learning, Semantic Segmentation and Networking" projects.

Muhammad Irshad He was born in Pakistan and graduated from "University of Punjab, Lahore, Pakistan." Currently, the author is doing a Ph.D. research in "Computer Science and Technology" at "Yanshan University, China" with a project called "Data Mining and Analysis, Networking and Software Security."

Sohail M. Noman received his B. Sc. (Hons) and M. Sc. from University of East London, UK in 2012 and 2014, respectively, and the Ph.D. degree from the Yanshan University, China, in 2020. He is currently pursuing a Post-Doctoral Research fellowship with Shantou University Medical College, China. His research interests include Computer Sciences and Information Technology, with exposure to Healthcare, Data mining, Bioinformatics, and Market Research.

Jehangir Arshad received his Bachelor's degree in Computer Engineering from COMSATS University of Islamabad, Lahore Campus, Pakistan in 2010 and his Master's degree in Electrical and Electronics Engineering from the School of Electronics Engineering, Bradford University, England in 2012. He worked as a lecturer at COM-

SATS Islamabad University, Sahiwal Campus, Pakistan from 2012 to 2017. After completing his Ph.D. studies at the State Key Laboratory of Integrated Service Networks, Xi'an, China University, he joined COMSATS as an Assistant Professor in July 2017. His research interests include the Wireless Communications Network, Network Security and User Authentication, 5th Generation Mobile Communication Systems and alternative energy systems.

Sana Ullah Badar holds a Bachelor's degree in Computer Science from NFC IET Multan , Pakistan in 2013 and a Master's degree in Computer Science from Government College Lahore , Pakistan in 2016. Since 2017, he has worked as a lecturer at COMSATS University of Islamabad, Sahiwal Campus, Pakistan. In parallel, he is pursuing Ph.D. in Computer Sciences. His research interests include Network Security and User Authentication, Data Science, Big Data, and alternative energy systems.

A Detection Method Based on Behavior-path Representation Against Application-layer DDoS Attacks

Yuntao Zhao[1], Wenjie Cui[1], and Yongxin Feng[2]
(Corresponding author: Yongxin Feng)

School of Information Science and Engineering, Shenyang Ligong University[1]
Shenyang, Zip code 110159 - CHINA
Graduate School, Shenyang Ligong University[2]
Shenyang, Zip code 110159 - CHINA
(Email: fengyongxin@263.net)

Abstract

With the huge increase in the number of network attack incidents, today's Cyberspace is facing unprecedented security threats. In all forms of attacks, Application-layer distributed denial of service (AL-DDoS) attacks have become one of the severest threats to the security of the internet. In the paper, we focus on the differences between AL-DDoS attack behavior and normal access behavior, and analyze the internal relationship and attack homology. Based on the analyses, a detection method against AL-DDoS attacks is proposed, which uses a relationship graph to reveal the consistency of group behaviors of AL-DDoS attacks. We called the relationship graph the behavior-path, which associates the attack behavior with URL access method, protocol type, source file name, host address and so on. Furthermore, we build up commutative matrix to construct the behavior-path. At the same time, we refine the attack behavior characteristics and divide the AL-DDoS attacks into three categories, which helps to match the behavior path. Finally, with ensemble learning we implement effectively detection of attack behavior. The experimental results show that the novel detection method has highest accuracy of 96.1% to AL-DDoS attacks.

Keywords: Behavior-path; Cyberspace Security; DDoS; Flash Crowd; Network Attack

1 Introduction

With the era of the Internet of Things [5], the interconnection of all things becomes possible, and today's world is more and more dependent on the availability of Cyberspace. Cyberspace [3], known as the "fifth dimensional space", is just changing the way of traditional essential life, from work to entertainment, education to medicine and so on. However, the inherent vulnerabilities of the web architecture of Cyberspace provide opportunities for various attacks [6]. In all forms of attacks, the application-layer distributed denial of service (AL-DDoS) attacks have become one of the severest threats to Cyberspace security. The AL-DDoS [13] is different from the traditional network-layer DDoS. In network layer, DDoS attackers send lots of redundant packets towards the victim hosts, whose vulnerabilities exist only on the network or transport layer. For instance, SYN Flood attackers keep continually sending useless connection requests with SYN flags to the server, which exhaust resources of the server on a massive number of TCP half-connections [28]. In contrast, AL-DDoS perpetrators attack the victims by masquerading Flash Crowd with numerous benign requests. Flash Crowd refers to the situation in which a very large number of legitimate users simultaneously access a popular website. The stealthiness of AL-DDoS makes the behaviors hardly to be detected, which evade most intrusion prevention systems. At the same time, they can generate enormous amount of traffic toward the victim and result in substantial loss of service and revenue for businesses under the attack [8].

One massive AL-DDoS attack [17] happened at yahoo.com on February 7, 2000. Similar attacks also occurred in other websites [22, 26] such as cnn.com, ebay.com and Amazon.com. At 7:10 on October 21, 2016 in the United States, hackers manipulated millions of webcams and related DVR, VCRs as "zombies", paralyzed the domain name server (DNS) belonging to Dyn company via the "Mirai bots" network for DDoS hijacking, resulting in Twitter, Paypal, Spotifyy, Netflix, Airbn. B, Github, Reddit and New York Times, which cannot be accessed, half of the United States into a broken state [5]. A week later, Singapore Telecom also claims to have suffered a similar DDoS attack [16]. Hence AL-DDoS attacks are the most significant security threat that Internet service providers and users have to face.

The reason that AL-DDoS attacks become enormous challenge is as follows. Firstly, AL-DDoS resembles a legitimate network session, which makes them penetrate through firewall and evade most intrusion prevention systems. Secondly, there are far more protocols on application layer than those on network layer. AL-DDoS perpetrators are able to adopt more abundant tactics and intelligent programs to launch attack. Although CAPTHCHA puzzles [10, 12] are designed for AL-DDoS, such mechanism will negatively affect the Quality of Experience (QoE) to all users. Since users have little patience to do a CAPTCHA test, a site of CAPTCHA may drive away legitimate users.

The study in this paper falls in the behavior detection, as we focus on analysis of the behavior and relationships of accessing URLs, whether the URLs belong to the same host address, whether are with the same resource name, or use the same access method,*etc.* To represent the abundant semantics of relationships, we first introduces representation graph of accessing URL to depict the entity (nodes or URLs) and relationship as the vertex and edge of a graph. Then we use behavior-path to reveal the consistency of group behaviors of AL-DDoS attacks. At the same time, we build up commutative matrix to model the behavior-path. In short, we make the following major contributions in this paper:

1) Novel URL relationship representation. Instead of using network traffic only, we further analyze the relationships among URLs. Through focusing on the consistency of AL-DDoS attacks,*i.e.* to access the same URL or to use the same access method,*etc.* we distinguish the malicious behaviors from the benigns in terms of the internal relations of these ordered behaviors, especially the behavioral differences from Flash Crowd.

2) Novel behavior-path and commutative matrix for URL relationship. In order to represent the inherent relation and coordination among entities, we adopt a graph structure to model the access behavior between node and URL or between URLs, which can be used to abstract a behavior track, named behavior-path, from hosts-URLs to the vertex-edge of a graph, so as to discover the group behavior of joint perpetrators. Further the representation of multiple integrated behavior-paths reconstruct a commutative matrix for URL relationship.

3) Ensemble learning for AL-DDoS attacks. Given a network structure with different kinds of entities and relationships, we can enumerate many link path form one node to another node with different connection. However, not all of the behavior-path are useful and representation for the group feature of AL-DDoS attack. Thus, an optimized way with a supervised learning, which integrate the behavior-path and commutative matrix, is proposed. We adopt an ensemble learning algorithm to classify the behaviors and define weights.

The content of this paper is organized as follows: In Section 2, we discuss previous works on AL-DDoS detection. In Section 3, we give details of our methods, which consist of feature extraction, relationship analysis and algorithms description, ensemble learning. In Section 4, we present and discuss the obtained experimental results. In Section 5, we conclude the paper.

2 Related Work

Today's security situation of Cyberspace is experiencing an accelerating evolution, which is far more dangerous than it was ten or even five years ago [2]. DDoS attacks have become one of the most serious threats to today's network of business, industry and other field. The detection technology is an important method against attacks and intrusions before any damage and loss. Researchers at home and abroad have gotten many significant achievements on network safety and defense, but attack technology is also evolving, such as polymorphism, rookit and obfuscation. Detection and antidetection, attack and counter attack, intrusion and antiintrusion have become a fierce competition from the traditional network-layer DDoS to the current AL-DDoS. In the application layer, Web DDoS perpetrators launch attacks through a flood of legitimate HTTP requests and attempt to let the server down, which do not saturate the bandwidth of the victim server through inbound traffic, but through outbound traffic [28]. According to a survey [19], Web server is the primary target of AL-DDoS attack. So far many studies have focus on Web DDoS attack detection. They can be categorized into two classes [23]:

1) Detection based on characteristic of traffic flow, namely in terms of the statistics characteristics of network traffic [18, 25, 27, 29].

2) Detection based on user behavior, namely the different access behavior between DDoS perpetrators and normal users [1, 9, 14, 20, 21].

For the first category, Wei Zhou and others [29] put forward a method to detect AL-DDoS attack in backbones traffic, which constructs a set of models. Through testing the entropy of attacks and Flash Crowd, these models can be used to recognize the AL-DDoS attack. However, this method needs plenty of calculation and many sensors. Yu*et al.* [24] uses the Sibson distance to measure the similarity between the flows and realize the distinction between the DDoS attack flow and the Flash Crowd flow. Zhang [27] proposed a detect method according to the entropy of request pages, which is simpler, less calculation, and is suitable for realtime detection, but it cannot recognize random request attack. Wang*et al.* [18] proposed a HTTP Flood attack detection based on large deviation statistical model. The method mainly uses the imbalance

of server page popularity, but there are two problems: one is some normal users do not satisfy popular page rules, and the other is that it may lead to misjudgment. This is because the actual webpages are changing dynamically, and these changes inevitably affect prior probability distribution. Yu*et al.* [25] proposed a method that utilized correlation coefficient between suspicious flows to detect AL-DDoS attack. However, detection modules of this method need to deploy in every router, but most sites do not have the ability to deploy these devices.

For the second category, Xie Yi*et al.* [21] utilize a hidden semi-Markov model to present the users behavior. It takes DDoS attack as a kind of abnormal user access and use hidden semi-Markov model to describe users accessing behavior. But there are some problems, such as large calculation, complicated process and updating model. In the paper [20], a detection algorithm based on fuzzy comprehensive evaluation is proposed, which exploits user access behavior as a signal for AL-DDoS attack detection. The paper [9] introduced wavelets to identify anomalies in network traffic. But wavelet analysis is too complex and cannot be used for online processing. Oikonomou*et al.* [11] built a normal behavior model to distinguish between attackers and normal visitors. Chen [7] proposes a detection algorithm of AL-DDoS attack based on information entropy. According to the information entropy of URL access rate, the algorithm can detect DDoS attack, but not distinguish between DDoS attacks and Flash Crowd. Ramamoorthi*et al.* [14] proposed a detection method using improved SVM, which also distinguishes users according to the differences between user behaviors, but there are also some problems. Firstly, normal user behavior profile is hard to obtain. Secondly, its detection model also cannot be updated in realtime. Jung*et al.* [4] deeply analyzed the difference between the AL-DDoS and the Flash Crowd. When Flash Crowd occurs, a large number of address cluster recurs, while a large number of new address cluster will appear with AL-DDoS attacks. The distribution of access addresses of Flash Crowd is uneven, while the distribution of access addresses is more uniform when DDoS attacks.

In this paper, we focus on the behavior process and internal relationship of URL access with machine learning, especially analysis difference factors, such as host address, access method, and resource name. Furthermore, we mine the abundant semantics of relationships by modelling behavior-path to reveal the consistency of group behaviors of AL-DDoS attacks, and then effectively detect AL-DDoS attack.

3 Methodology

3.1 Feature Extraction

Feature extraction is the most influential stage in DDoS attack detection, and it could directly affect the detection accuracy. If a set of robust features can be found to represent behaviors of URL access in this stage, the performance can be significantly improved. In terms of access behaviors, the AL-DDoS attacks are divided into three categories [28], including the fixed URL attack, the random URL attack and traversal URL attack. The URL access mode for DDoS attack is as the following Table 1.

Table 1: URL access mode for DDoS attack

URL access mode	Description
fixed URL	Repeatedly request the same URL page
random URL	Repeatedly request random URL pages (from the same service source)
traversal URL	Repeatedly and Periodically request a set of URL pages (from the same service source)

In order to represent the internal relationships of access behaviors, we extract six types of relationship of URL access, including access relationship between user (IP address) and URL, protocol relationship of the same access protocol type between URLs, host relationship of the same access host address between URLs, name relationship of the same access source file name between URLs, method relationship of the same access method between URLs, intensity relationship of the access intensity from user to URL. The all relationships are as the following Table 2.

Table 2: Internal relations of access behaviors

Relationship	Description
Access (A)	The source IP node access a URL page
Protocol (P)	The access protocol type between two URLs
Host (H)	The access host address between two URLs
Name (N)	The access source file name between two URLs
Method (M)	The access function method between two URLs
Intensity (I)	The URL access intensity

Furthermore, in order to characterize and quantify the internal relationship among the above mentioned Web access behaviors, we establish a matrix-based mathematical representation. The description of each matrix is as the following Table 3.

3.2 Algorithm Description

Given the analysis of rich relationship types of URL for AL-DDoS, it is important to model them in a proper way

Table 3: Description of each matrix

Matrix	Element	Description
I	i_{kj}	If the access intensity that the k_{th} node access the j_{th} URL page exceeded a threshold, i_{kj} =1, else i_{kj} =0.
A	a_{ij}	If the i_th node access the j_{th} URL page, a_{ij} = 1, else a_{ij} =0.
P	p_{ij}	If there is same access protocol type between the i_{th} URL and the j_{th}, p_{ij} =1, else p_{ij} =0.
H	h_{ij}	If there is same access host address between the i_{th} URL and the j_{th}, h_{ij} =1, else h_{ij} =0
N	n_{ij}	If there is same access source file name between the i_{th} URL and the j_{th}, n_{ij} =1, else n_{ij} =0.
M	m_{ij}	There is same access method between the i_{th} URL and the j_{th}. If POST method, m_{ij} =1, else GET method, m_{ij} =0.

so that different relations can be better and easier handled. When we adopt machine learning algorithms, it is also better to distinguish different types of relationships. Thus, in this section, we introduce how to represent the Web access by establishing a relationship graph (including URL and the relationships among them).

The URL and the relationship between different URLs are defined as a graph, such as G = (V, E). V denotes the entity set of URLs and E denotes the relationship type set between URLs. The number of V is expressed as $|V|$, $|V| > 1$,and the same to $|E|$, $|E| > 1$. G is a graph with entities as V and edges as E.

In AL-DDoS detection, there are two types of entities. The first type is IP address of users that access the URL, and the second is URL. In the graph G, the entity is the note of G. There are six types of relations as the edges of E, for example, user (IP address) accessing URL, the same access protocol type between URLs, the same access host address to URL, the same access source file name between URLs, the same access method between URLs, the same access method between URLs, and the access intensity from user to URL. The different source nodes and different relations of URLs drives us to use a machine-readable representation to diversify the characteristics among behaviours of web access. In order to descript the behaviours, we construct a relation track, called "behavior-path", which represent the process from one source node to accessed URLs. When more than one nodes have the same types of relations, there is a group behavior that represent a group characteristic.

Definition 1. *A behavior-path is a track defined on graph of application-layer access G = (V, E), and is denoted in the form as follows:*

$$node_1 \overset{relation_1}{\longrightarrow} URL_1 \overset{relation_2}{\longrightarrow} URL_2 \ldots \overset{relation_i}{\longrightarrow} node_j$$

For example, if there were two nodes which carry out the AL-DDoS attack with the type of traversal URL, the scanning range changed from URL_1 to URL_2 within a period of time. The relationship is the same access host address between URL_1 and URL_2. A typical behavior-path with the group behavior including two nodes is as follows:

$$node_1 \overset{access}{\longrightarrow} URL_1 \overset{samehost}{\longrightarrow} URL_2 \overset{access^{-1}}{\longrightarrow} node_2$$

where 'access' represents the relationship that $node_1$ accessed URL_1. The 'same host' represents that there is the same access host-address between URL_1 and URL_2. The '$access^{-1}$' represents the relationship that URL_2 is accessed by $node_2$.

Definition 2. *Given a graph of network G = (V, E), a commutative matrix M_p for a behavior-path ($V_1 \longrightarrow V_2 \longrightarrow \ldots V_n$) is defined as follows:*

$$M_p = G_{V_1V_2} G_{V_2V_3} G_{V_{n-1}V_n}$$

Where $G_{V_iV_j}$ is the adjacency commuting matrix from entity V_i to entity V_j. $M_p(i, j)$ that is the of element matrix M_p represents the number of track between entity V_i and entity V_j.

The sophisticated behaviors track of AL-DDoS can be defined by M_p based on behavior-path. Here we set the M_p matrixas symmetric, which means that there is a same measure between two entities not only considering the URLs, but also considering the relationship between URLs. So the behavior-path is symmetric. In order to explain the matrix representation, the three categories of AL-DDoS are analyzed.

1) Fixed URL attack of AL-DDoS. A fixed URL attacker concentrates intensive fire on one or a few of URLs. For example, the attacker $node_1$ chooses the URL_1 as the target and $node_2$ has launched a coordinated attack on URL_2. URL_1 and URL_2 have the same relationship such as the same access method and so on. The $node_1$ and $node_1$ may be only puppets controlled by the same botnet, who have the same behavior and attack together. The behavior-path is as follows:

$$node_1 \overset{I}{\longrightarrow} URL_1 \overset{H}{\leftrightarrow} URL_2 \overset{I_T}{\longleftarrow} node_j$$

We used the matrix I to filter the node that access URL with intensity (access intensity or attack intensity) beyond a threshold. Also we adopt the matrix H

to descript the relation between the victimized URLs who come from the same host. Thus, we can represent the coordinated behavior as IHI^T. The matrix is descripted in the Table 3.

Given an example of 5*5 matrix I_1. I_1 is expressed as follows:

$$I = \begin{array}{c} \\ IP_1 \\ IP_2 \\ IP_3 \\ IP_4 \\ IP_5 \end{array} \begin{bmatrix} U_1 & U_2 & U_3 & U_4 & U_5 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The sign U_j stands for an URL and IP_i stands for an IP address. If the node of IP_i access the URL of U_j, the matrix element a_{ij} is 1, and else that is 0. H_1 is also an example of 5*5 matrix that represents the relation between the victimized URLs who come from the same host.

$$H_1 = \begin{array}{c} \\ U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \end{array} \begin{bmatrix} H_1 & H_2 & H_3 & H_4 & H_5 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix} \quad (1)$$

In Equation (1), $a_{ij} = 1$ shows that the URL (U_i) belongs the host (H_j). Otherwise, $a_{1,1}$ shows that the URL (U_i) don't belong the host (H_i).

$$I_1 \cdot H_1 = \begin{array}{c} \\ IP_1 \\ IP_2 \\ IP_3 \\ IP_4 \\ IP_5 \end{array} \begin{bmatrix} H_1 & H_2 & H_3 & H_4 & H_5 \\ 2 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (2)$$

Equation (2) shows that the IP_i access the Host H_i. For example, $a_{1,1} = 2$ stands for the access intensity (2) from IP_1 to H_1.

2) Random URL attack of AL-DDoS

A random URL attack is a low intensity attack, which imitates the randomness of legitimate users. The master of botnet usually uses the different puppets, different protocol (P), different access method (M), different access source file (N) to disguise as Flash Crowd, but for attacking the URLs from the same host address. Thus, the behavior-path can be descripted as follows:

$$node_1 \xrightarrow{A} URL_1 \xrightarrow{H} URL_2 \begin{Bmatrix} P \\ N \\ \cdots \\ M \end{Bmatrix} URL_3 \xleftarrow{H^T} URL_4 \xleftarrow{A^T} node_2$$

We take the matrix A as the representation of accessing URL. In the all relation of URLs, the URLs usually come from the same host. As the same, consistent attack target, the relation H is inevitable. However, the relation P, M, N and so on are chosen randomly. Hence we can descript the coordinated behaviors of random URL attacks of AL-DDoS as $AHPH^T$, $AHNH^TA^T$, $AHMH^TA^T$, even more complex combination $AHNPN^TH^TA^T$, $AHNMPM^TN^TH^TA^T$.

3) Traversal URL attack of AL-DDoS. A traversal URL attack is also a low intensity attack. The behaviour is more like Web Crawler. In terms of a sequence, the attacker periodically access URLs. The behavior-path can be descripted as follows:

$$node_1 \xrightarrow{A} \begin{Bmatrix} URL_1 \\ URL_3 \\ \cdots \\ URL_i \end{Bmatrix} \xrightarrow{} \begin{Bmatrix} H \\ \cdots \\ M \end{Bmatrix} \xrightarrow{} \begin{Bmatrix} URL_2 \\ URL_4 \\ \cdots \\ URL_j \end{Bmatrix} \xrightarrow{A^T} node_2$$

For example, node 1 periodically accesses the target of the URL set and $node_2$ does the one of another URL set. The two sets have a similar relationship, e.g. coming from a same host address, adopting same access method and so on. Therefore, we believe that these two nodes have the same characteristics of group behavior as a traversal URL attack of AL-DDoS. Hence we can descript the coordinated behaviors of the traversal URL attacks of AL-DDoS as AHA^T,AMA^T, APA^T, ANA^T.

3.3 Ensemble Learning

Given the behavior-path P_k, $k = 1, 2, \cdots, K$. We can compute the commuting matrix M_{P_k}, which is taken as an individual learner or weak learner.

$$M = \sum_{k}^{K} \alpha_k M_{P_k}$$

where the weights $\alpha_k > 0$. To learn the weight of each behavior-path, a labelled dataset $D = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$ is assumed, where $x_i = \left\{x_i^{(1)}, x_i^{(2)}, ..., x_i^{(K)}\right\}$. Then we use ensemble learning algorithm for AL-DDoS attacks is as the following Algorithm 1.

The schematic diagram of ensemble learning for AL-DDoS detection is shown in Figure 1.

4 Experimental Result and Analysis

The experimental hardware and software environment are shown in Table 4.

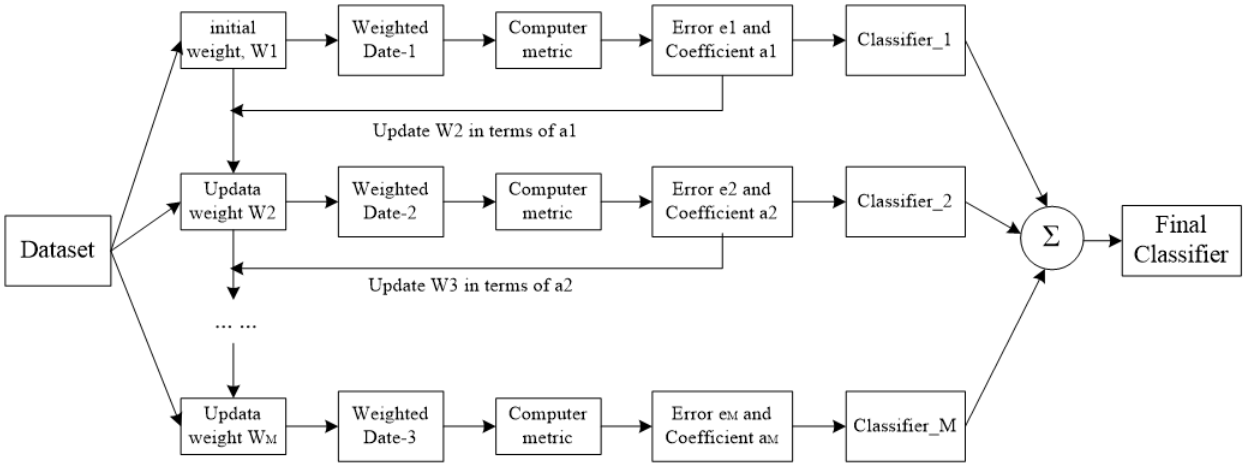


Figure 1: Schematic diagram of ensemble learning for AL-DDoS detection

Algorithm 1 Algorithm procedure

Begin
input: training dataset $D = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$
$D_1 = (\omega_{11}, ..., \omega_{1i}, ..., \omega_{1N}), \omega_{1i} = \frac{1}{N}, i = 1, 2, ..., N$
for m=1,2,...,M,k=1,2,...,K **do**
  if $e_m^{(k)} = \sum_{i=1}^{N} \omega_{m,i} I(M_m^{(k)} \neq y_i) < 0.5$ **then**
    $M_m^{(x)} = M_m^{(x)}$
  else
    $M_m^{(x)} = \overline{M_m^{(x)}}$
  end if
end for
$\omega_{m+1,i} = \frac{\omega_{mi}}{Z_m} \exp\left\{-\frac{y_i}{K} \sum_{k=1}^{K} \alpha_m^{(k)} M_m^{(k)}(x)\right\}$, i=1,2,...N
f(x)=$\sum_{m=1}^{M} \frac{1}{K_m} \sum_{k=1}^{K} \alpha_m^{(k)} M_m^{(k)}(x)$
End

Table 4: Hardware and software environment

Hardware	Software
CPU:Intel(R)Core(TM) i7-8550U@1.80GHz	OS:Windows 10 professional System type: x64
RAM:8G	IDE:PyCharm 2018.3.5 x64
Memory type: DDR4 2400MHz	Language:Python 2.7, Matlab 2016a

Our experimental dataset come from the HTTP dataset CSIC 2010 [15]. The HTTP dataset CSIC 2010 contains thousands of web requests automatically generated. It can be used for the testing of web attack protection systems. It was developed at the "Information Security Institute" of CSIC (Spanish Research National Council). The dataset is generated automatically and contains 36,000 normal requests and more than 25,000 anomalous requests [7]. We evaluate the AL-DDoS detection performance of different methods using the measures shown in Table 5.

Table 5: Performance indices of AL-DDoS attack

Indices	Description
TP	of URL access correctly classified as attack
TN	of URL access correctly classified as normal
FP	of URL access mistakenly classified as attack
FN	of URL access mistakenly classified as normal
Precision	TP/(TP + FP)
Recall	TP/(TP + FN)
Accuracy (ACY)	(TP + TN)/(TP + TN + FP + FN)

Based on the HTTP dataset CSIC 2010, we use MATLAB software to integrate data and simulate the access of the Web server. In the experiment, the scenario is constructed by the interaction process from 3000 IP source addresses to 600 URLs access addresses. There are 300 IP nodes of the fixed URL attack, 600 IP nodes of the random URL attack, 600 IP nodes of the traverse URL attack and 1500 legitimate nodes in simulation experiment, whose proportion of all nodes respectively is 10%,20%,20% and 50%. There are 500 of legitimate nodes (1500) on Flash Crowd, whose proportion of all URLs is 16.7%, Half of the data is used for training, and the other half is for testing. The experimental results are shown in Table 6.

Based on the relationship graph of URL access, we design a series of mixed matrices, such as IHI^T, $AHPH^TA^T$, $AHNH^TA^T$, AHA^T, AMA^T, APA^T, ANA^T and $AHMH^TA^T$. IHI^T, which establishes the

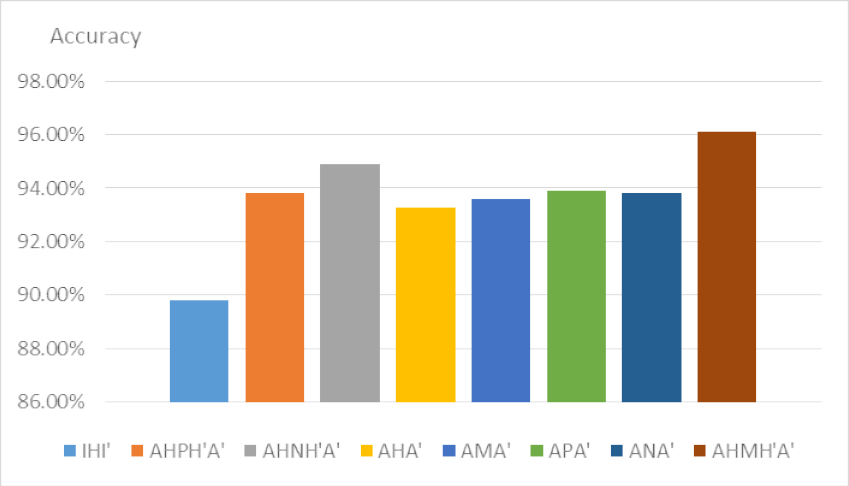


Figure 2: Accuracy of different methods for the different URL modes

Table 6: Detection performance evaluation

Method	URL mode	ACY	TP	TN	FP	FN
IHI^T	fixed	89.8%	123	461	39	27
$AHPH^TA^T$	random	93.8%	281	469	31	19
$AHNH^TA^T$	random	93.3%	274	462	38	26
AHA^T	traversal	93.6%	265	484	16	35
AMA^T	traversal	93.9%	278	483	17	32
APA^T	traversal	93.8%	271	479	21	29
ANA^T	traversal	94.9%	289	470	30	11
$AHMH^TA^T$	random	96.1%	292	477	23	8

relation between access intensity and host address, is used to distinguish the fixed URL attacks. $AHPH^TA^T$, which establishes the relation among URL address, host address, protocol type, is used to distinguish the random URL attacks. $AHNH^TA^T$, which establishes the relation among URL address, host address, source file name, is used to distinguish the random URL attacks. AHA^T, which establishes the relation between URL address and host address, is used to distinguish the traversal URL attacks. AMA^T, which establishes the relation between URL address and access method, is used to distinguish the traversal URL attacks. APA^T, which establishes the relation between URL address and protocol type, is used to distinguish the traversal URL attacks. ANA^T, which establishes the relation between URL address and source file name, is used to distinguish the traversal URL attacks. $AHMH^TA^T$, which establishes the relation among URL address, host address, access method, is used to distinguish the random URL attacks. The experimental results in Table 6 show that the accuracy of AL-DDoS attack detection is from 89.9% to 96.1%. The novel detection method can accurately distinguish there kind of AL-DDoS including the fixed URL attacks, the random URL attacks, the traversal URL attacks.

Figure 2 shows the accuracy of different methods for the different URL modes.

5 Conclusion and Future Work

With the popularity of the network and the explosive growth of network traffic, the burst traffic caused by hotspot events and centralized access often leads into the Web service congestion and even paralysis. Burst traffic is usually called "Flash Crowd". Flash Crowd and DDOS attacks are essentially different. In all Cyberspace attacks, AL-DDoS attacks have become one of the most difficult to defend, and the stealthiness of AL-DDoS attacks make the behaviors hardly to be detected. In this paper, we further analyze the relationships among URLs. By focusing on the consistency of AL-DDoS attacks,*i.e.* to access the same URL or to use the same access method,*etc.*, we model the behavior-path and commutative matrix, which can obtain the differences between the normal behaviors and the attacks. Then we adopt an ensemble learning method to integrate the matrix and construct model. Furthermore, we propose the mixed matrices that mining internal relationship of URL access and extract AL-DDoS attack feature. For example, the mixed matrix of IHI^T can establish the relation between access intensity and host address, and then is used to distinguish the fixed URL attacks. The experimental results show that the novel detection method perform well on the test dataset and achieve the highest accuracy of 96.1%. In the future work, as we have discussed, there is currently a lack of big data analysis and experiments on application layer network attacks, for example, hundreds of thousands of zombie machines. Focusing more closely on the application-layer, in future we plan to further detect attacks on a larger network scale and more nodes. With improving experimental conditions and environment, in subsequent studies we will analyze the complexity of the

defense technique under increasing the number of simulation nodes. As future work, we also plan to extend the detection capabilities of the framework, namely by supporting detection with other machine learning algorithms, which can be used as an improvement of the attack detection.

Acknowledgments

This work was supported by General Project of Liaoning Provincial Department of Education (LG201908), Postdoctoral fund of Shenyang Ligong University, Project of Applied Basic Research of Shenyang(18-013-0-32), 2017 Distinguished Professor Project, Liaoning Nature Science Foundation (20180551066), Program for Liaoning Distinguished Professor.

Data Availability Statement

The dataset [15] is from Microsoft Malware Classification Challenge (BIG 2015). This competition is hosted by WWW 2015 / BIG 2015 and the following Microsoft groups including Microsoft Malware Protection Center, Microsoft Azure Machine Learning and Microsoft Talent Management. The dataset is real and available.

References

[1] L. Ahn, M. Blum, and J. Langford, "Telling humans and computers apart automatically," *Communications of the ACM*, vol. 47, no. 2, pp. 56–60, 2004.

[2] Cisco, *Cisco 2017 Annual Cybersecurity Report: Chief Security Officers Reveal True Cost of Breaches and the Actions Organizations are Taking*, Technical Report, 2017.

[3] S. Hameed and H. Ahmed Khan, "SDN based collaborative scheme for mitigation of ddos attacks," *Future Internet*, vol. 10, no. 3, pp. 23, 2018.

[4] J. Jung, B. Krishnamurthy, and A. M. Rabinovich, "Flash crowds and denial of service attacks: Characterization and implications for cdns and web sites," pp. 293–304, 2002.

[5] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.

[6] P. A. R. Kumar and S. Selvakumar, "Distributed denial of service attack detection using an ensemble of neural classifier," *Computer Communications*, vol. 34, no. 11, pp. 1328–1341, 2011.

[7] S. Lee, G. Kim, and S. Kim, "Sequence-order-independent network profiling for detecting application layer ddos attacks," *Eurasip Journal on Wireless Communications and Networking*, vol. 2011, no. 1, pp. 50–67, 2011.

[8] Q. Liao, H. Li, S. Kang, and C. Liu, "Application layer DDoS attack detection using cluster with label based on sparse vector decomposition and rhythm matching," *Security and Communication Networks*, vol. 8, no. 17, pp. 3111–3120, 2015.

[9] W. Lu and A. A. Ghorbani, "Network anomaly detection based on wavelet analysis," *EURASIP Journal on Advances in Signal Processing*, no. 4, pp. 1–16, 2009.

[10] W. G. Morein, A. Stavrou, D. L. Cook, A. D. Keromytis, V. Misra, and D. Rubensteiny, "Using graphic turing tests to counter automated ddos attacks against web servers," in *Proceedings of the 10th ACM conference on Computer and communications security*, pp. 8–19, 2003.

[11] G. Oikonomou and J. Mirkovic, "Modeling human behavior for defense against flash-crowd attacks," in *IEEE International Conference on Communications*, pp. 1–16, June 2009.

[12] J. F. Podevin, "Telling humans and computers apart automatically," *Communications of the ACM*, vol. 47, no. 2, pp. 57–60, 2004.

[13] A. Praseed and P. S. Thilagam, "Ddos attacks at the application layer: Challenges and research perspectives for safeguarding web applications," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 661–685, 2019.

[14] A. Ramamoorthi, T. Subbulakshmi, and S. M. Shalinie, "Real time detection and classification of ddos attacks using enhanced svm with string kernels," in *International Conference on IEEE Recent Trends in Information Technology (ICRTIT'11)*, pp. 91–96, June 2011.

[15] R. Ronen, M. Radu, C. Feuerstein, E. Y. Tov, M. Ahmadi, "Microsoft malware classification challenge (Big 2015)," *Cryptography and Security*, 2018. (`https://arxiv.org/abs/1802.10135`)

[16] Sohu, *Second America? A Hacker DDoS Attack Paralyzed the Singapore Network (in Chinese)*, Oct. 28, 2016. (`https://www.sohu.com/a/117494668\_223764`)

[17] D. Stevanovic, N. Vlajic, and A. An, "Detection of malicious and non-malicious website visitors using unsupervised neural network learning," *Applied Soft Computing*, vol. 13, no. 1, pp. 698–708, 2013.

[18] J. Wang, X. Yang, and K. Long, "Web ddos detection schemes based on measuring user's access behavior with large deviation," in *IEEE Global Telecommunications Conference (GLOBECOM'11)*, pp. 1–5, Dec. 2011.

[19] Worldwide Infrastructure Security Report, *Arbor Networks*, Technical Report, 2008. (`https://www.fbiic.gov/public/2008/nov/appsfordemocracy.pdf`)

[20] Y. Xie and S. Yu, "A large-scale hidden semi-markov model for anomaly detection on user browsing behaviors," *IEEE/ACM Transactions on Networking*, vol. 17, no. 1, pp. 54–65, 2009.

[21] Y. Xie and S. Yu, "Monitoring the application-layer ddos attacks for popular websites," *IEEE/ACM*

Transactions on Networking, vol. 17, no. 1, pp. 15–25, 2009.

[22] J. Yu, H. Kang, D. H. Park, H. C. Bang, and D. W. Kang, "An in-depth analysis on traffic flooding attacks detection and system using data mining techniques," *Journal of Systems Architecture the Euromicro Journal*, vol. 59, no. 10, pp. 1005–1012, 2013.

[23] S. Yu, *Research on DDoS Attack Detection Technology (in Chinese)*, Ph.D. Thesis, Beijing University of Posts and Telecommunications, 2013.

[24] S. Yu, T. Thapngam, J. Liua, S. Wei, and W. Zhou, "Discriminating ddos flows from flash crowds using information distance," in *The Third International Conference on Network and System Security*, pp. 351–356, Oct. 2009.

[25] S. Yu, W. Zhou, W. Jia, S. Guo, Y. Xiang, and F. Tang, "Discriminating ddos attacks from flash crowds using flow correlation coefficient," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 6, pp. 1073–1080, 2009.

[26] C. Zhang, Z. Cai, W. Chen, X. Luo, and J. Yin, "Flow level detection and filtering of low-rate ddos," *Computer Networks the International Journal of Computer and Telecommunications Networking*, vol. 56, no. 15, pp. 3417–3431, 2012.

[27] X. Zhang, *Application Layer DDoS Attack Detection Research (in Chinese)*, Ph.D. Thesis, Beijing University of Posts and Telecommunications, 2009.

[28] Y. Zhao, W. Zhang, Y. Feng, and B. Yu, "A classification detection algorithm based on joint entropy vector against application-layer ddos attack," *Security and Communication Networks*, vol. 2018, pp. 1–8, 2018.

[29] W. Zhou, W. Jia, S. Wen, Y. Xiang, and W. Zhou, "Detection and defense of application-layer ddos attacks in backbone web traffic," *Future Generation Computer Systems*, vol. 38, pp. 36–46, 2014.

Biography

YunTao Zhao received the Ph.D. degree in control science and engineering from Nanjing University of Science and Technology, Nanjing, China, in 2013. He is a Post-Doctoral Researcher with the pattern recognition and artificial intelligence, Northeastern University, Shenyang, China, from 2015 to today. He is currently Professor with the Communication and Network Institute and also with the School of Information Science and Engineering, Shenyang Ligong University, Shenyang. He has authored over 30 papers published in related international conference proceedings and journals is the holder of 10 patents and software copyrights. His research interests include mainly network communications and security and satellite communications and protocol analysis.

WenJie Cui was born in Zibo, Shandong Province, China in 1996. She received the B.S. degrees in communication engineering from the Binzhou University, in 2018. She is currently pursuing the M.S. degree in communication engineering at Shenyang Ligong University, Shenyang, China. Her research interest includes network communications and security, deep learning, and malicious code classification technology.

YongXin Feng received the M.S. degree in computer science from Northeastern University in 2000 and the Ph.D. degree in computer science and technology from the School of Information Science and Engineering, Northeastern University, in 2003. She is currently a Professor with Shenyang Ligong University. She has authored over 60 papers in related international conferences and journals. She was a recipient of the ICINIS 2011 best paper awards and 15 science and technology awards including the National Science and Technology Progress Award and youth science and technology awards from the China Ordnance Society. Her research interests are in the areas of network management, wireless sensor network, and communication and information systems.

A Novel Certificateless Aggregation Signcryption Scheme Under Cloud Computing

Mingju Zhao and Yuping Peng
(Corresponding author: Yuping Peng)

School of Electrical Engineering, Zhengzhou University of Science and Technology
Zhengzhou 450000, China
(Email:352720214@qq.com)

(Received Mar. 2, 2020; Revised and Accepted Nov. 10, 2020; First Online Feb. 15, 2021)

Abstract

The traditional certificateless aggregation signcryption scheme (CLASC) scheme has low computational efficiency and time-consuming. Therefore, this paper proposes a novel CLASC with non-bilinear pairings under the cloud computing environment. Based on the discrete logarithm problem, it is proved that the new scheme satisfies the confidentiality and unforgerability under the random oracle model. In the verification phase of the aggregation signature, the third party's secret information is not needed, so the new scheme meets the public verifiability. Compared with the state-of-the-art signcryption schemes, it reveals that the new scheme can achieve higher security at a lower computing rate under cloud computing.

Keywords: Certificateless Aggregation Signcryption; Cloud Computing; Non-bilinear Pairings; Random Oracle Model

1 Introduction

At present, more and more countries have invested in the research on the cloud computing and achieved fruitful results. The cloud computing has been widely used in food safety, public safety, health monitoring, intelligent transportation, security, environmental protection and many other industries [6, 8, 9, 15]. The network scale has also been expanding from a laboratory to a building to a community, and different systems have been integrated. With the expansion of network scale, the problems of cloud computing system are exposed. The application industries of the cloud computing, such as food safety and intrusion detection, require the cloud computing to be able to provide fast and accurate response to emergencies, users and managers, so as to achieve accurate communication between people and Things [7, 12, 18]. Meanwhile, it also needs to ensure that the network infrastructure has an economic deployment. This requires the system to operate in an efficient, reliable and secure mode, so cryptography is used to design secure and efficient algorithms and protocols, which is the focus of cloud computing research. The core technology to ensure information security is modern cryptography, which can ensure the confidentiality, integrity, availability and non-repudiation of information in the network environment. Where, confidentiality can be achieved by encryption, and authentication can be achieved by digital signature [21]. If you need to achieve confidentiality and authentication, the traditional public key cryptography is to use "sign first and then encrypt", but this method is inefficient. In 1997, Zheng *et al.* proposed the concept of signcryption and gave a specific scheme [20]. In 2002, Baek *et al.* defined the security model of signcryption scheme for the first time [2]. In practical application, when there are a large number of signers, recipients need to verify multiple ciphertexts at the same time. In order to enhance the efficiency of ciphertext validation, Selvi *et al.* [13] proposed the concept of aggregation signature making full use of the advantages of aggregation signature. In 2003, AlRiyami *et al.* [4] first proposed the certificateless aggregation signcryption (CLASC) system, which not only avoided the problem of public key certificate management and verification, but also solved the key escrow problem. In 2008, Barbosa *et al.* [1] proposed a certificate-free sign-off scheme for the first time and presented its corresponding security model. Subsequently, references [16, 17, 19] proposed the certificateless aggregation signcryption scheme.

2 Preliminaries

The equation of the elliptic curve is defined as $y^2 = x^3 + ax + b$ $(a, b \in F_p)$ on F_p(F_p represents a finite field with p elements, $p > 3$ is prime). The discriminant is $4a^3 + 27b^2 \neq 0 \mod p$. A set of all solutions on the elliptic curve and an infinite point O is represented by $E(F_p)$, that is, $E(F_p) = \{(x, y)|x, y \in F_p\}$, and satisfies the equation $y^2 = x^3 + ax + b \cup O$. The number of points on $E(F_p)$ is represented by q, which becomes the order of the elliptic curve.

- Discrete logarithm problem (DLP). Let G be an additive cyclic group with order q, and P is the generator of G. For $b \in Z_q^*$, finding an integer makes $b = aP$ be difficult.
- Computational Diffie-Hellmanproblem (CDHP). For unknown $a, b \in Z_q^*$, computing abP is difficult.

3 The Proposed Security Model

The security model for certificateless signcryption schemes is introduced by Barbosa and Farshim (2008). In this section, we propose a security model for certificateless aggregate signcryption schemes. The ciphertext indistinguishability and the existential unforgeability security models are used to capture the confidentiality and authenticity requirements, respectively. As for the adversarial model, we follow the common approach in the certificateless setting, which considers two types of adversaries. A Type I adversary A_I who does not have access to the master secret key but can replace the public key of any entity with another value and a Type II Adversary A_{II} who has access to the master secret key but is unable to perform public key replacement. We now define the required security games to capture.

The confidentiality property is defined based on the concept of indistinguishability of encryptions under adaptively chosen ciphertext attacks (IND-CCA2). We define the following two games against Type I and Type II adversaries.

Game I. The game is performed by a challenger C and a Type I adversary A_I.

1) Initialization. C runs the Setup algorithm to generate a master secret key msk and the public system parameters $params$. C keeps msk secret and gives $params$ to A_I. Note that A_I does not know msk.
2) Phase 1. A polynomially bounded number of the following queries is performed by A_I. The queries can be made adaptively so that answers to the previous queries might affect subsequent ones.
   a. $RequestPublicKey$. When A_I supplies an identity ID_u and requests u's public key, C responds with the public key P_u for the identity.
   b. $ExtractPartialPrivateKey$. When A_I supplies an identity ID_u and requests u's partial private key, C responds with the partial private key D_u for the identity.
   c. ReplacePublicKey. When A_I supplies an identity ID_u and a new valid public key value P'_u; C replaces the current public key value with the value P'_u.
   d. $ExtractSecretValue$. When A_I requests the secret value of an identity ID_u, the challenger returns the secret value x_u of u. The public key of u should not have been replaced by A_I.
   e. $Signcrypt$. When A_I submits a sender with an identity ID_S, a receiver with an identity ID_R, a message M and some state information Δ to the challenger, C responds by running the Signcrypt algorithm on the message M, the state information Δ, the sender's private key (D_S, x_S) and the receiver's public key P_R.
   f. $AggregateUnsigncrypt$. When A_I submits an aggregate ciphertext c, some state information Δ, senders with identities $ID_{i\,i=1}^{n}$ and a receiver with the identity ID_R, C checks the validity of c and if it is a valid ciphertext, then C returns the result of running the AggregateUnsigncrypt algorithm on the ciphertext c, the state information Δ, the receiver's private key (D_R, x_R) and the senders' public keys $P_{i\,i=1}^{n}$.
3) Challenge. When Phase 1 ends, the adversary outputs $n+1$ distinct identities $ID_{i\,i=1}^{*n}$, ID_R^*, some state information Δ^* and two sets of n messages $M_0^* = m_{0i\,i=1}^{*\,n}$, $M_1^* = m_{1i\,i=1}^{*\,n}$. Now, a bit μ is randomly chosen by C who then produces c^* as the aggregate signcryption of messages M_μ^* using the state information Δ^*, the private keys corresponding to $ID_{i\,i=1}^{*n}$ and the public key and the identity of u_R^*. The challenger returns c^* to the adversary.
4) Phase 2. The adversary can continue to probe the challenger as in Phase 1.
5) Response. The adversary returns a bit μ'.

We say that the adversary wins the game if $\mu' = \mu$, subject to the following conditions:

1) A_I never queries the partial private key for ID_R^*.
2) A_I cannot make an AggregateUnsigncrypt query on c^* under ID_R^* and $ID'^{n}_{i\,i=1}$ where at least for one i, $ID_i^* = ID'_i$. The only exception is when the public key P_i^* of all of the senders ID_j^* with $ID_j^* = ID'_j$ or that of the receiver P_R^* used to signcrypt M_μ^* have been replaced after the challenge was issued.

The advantage of A_I is defined as follows:

$$Adv_{A_I}^{IND-CLASC-CCA2} = |2Pr[\mu' - \mu] - 1|.$$

where $Pr[\mu' - \mu]$ denotes the probability that $\mu' = \mu$.

Game II. The game is performed by a challenger C and a Type II adversary A_{II}.

1) Initialization. C first generates $(params, msk)$ and outputs them to A_{II}.
2) Phase 1. A_{II} may adaptively make a polynomially bounded number of queries as in Game I. The only constraint is that A_{II} cannot replace any public keys. Note that since A_{II} knows the master secret key, it can compute the partial private key of any identity.

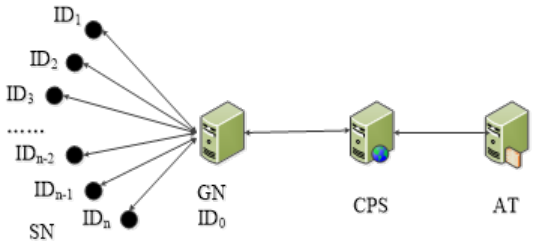


Figure 1: WSN architecture

3) Challenge. When Phase 1 ends, the adversary outputs $n+1$ distinct identities $ID^{*n}_{i\,i=1}$, ID^*_R, some state information Δ^* and two sets of n messages $M^*_0 = m^{*\,n}_{0ii=1}$, $M^*_1 = m^{*\,n}_{1ii=1}$. Now, a bit μ is randomly chosen by C who then produces c^* as the aggregate signcryption of messages M^*_μ using the state information Δ^*, the private keys corresponding to $ID^{*n}_{i\,i=1}$ and the public key and the identity of u^*_R. The challenger returns c^* to the adversary.

4) Phase 2. The adversary can continue to probe the challenger as in Phase 1.

5) Response. The adversary returns a bit μ'.

We say that the adversary wins the game if $\mu = \mu'$,and the following constraints are fulfilled:

1) A_{II} never queries the secret value for the challenge identity ID^*_R.

2) In Phase 2, A_{II} cannot make an AggregateUnsigncrypt query for the challenge ciphertext c^* under ID^*_R, where at least for one i, $ID^*_i = ID'_i$.

As in Game I, the advantage of A_{II} is defined as follows:

$$Adv^{IND-CLASC-CCA2}_{A_{II}} = |2Pr[\mu = \mu'] - 1|.$$

4 Proposed Certificateless Aggregation Signcryption Scheme

This paper proposes a novel Certificateless aggregation signcryption Scheme (CLASC) under cloud computing. A complete cloud computing system is composed of sensory node ($SN_i, 1 \le i \le n$), gateway node (GN), cloud platform server (CPS) and application terminal (AT), as shown in Figure 1.

The function of the SN is to transmit the collected data hop by hop along other sensing nodes and send it to the gateway node. The gateway node automatically saves the data and periodically transfers the automatically collected data to the Internet cloud platform server within a certain time interval. The cloud platform server sends the data to the application terminal for decryption and analysis. The cloud platform server is honest and reliable, responsible for system management and maintenance, including SN, GN and AT registration, private key distribution, etc. The cloud platform server communicates wirelessly with GN, GN and SN, and GN and GN. The implementation process is as follows:

1) System initialization. The algorithm is executed by GN. Enter the security parameter k and select a large prime number $q > 2^k$. Let G be a cyclic group of elliptic curves. And P is the generator of G. GN selects four collash-resistant hash functions: $H_1 : 0,1^* \times G \times G \to Z^*_q$, $H_2 : G \times G \to Z^*_q$, $H_3, H_4 : G \times G \times G \times G \to Z^*_q$. GN randomly selects the master key $s \in Z^*_q$ and preserves it secretly. GN computes the master key $P_{pub} = sP$. CPS publishes system public parameter $params = \{q, P, G, P_{pub}, H_1, H_2, H_3\}$.

2) Key generation. This step is performed by SN_i. SN_i randomly selects secret value $x_i \in Z^*_q$ and saves it, then computes $X_i = x_iP$. The (ID_i, X_i) is sent to CPS. Where x_i is the private key and X_i is the public key.

3) Part private key generation. This step is performed by the CPS. CPS randomly selects the secret value $r_i \in Z^*_q$ and calculates $R_i = r_iP$, $h_{i1} = H_1(ID_i, R_i, X_i)$, $D_i = r_i + sh_{i1}$. And it sends (R_i, D_i) to each sensing node SN_i through the security channel. Where R_i is the user's partial public key and D_i is the user's partial private key. So, the private key of SN_i is $SK_i = (D_i, x_i)$, and the public key is $PK_i = (R_i, X_i)$. Similarly, the private key of the application terminal AT is $SK_B = (D_B, x_B)$, and the public key is $PK_B = (R_B, X_B)$.

4) Individual signcryption. The algorithm is implemented by SN_i. The steps for encrypting the message m_i sent by SN_i to AT are as follows.

   a. SN_i randomly selects $k_i, t_i \in Z^*_q$.
   b. Computing $K_i = k_iP$, $T_i = t_iP$.
   c. Computing $Q_{i1} = k_iX_B$, $Q_{i2} = t_i(R_B + P_{pub}H_1(ID_B, R_B, X_B))$.
   d. Computing $h_{i2} = H_2(Q_{i1}, Q_{i2})$.
   e. Encrypting $C_i = h_{i2} \oplus (m_i||ID_i)$.
   f. Computing $h_{i3} = H_3(C_i, Q_{i1}, Q_{i2}, K_i)$, $h_{i4} = H_4(C_i, Q_{i1}, Q_{i2}, T_i)$.
   g. Signature. $S_i = k_i + t_i + h_{i3}D_i + h_{i4}x_i$.

   The signcryption of the key m_i sent by SN_i to AT is $\sigma_i = (C_i, K_i, T_i, S_i)$.

5) Aggregation signcryption. The algorithm is executed by the gateway node CN. It receives signcryptioner' information $\sigma_i = (C_i, K_i, T_i, S_i)$, the aggregator CN calculates $S = \sum_{i=1}^n S_i$. Then the aggregation signcryption is $\sigma = (\{K_i, T_i, C_i\}^n_{i=1}, S')$, and it is sent to AT.

6) De-signcrypt. This step is performed by the application terminal AT. The steps to decrypt the signcryption $\sigma_i = (C_i, K_i, T_i, S_i)$ sent by AT to SN_i are as follows:

   a. Computing $Q_{i1} = k_i x_B$, $Q_{i2} = T_i(r_B + sH_1(ID_B, R_B, X_B)) = T_i D_B$.

   b. Computing $h_{i2} = H_2(Q_{i1}, Q_{i2})$.

   c. Decrypting $(m_i||ID_i) = h_{i2} \oplus C_i$.

   d. Computing $h_{i3} = H_3(C_i, Q_{i1}, Q_{i2}, K_i)$, $h_{i4} = H_4(C_i, Q_{i1}, Q_{i2}, T_i)$.

   It verifies that whether the signature is correct $S_i P = K_i + T_i + h_{i3}(R_i + P_{pub}H_1(D_i, R_i, X_i)) + h_{i4}X_i$. If it is correct, it proves that the aggregation signcryption is valid, and outputs $(m_i||ID_i)$. Otherwise, output false.

7) Aggregation de-signcrypt. This step is performed by the application side AT. The decryption steps of signcryption $\sigma = (\{K_i, T_i, C_i\}_{i=1}^n, S')$ sent by AT to SN_i are as follows:

   a. Computing $Q_{i1} = k_i x_B$, $Q_{i2} = T_i(r_B + sH_1(ID_B, R_B, X_B)) = T_i D_B$.

   b. Computing $h_{i2} = H_2(Q_{i1}, Q_{i2})$.

   c. Decrypting $(m_i||ID_i) = h_{i2} \oplus C_i$.

   d. Computing $h_{i3} = H_3(C_i, Q_{i1}, Q_{i2}, K_i)$, $h_{i4} = H_4(C_i, Q_{i1}, Q_{i2}, T_i)$.

   It verifies that whether the signature is correct $SP = \sum_{i=1}^n K_i + \sum_{i=1}^n T_i + \sum_{i=1}^n h_{i3}(R_i + P_{pub}H_1(D_i, R_i, X_i)) + \sum_{i=1}^n h_{i4}X_i$. If it is correct, it proves that the aggregation signcryption is valid, and outputs $(m_i||ID_i)$. Otherwise, output false.

5 Analysis of Proposed Scheme

5.1 Correctness of Proposed Scheme

Theorem 1. *The receiver can verify the correctness of the signcryption and aggregation signature, and can get the correct decrypted plaintext m_1.*

Proof.

1) AT can verify the correctness of signcryption $\sigma_i = (C_i, K_i, T_i, S_i)$.

$$\begin{aligned} S_i P &= [k_i + t_i + h_{i3}D_i + h_{i4}x_i]P \\ &= [k_i + t_i + h_{i3}D_i + h_{i4}x_i]P \\ &= K_i + T_i + h_{i3}(R_i + P_{pub}H_1(ID_i, R_i, X_i)) \\ &\quad + h_{i4}X_i \end{aligned}$$

2) AT can verify the correctness of aggregation signcryption $\sigma = (\{K_i, T_i, C_i\}_{i=1}^n, S)$.

$$\begin{aligned} SP &= \sum_{i=1}^n [k_i + t_i + h_{i3}D_i + h_{i4}x_i]P \\ &= \sum_{i=1}^n [K_i + T_i + h_{i3}D_i + h_{i4}(R_i + P_{pub}H_1(ID_i, R_i, X_i))] \\ &= \sum_{i=1}^n K_i + \sum_{i=1}^n T_i + \sum_{i=1}^n [h_{i3}(R_i + P_{pub}H_1(ID_i, R_i, X_i))] \\ &\quad + \sum_{i=1}^n h_{i4}D_i \end{aligned}$$

3) AT can obtain the correct decrypted plaintext m_i.

$$\begin{aligned} h_{i2} &= H_2(Q_{i1}, Q_{i2}) \\ &= H_2(k_i X_B, t_i(X_B + R_B + P_{pub}H_1(ID_B, R_B, X_B))) \\ &= H_2(k_i x_B P, t_i P(x_B + r_B + sH_1(ID_B, R_B, X_B))) \\ &= H_2(k_i X_B, T_i(x_B + r_B + sH_1(ID_B, R_B, X_B))) \\ &= H_2(K_i x_B, T_i D_B) \\ &= h'_{i2} \end{aligned}$$

□

Since SN_i encrypts the plaintext by calculating $C_i = h_{i2} \oplus (m_i||ID_i)$. AT decrypts ciphertext by calculating $m_i||ID_i = h'_{i2} \oplus C_i$, and $h_{i2} = h'_{i2}$, CPS can finally get the correct plaintext.

5.2 Unforgeability of Proposed Scheme

Theorem 2. *In the case of random prediction model and DLP situation, the proposed CLASC scheme in this paper is unforgeability under adaptive selective message attack.*

Lemma 1. *Under the random prediction model, if there is a probability polynomial time attacker A_I wins the game with a non-negligible probability, then there is algorithm C_1 that can solve the DLP (where A_I can execute at most q_{H_i} $(i = 1, 2, 3, 4)$ times of H_i query, q_{SK} times of private key query, q_{PSK} times of partial private key query, q_{PK} times of public key query and q_{SC} times of signcryption query. The user number of aggregation signcryption is n).*

Proof. Supposing algorithm C_1 is a DLP solver with input tuple (P, bP), where $b \in Z_q^*$ is unknown. The goal is to compute b with A_I as the challenger of the subroutine. C_1 maintains the following six lists L_1, L_2, L_3, L_4, L_{ID} and L_{SC} to record query data for predictor H_1, H_2, H_3, H_4, user creation and signcryption, respectively. The list is initialized with empty.

- System initialization stage. C_1 sets $P_{pub} = bP$ (here b is the default system key and secret to A_I, selects and sends the system parameter $params = \{q, P, G, P_{pub}, H_1, H_2, H_3\}$ to the adversary A_I.
- Query phase.

1) H_1 query. C_1 maintains list $L_1 = \{ID_i, R_i, X_i, h_{i1}\}$. When A_I inputs (ID_i, R_i, X_i), C_1 responds to this challenge in the following ways. If the query for this (ID_i, R_i, X_i) already exists in the list L_1, then it returns the corresponding h_{i1} to A_I. Otherwise, C_1 randomly selects $h_{i1} \in Z_q^*$, adds $\{ID_i, R_i, X_i, h_{i1}\}$ to listing L_1 and returns to A_I.
2) H_2 query. C_1 maintains list $L_2 = \{Q_{i1}, Q_{i2}, h_{i2}\}$. When A_I inputs (Q_{i1}, Q_{i2}), C_1 responds to this challenge in the following ways. If the query for (Q_{i1}, Q_{i2}) already exists in the list L_2, then it returns the corresponding h_{i2} to A_I. Otherwise, C_1 randomly selects $h_{i2} \in Z_q^*$, adds $\{Q_{i1}, Q_{i2}, h_{i2}\}$ to listing L_2 and returns h_{i2} to A_I.
3) H_3 query. C_1 maintains list $L_3 = \{C_i, Q_{i1}, Q_{i2}, K_i, h_{i3}\}$. When A_I inputs $\{C_i, Q_{i1}, Q_{i2}, K_i\}$, C_1 responds to this challenge in the following ways. If the query for $\{C_i, Q_{i1}, Q_{i2}, K_i\}$ already exists in the list L_3, then it returns the corresponding h_{i3} to A_I. Otherwise, C_1 randomly selects $h_{i3} \in Z_q^*$, adds $\{C_i, Q_{i1}, Q_{i2}, K_i, h_{i3}\}$ to listing L_3 and returns h_{i3} to A_I.
4) H_4 query. C_1 maintains list $L_4 = \{C_i, Q_{i1}, Q_{i2}, T_i, h_{i4}\}$. When A_I inputs $\{C_i, Q_{i1}, Q_{i2}, T_i\}$, C_1 responds to this challenge in the following ways. If the query for $\{C_i, Q_{i1}, Q_{i2}, T_i\}$ already exists in the list L_4, then it returns the corresponding h_{i4} to A_I. Otherwise, C_1 randomly selects $h_{i4} \in Z_q^*$, adds $\{C_i, Q_{i1}, Q_{i2}, T_i, h_{i4}\}$ to listing L_4 and returns h_{i4} to A_I.
5) User creation query. C_1 maintains initialization list $L_{ID_i} = \{ID_i, h_{i1}, D_i, r_i, R_i, x_i, X_i\}$. It submits user ID_i, if $\{ID_i, h_{i1}, D_i, r_i, R_i, x_i, X_i\}$ already exists in L_{ID_i}, then it will be ignored. Otherwise, C_1 executes the H_1 query and obtains the h_{i1}. If $ID_i = ID_j$, C_1 randomly selects $r_j, x_j \in Z_q^*$, calculates $R_j = r_j P$ and $X_j = x_j P$, inserts $\{ID_j, h_{j1}, \perp, r_j, R_j, x_j, X_j\}$ into L_{ID}. Otherwise, C_1 randomly selects $D_i, x_i \in Z_q^*$, computes $R_i = D_i P - h_{i1} P_{pub}$ and $X_i = x_i P$, inserts $\{ID_j, h_{j1}, \perp, r_j, R_j, x_j, X_j\}$ into L_{ID}.
6) Partial private key query. A_I submits the user ID_i. C_1 makes the following response: if $ID_i = ID_j$, C terminates the game; Otherwise, C_1 returns D_i to A_I.
7) Private key query. A_I submits user identity ID_i. C_1 returns the corresponding x_i to A_I.
8) Public key query. A_I submits ID_i, C_1 returns public key (R_i, X_i) corresponding to ID_i as response.
9) Public key substitution query. A_I adopts a new public key (X_i', R_i') to replace the original public key (X_i, R_i) of the signcryption ID_i.
10) Signcryption query. C_1 maintains initialization list $L_{SC} = \{m_i, ID_i, ID_B, K_i, T_i, h_{i2}, h_{i3}, h_{i4}, S_i, c_i\}$. A_I submits un-signcryption information m_i, sender identity ID_i and receiver identity ID_B. If $ID_i = ID_j$, C_1 randomly selects $S_i, h_{i3}, h_{i4}, k_i \in Z_q^*$, calculates $K_i = k_i P$, $T_i = S_i P - h_{i3} x_j - h_{i4} D_j - K_i$ and $h_{i2} = H_2(K_i x_B, T_i(x_B + r_B + sH_1(ID_B, R_B, X_B)))$. It queries list L_3 and L_4, if L_3 exists in $(C_i, Q_{i1}, Q_{i2}, K_i, h_{i3}')$ or L_4 exists in $(C_i, Q_{i1}, Q_{i2}, K_i, h_{i4}')$, and $h_{i3} \neq h_{i3}' \vee h_{i4} \neq h_{i4}'$, C_1 re-selects $(S_i, h_{i3}, h_{i4}, k_i)$. Otherwise, C_1 computes $C_i = h_{i2} \oplus (m_i || ID_i)$ and returns ciphertext $\sigma_i = (C_i, K_i, T_i, S_i)$. If $ID_i \neq ID_j$, C_1 is calculated according to the signcryption algorithm. H_i query and key query are performed as required, and then the signcryption message $\sigma_i = (C_i, K_i, T_i, S_i)$ is returned. Finally, C_1 inserts $\{m_i, ID_i, ID_B, K_i, T_i, h_{i2}, h_{i3}, h_{i4}, S_i, c_i\}$, $(C_i, Q_{i1}, Q_{i2}, K_i, h_{i3})$ and $(C_i, Q_{i1}, Q_{i2}, T_i, h_{i4})$ into the L_{SC}, L_3 and L_4, respectively.

- Forgery phase. After the query phase, A_I submits the challenge user identity (ID_j, ID_B), the challenge message m_j and its signcryption ciphertext (C_j, K_j, T_j, S_j). C_1 calculates the $h_{i2} = H_2(K_i x_B, T_i(x_B + r_B + sH_1(ID_B, R_B, X_B)))$ to decrypt the message $m_j = h_{i2} \oplus C_j$. According to the forking lemma [5], C_1 uses predictor replay attack technique that can obtain two legal signatures $(m_j, ID_j, ID_B, K_j, T_j, h_{j3}, h_{j4}, S_j)$ and $(m_j, ID_j, ID_B, K_j, T_j, h_{j3}', h_{j4}, S_j')$, where $S_i \neq S_j'$, $h_{j3} \neq h_{j3}'$ and it satisfies:

$$\begin{aligned} S_j &= k_j + t_j + h_{j3} D_j + h_{j4} x_i. \\ S_j' &= k_j + t_j + h_{j3}' D_j + h_{j4} x_i. \end{aligned}$$

Therefore, C_1 calculates:

$$\begin{aligned} S_j' - S_j &= (h_{j3}' - h_{j3}) D_j. \\ b &= \frac{S_j' - S_j - (h_{j3}' - h_{j3}) r_j}{(h_{j3}' - h_{j3}) H_1(ID_j, R_j, X_j)}. \end{aligned}$$

The results are as the response to DLP. Therefore, C_1 successfully obtains an example of DLP problem. The advantage of successfully solving DLP problems is:

$$\varepsilon' = \varepsilon \frac{1}{q_{PSK} + n} (1 - \frac{1}{q_{PSK} + n})^{q_{PSK} + n - 1}.$$

So Theorem 2 and Lemma 1 are correct. □

Lemma 2. *Under the random prediction model, if there is a probability polynomial time A_{II} attacker wins the game with a non-negligible probability, then there is algorithm C_2 that can solve the DLP (where A_{II} can execute at most q_{H_i} $(i = 1, 2, 3, 4)$ times of H_i query, q_{SK} times of private key query, q_{PSK} times of partial private*

key query, q_{PK} times of public key query and q_{SC} times of signcryption query. The user number of aggregation signcryption is n).

Proof. Supposing algorithm C_2 is a DLP solver with input tuple (P, bP), where $b \in Z_q^*$ is unknown. The goal is to compute b with A_I as the challenger of the subroutine. C_1 maintains the following six lists L_1, L_2, L_3, L_4, L_{ID} and L_{SC} to record query data for predictor H_1, H_2, H_3, H_4, user creation and signcryption, respectively. The list is initialized with empty.

- System initialization stage. Supposing $P_{pub} = sP$, $s \in Z_q^*$. The system parameter $params = \{q, P, G, P_{pub}, H_1, H_2, H_3\}$, C_2 sends (q, P, G, P_{pub}, s) to A_{II}.
- Query phase. A_{II} performs the following polynomial bounded query.
  1) H_1, H_2, H_3, H_4 queries are same as Theorem 1.
  2) User creation query. C_2 maintains initialization list $L_{ID_i} = \{ID_i, h_{i1}, D_i, r_i, R_i, x_i, X_i\}$. It submits user ID_i, if $\{ID_i, h_{i1}, D_i, r_i, R_i, x_i, X_i\}$ already exists in L_{ID_i}, then it will be ignored. Otherwise, C_2 executes the H_1 query and obtains the h_{i1}. If $ID_i = ID_j$, let $X_j = bP$, it calculates $R_j = r_jP$ and $D_j = r_j + sH_1(ID_j, R_j, X_j)$, inserts $\{ID_j, h_{j1}, \perp, r_j, R_j, x_j, X_j\}$ into L_{ID}. Otherwise, C_2 randomly selects $D_i, x_i \in Z_q^*$, computes $R_i = r_iP$ and $X_i = x_iP$, inserts $\{ID_j, h_{j1}, \perp, r_j, R_j, x_j, X_j\}$ into L_{ID_i}.
  3) Partial private key query. A_{II} submits the user ID_i. C_2 makes the following response: if $ID_i = ID_j$, C_2 terminates the game; Otherwise, C_2 returns corresponding D_i to A_{II}.
  4) Private key query. A_{II} submits user identity ID_i. C_2 makes the following response: if $ID_i = ID_j$, C_2 terminates the game; Otherwise, C_1 returns x_i to A_I.
  5) Public key query. A_{II} submits ID_i, C_1 returns public key (R_i, X_i) corresponding to ID_i as response.
  6) Public key substitution query. A_{II} submits ID_i and X_i', if $ID_i = ID_j$, C_2 terminates the game; Otherwise, A_{II} adopts X_i' to replace the original public key X_i of the signcryption ID_i.
  7) Signcryption query. C_2 maintains initialization list $L_{SC} = \{m_i, ID_i, ID_B, K_i, T_i, h_{i2}, h_{i3}, h_{i4}, S_i, c_i\}$. A_{II} submits unsigncryption information m_i, sender identity ID_i and receiver identity ID_B. If $ID_i = ID_j$, C_1 randomly selects $S_i, h_{i3}, t_i \in Z_q^*$, calculates $T_i = t_iP$, $K_i = S_iP - h_{i3}(x_i + D_j) - T_i$ and $h_{i2} = H_2(K_ix_B, T_i(x_B + r_B + sH_1(ID_B, R_B, X_B)))$. It queries list L_3 and L_4, if L_3 exists in $(C_i, Q_{i1}, Q_{i2}, K_i, h_{i3}')$ or L_4 exists in $(C_i, Q_{i1}, Q_{i2}, K_i, h_{i4}')$, and $h_{i3} \neq h_{i3}' \vee h_{i4} \neq h_{i4}'$, C_2 re-selects $(S_i, h_{i3}, h_{i4}, t_i)$. Otherwise, C_2 computes $C_i = h_{i2} \oplus (m_i || ID_i)$ and returns ciphertext $\sigma_i = (C_i, K_i, T_i, S_i)$. If $ID_i \neq ID_j$, C_1 is calculated according to the signcryption algorithm. H_i query and key query are performed as required, and then the signcryption message $\sigma_i = (C_i, K_i, T_i, S_i)$ is returned. Finally, C_2 inserts $\{m_i, ID_i, ID_B, K_i, T_i, h_{i2}, h_{i3}, h_{i4}, S_i, c_i\}$, $(C_i, Q_{i1}, Q_{i2}, K_i, h_{i3})$ and $(C_i, Q_{i1}, Q_{i2}, T_i, h_{i4})$ into the L_{SC}, L_3 and L_4, respectively.
- Forgery phase. After the query phase, A_{II} submits the challenge user identity (ID_j, ID_B), the challenge message m_j and its signcryption ciphertext (C_j, K_j, T_j, S_j). C_2 calculates the $h_{i2} = H_2(K_ix_B, T_i(x_B + r_B + sH_1(ID_B, R_B, X_B)))$ to decrypt the message $m_j = h_{i2} \oplus C_j$. According to the forking lemma [5], C_2 uses predictor replay attack technique that can obtain two legal signatures $(m_j, ID_j, ID_B, K_j, T_j, h_{j3}, h_{j4}, S_j)$ and $(m_j, ID_j, ID_B, K_j, T_j, h_{j3}', h_{j4}, S_j')$, where $S_i \neq S_j'$, $h_{j3} \neq h_{j3}'$ and it satisfies:

$$\begin{aligned} S_j &= k_j + t_j + h_{j3}D_j + h_{j4}x_i \\ S_j' &= k_j + t_j + h_{j3}D_j + h_{j4}'x_i \end{aligned}$$

Therefore, C_2 calculates:

$$\begin{aligned} S_j' - S_j &= (h_{j4}' - h_{j4})x_j \\ b &= \frac{S_j' - S_j}{(h_{j4}' - h_{j4})} \end{aligned}$$

The results are as the response to DLP. Therefore, C_2 successfully obtains an example of DLP problem. The advantage of successfully solving DLP problems is:

$$\varepsilon' = \varepsilon \frac{1}{q_{PSK} + n}(1 - \frac{1}{q_{PSK} + n})^{q_{PSK}+n-1}$$

So Lemma 2 is correct. □

5.3 Confidentiality of Proposed Scheme

Theorem 3. *Under the random prediction model, based on CDHP, the proposed CLASC scheme in this paper is indistinct under the adaptive selective ciphertext attack, that is, IND-CLASC-CCA2 is security.*

Lemma 3. *Under the random prediction model, if there is a probability polynomial time adversary A_I (A_{II}) wins the game with non-negligible probability, then there is an instance of CDPH where the challenger can solve with non-negligible probability.*

The proof method of Lemma 3 is similar to the confidentiality proof in document [5]. Due to the limited space, we will not give the process.

Table 1: Comparison of computation and security performance of aggregation signcryption

Scheme	PF-CLRSC	PAS	ESAS	ASS	Proposed
signcryption	np+ne	np+2ns	ne+4ns	3ne+np+ns	(2n+1)s
De-signcrypt	(2n+3)p+(n+1)s	3p+np	(2+n)p+ns	np+ns	(5n+1)s
Total operation	ne+(3n+3)p+(n+1)s	(2n+3)p+2ns	ne+5ns+(n+2)p	2np+3ne+2ns	(7n+2)s
Cost consumption	72.06n+60.85	41.68n+60.03	35.36n+40.02	75.28n	5.81n+1.66
Security	Provable Security	Provable Security	Provable Security	Provable Security	Provable Security
Public verifiability	YES	NO	NO	NO	YES

5.4 Public Verifiability of Proposed Scheme

In this scheme, any third party can verify the following equation when there is a dispute between the signcryption sender and the signcryption receiver about the authenticity of the aggregation signcryption text.

$$\begin{aligned} SP &= \sum_{i=1}^{n} K_i + \sum_{i=1}^{n} T_i + \sum_{i=1}^{n} h_{i3} W + \sum_{i=1}^{n} h_{i4} X_i \\ W &= R_i + P_{pub} H_1(ID_i, R_i, X_i). \end{aligned}$$

Because the verification of this equation does not require the participation of the receiver, and does not require any secret information of the signcryptioner, so the scheme is publicly verifiable.

5.5 Performance Analysis and Discussion

In order to compare the computational efficiency of the proposed scheme, it is assumed that there are n users participating in the scheme. In here, three operations are considered: the exponential operation (e), the multiplication operation on group G(s), and the bilinear pair operation (p). Compared with the three operations, the effect of hashing and XOR operation on the overall efficiency is negligible.

In the proposed scheme, in the signcryption phase, n signcryptioners calculate $Q_{i1} = k_i X_B$, $Q_{i2} = t_i(R_B + P_{pub} H_1(ID_B, R_B, X_B))$ that requires $2n+1$ point multiplication operations. The value of $P_{pub} H_1(ID_B, R_B, X_B)$ is fixed, it only needs to be calculated once. In the de-signcrypt phase, computing SP, $Q_{i1} = K_i x_B$, $Q_{i2} = T_i D_B$ needs $5n+1$ point multiplication operations.

As can be seen from Table 1, when the same number of messages are executed with aggregation signcryption, the operation efficiency of this scheme is greatly improved compared with the schemes in references [3, 10, 11, 14]. Compared with the scheme with relatively high operation efficiency, the operation efficiency is improved by nearly 6 times. From the perspective of the security performance of the scheme, only reference [14] and proposed scheme satisfy the public verifiability. Considering the operation efficiency and security of the scheme, this scheme is better than the above four schemes.

The following is an example of two-column Table 1.

6 Conclusion

Aggregation signcryption has many features such as encryption, signature and batch processing, it is of great application value in the cloud computing environment. In order to improve the computational efficiency of certificateless aggregation signcryption, an non-bilinear pairless aggregation signcryption scheme is proposed based on the random prediction model. Compared with the existing schemes, this scheme has a faster computing speed and is more suitable for application in the Internet of Things. In the future, we will research deep learning methods to improve the certificateless aggregation signcryption.

Acknowledgments

The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] S. S. Al-Riyami, K. G. Paterson, "Certificateless public key cryptography," *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 452-473, 2003.

[2] J. Baek, R. Steinfeld, Y. Zheng, "Formal proofs for the security of signcryption," *Public Key Cryptography, 5th International Workshop on Practice and Theory in Public Key Cryptosystems*, vol. 20, pp. 203-235, 2007.

[3] S. Chandrasekhar, M. Singhal, "Efficient and scalable aggregate signcryption scheme based on multitrapdoor hash functions," in *IEEE Conference on Communications and Network Security (CNS'15)*, 2015. DOI: 10.1109/CNS.2015.7346875.

[4] L. Cheng, Q. Wen, Z. Jin, *et al.*, "Cryptanalysis and improvement of a certificateless aggregate signature scheme," *Information Sciences*, vol. 295, pp. 337-346, 2015.

[5] Z. Eslami, N. Pakniat, "Certificateless aggregate signcryption: Security model and a concrete construction secure in the random oracle model," *Journal of King Saud University-Computer and Information Sciences*, vol. 26, no. 3, pp. 276-286, 2014.

[6] C. Lan, H. Li, S. Yin, *et al.*, "A new security cloud storage data encryption scheme based on identity proxy re-encryption," *International Journal of Network Security*, vol. 19, no. 5, pp. 804-810, 2017.

[7] P. Li, Z. Chen, L. T. Yang, *et al.*, "An incremental deep convolutional computation model for feature learning on industrial big data," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, pp. 1341-1349, 2019.

[8] H. Li, S. L. Yin, C. Zhao and L. Teng, "A proxy re-encryption scheme based on elliptic curve group," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 8, no. 1, pp. 218-227, Jan. 2017.

[9] J. Liu, S. L. Yin, H. Li and L. Teng, "A density-based clustering method for k-anonymity privacy protection," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 8, no. 1, pp. 12-18, Jan. 2017.

[10] J. Liu, C. Zhao, K. Mao, "Efficient certificateless aggregate signcryption scheme based on XOR," *Computer Engineering and Applications*, vol. 52, no. 12, pp. 131-135, 2016.

[11] S. Niu, L. Niu, C. Wang, *et al.*, "A provable aggregate signcryption for heterogeneous systems," *Dianzi Yu Xinxi Xuebao/Journal of Electronics and Information Technology*, vol. 39, no. 5, pp. 1213-1218, 2017.

[12] L. Peng, Z. Chen, L. T. Yang, *et al.*, "Deep convolutional computation model for feature learning on big data in internet of things," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 2, pp. 790-798, 2018.

[13] S. S. D. Selvi, S. S. Vivek, J. Shriram, *et al.*, "Identity based aggregate signcryption schemes," in *Progress in Cryptology*, pp. 378-397, 2009.

[14] G. Sharma, S. Bala, A. K. Verma, "Pairing-free certificateless ring signcryption (PF-CLRSC) scheme for wireless sensor networks," *Wireless Personal Communications*, vol. 84, no. 2, pp. 1469-1485, 2015.

[15] L. Teng, H. Li, "A high-efficiency discrete logarithm-based multi-proxy blind signature scheme," *International Journal of Network Security*, vol. 20, no. 6, pp. 1200-1205, Nov. 1, 2018.

[16] L. Teng, H. Li and S. Yin, "IM-Mobishare: An improved privacy preserving scheme based on asymmetric encryption and bloom filter for users location sharing in social network," *Journal of Computers (Taiwan)*, vol. 30, no. 3, pp. 59-71, 2019.

[17] S. Yin, J. Liu and L. Teng, "Improved elliptic curve cryptography with homomorphic encryption for medical image encryption," *International Journal of Network Security*, vol. 22, no. 3, pp. 421-426, 2020.

[18] Q. Zhang, C. Bai, L. T. Yang, Z. Chen, P. Li, and H. Yu, "A unified smart Chinese medicine framework for healthcare and medical services," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2019. DOI: 10.1109/TCBB.2019.2914447.

[19] J. Zhang, J. Mao, "On the security of a pairing-free certificateless signcryption scheme," *The Computer Journal*, vol. 61, no. 4, pp. 469-471, 2018.

[20] Y. Zheng, "Digital signcryption or how to achieve cost(signature & encryption) << cost(signature) + cost(encryption)," *Lecture Notes in Computer Science*, vol. 1294, pp. 165-179, 1997.

[21] L. Zou, X. Wang, S. Yin, "A data sorting and searching scheme based on distributed asymmetric searchable encryption," *International Journal of Network Security*, vol. 20, no. 3, pp. 502-508, 2018.

Biography

Mingju Zhao is a lecturer in the School of Electrical Engineering & Zhengzhou University of Science and Technology. His research interests focus on computer and network security.

Yuping Peng is a lecturer in the School of Electrical Engineering & Zhengzhou University of Science and Technology. His research interests focus on computer and network security.

Security Analyses of Android APPs on Ad Libs and Linked URLs

Ming-Yang Su, Sheng-Sheng Chen, Tsung-Ren Wu, Hao-Sen Chang, and You-Liang Liu
(Corresponding author: Ming-Yang Su)

Department of Computer Science and Information Engineering, Ming Chuan University
5 De Ming Rd., Gui Shan District, Taoyuan, Taiwan
(Email: minysu@mail.mcu.edu.tw)

(Received Aug. 9, 2020; Revised and Accepted Nov. 10, 2020; First Online Feb. 15, 2021)

Abstract

Security threats posed by free apps with advertising have become a significant concern recently. An app developer must put at least one Software Development Kit (SDK), called ad library (ad lib), into his or her host program and compile the host program and ad-lib(s) into an executable Android Package (APK) file. Therefore, the ad-lib(s) become part of the APK and have all permissions granted to the app. This study proposes a method of evaluating apps' security focusing on two types of threats: (1) permission misuse of ad-libs in an app and (2) the risk of linked URLs when an app is executing. For the first concern, this study observes the SecurityException and checkPermission mechanisms used by ad-libs to attempt permission misuse. For the second concern, this study conducts both static and dynamic analyses to identify all possible linked URLs of an app and evaluates their risks through third-party utilities. The two issues addressed in this paper are beyond the reach of traditional anti-virus software, which normally inspects the codes and is normally unable to determine threats posed by embedded ad-lib(s) and linked URLs, because embedded ad-lib(s) may steal personal information using the host app's permissions, and the danger of linked URLs does not lie in the app itself. The proposed system thus complements traditional anti-virus software to ensure free-app users' security and privacy.

Keywords: Ad Lib; CheckPermission; Permission Misuse; URL (Uniform Resource Locator); SecurityException

1 Introduction

An ad library (ad lib) is a Software Development Kit (SDK) that allows app developers to get advertisement income from their apps. However, ad libs also pose potential security risks that traditional anti-virus software is not designed to detect. Moreover, linked URLs may pose a security threat when an app is executed. Kaspersky Lab [7] detected 905,174 malicious installation packages and recognized 113,640,221 unique URLs as malicious in Q1 2019. In addition, the number of advertising apps (adware) doubled compared to Q4 2018. In recent years, many noticeable security incidents have been related to ad libs and their connected URLs.

A malware named RottenSys, disguised as an app of System Wi-Fi service, appeared in 2018 and was estimated to have infected 5 million brand new smartphones through a supply chain attack [13]. In addition to displaying uninvited ads, CheckPoint researchers found that RottenSys was designed to download and install some components from its C&C server, which means the attackers can easily fully control the infected devices. Another family of malware, called Tekya, was disclosed in 2020 and generated fake clicks on ads and banners delivered by some well-known ad libs, like Google's AdMob [10]. According to the report, more than 56 apps with malware were download and installed on almost 1.7 million devices before CheckPoint found out about them. Kaspersky Lab identified an Android malware called Loapi [11]. Users' mobile phones would be infected if they accidentally clicked on a banner ad while browsing websites, which triggered the download of a counterfeit anti-virus app or adult app. Loapi asks the user for administrator privileges, and if the request is declined, the prompt continues to display on the screen until the user gets fed up and clicks the confirm button. Once the privileges are obtained, due to its modular design, Loapi can download and install new components via remote server commands.

From the above-mentioned incidents, this study therefore addresses the threats of permission misuse of ad libs and linked URLs by an app. In order to gain ad revenues, the developer includes one or more ad libs in the host program and compiles them into an executable APK file, which means that all of the app's declared permissions are now shared by the ad libs. For example, if an ad lib claims to use only permissions p1 and p2 in the document of its official website, and the host app claims to use permissions p3, p4 and p5, then once combined and compiled into an app, the ad lib can use all the permissions from p1

to p5. Permission misuse occurs if the ad lib attempts to use permissions p3, p4, and p5. App developers usually include multiple ad libs in their apps to increase ad revenue, which compounds the security risk. Gao *et al.* [9] designed a system called PmDroid, which uses graphical interfaces to show the severity of permission misuse of ad libs in an app. They found that a large amount of personal information is sent to sites like personal clouds instead of advertising servers. In their experiment, which observed the use of the READ-PHONE-STATE permission, mobile phone status information was sent to advertising servers 89 times, to some cloud servers 29 times, and to unidentified servers 68 times. Wei *et al.* [24] noted that even some apps that are identified by anti-virus software as normal are likely to link to malicious websites during their operations. The authors combined a static method (decompiling and checking program code) with a dynamic one (operating an app for two hours in an emulator and observing its behavior). They collected 13,500 normal apps, and 1,260 known malicious apps and found they linked to 254,022 and 19,510 URLs during operations, respectively. According to the inspection of Web-Of-Trust (WOT) [25] and VirusTotal [23], in the experiment for normal apps, 8.8% of apps linked to malicious sites, 15% linked to bad websites, 73% linked to low-reputation websites, and a total of 74% linked to sites unsuitable for children. Interestingly, although the results of the experiment with known malicious apps were expected to be worse, it was found that the distribution of linked URLs was similar to that of normal apps.

This study therefore addresses two types of security issues about apps: (1) permission misuse of ad libs in an app, and (2) the security threat posed by app-linked URLs. For the first issue, this study adopts SecurityException and/or checkPermission mechanisms to analyze permission misused by ad libs and builds a blacklist as a reference for the proposed system. For the second issue, this study applies static analysis through decompiled APKs and dynamic analysis using an emulator to find all linked URLs. The linked URLs are then assessed by impartial third-party utilities, WOT [25] and VirusTotal [23], in a real-time manner. The two problems addressed in this paper are beyond the reach of traditional anti-virus software, because ad libs use permissions declared by the host app to steal private information, and any linked URLs are not part of the app itself.

The remainder of this paper is organized as follows. Section 2 reviews the related literature. Section 3 describes the research method applied. Section 4 presents the experiment results obtained. Section 5 offers conclusions.

2 Literature Review

According to Ruiz *et al.* [16], the recent notable increase of free apps has caused the top 40 ad networks to respond to less than 18% of ad requests from applications. This has forced free app developers to turn to less well-known ad networks and to include more ad libs in their apps in order to raise their advertising revenue. Stevens *et al.* [20] evaluated 13 ad libs from different ad networks and found several security issues, like using permissions irrelevant to advertising to send SMSs, read Calendars, or make phone calls. In addition, seven of the thirteen ad libs analyzed in this study contained JavaScript interface, meaning that those modules could execute external JavaScript.

Diamantaris *et al.* [8] mentioned that users do not have enough knowledge to distinguish whether a permission request is from the host app or possibly dangerous ad libs. The authors examined over 5,000 popular apps and found 65% of permission requests are not from the host apps, but from ad libs. A system named Reaper was proposed to trace and tell if permission requests are issued by the functionality of host apps or the embedded ad libs. Lee and Ryu [14] suggested that although app developers are informed about permissions required by the ad libs, they may not be aware of the rich functionalities of the permissions declared for the ad libs, which mean they pose serious security threats. Lee and Ryu showed some ad libs can exploit powerful APIs to conduct well-known malicious behaviors. An open-source tool, ADLIB, was presented to detect APIs that are accessible from any advertisements.

Athanasopoulos *et al.* [5] suggested that more than half of the apps on Google Play contain ad libs that link to third-party advertisers, leading to possible privacy leakage. Therefore, the NaClDroid architecture was proposed in order to separate the code of ad libs from that of the host app without sharing permissions. Zhu *et al.* [28] developed a system called AdCapsule, which included two functions: a permission sandbox and a file sandbox. This system aimed to constrain the permission and file use of ad libs to within a permissible range. AdCapsule did not need to change the framework of the Android system or to root the mobile phone, and the cost of running AdCapsule was very low. He *et al.* [12] explored how to select ad networks from the perspective of app developers to maximize their advertising profits and recommended including no more than six ad libs in an app.

Ruiz *et al.* [17] discussed the problems caused by ad-lib updating. According to their experiment, more than 90% of apps are free, and so the only income for these app developers is advertising, thereby making it important to ensure that the ad libs generate expected profits. Any ad lib that fails to obtain expected profits should be replaced or supplemented with other ad libs. They collected 13,983 versions of 5,937 apps and found that nearly 50% of those apps had changed their ad libs within 12 months in the form of addition, removal, or update. Su *et al.* [22] used HTTP data mining through network traffic, based on three aspects: quantitative, timing and semantic. The authors claimed an accuracy rate of 95% for the traffic of malicious ad libs. Shuba and Markopoulou [19] proposed a system called AntWall to prevent ad libs from leaking users' personal information. AntWall can be exe-

cuted in the background without rooting the mobile phone and utilizes low energy consumption and network traffic. However, AntWall relies on the manual input of the name of an ad network to operate effectively. Yan *et al.* [27] designed a new Android model called RTDroid that basically modifies some internal Android components to replace the original Android Dalvik virtual machine (VM) with a real-time VM, ensuring that the execution of any app and its ad libs are more predictable.

Liu *et al.* [15] discussed analytics libraries, which are used to trace ad presence and clicks and are more likely to leak users' personal information than ad libs. The authors implemented a framework called Alde to check if users' in-app actions were collected through analytics libraries. According to their experiments involving 300 apps, some apps did indeed leak users' privacy to analytics libraries without notifying users. Shao *et al.* [18] focused on whether the websites linked in the ad libs threatened the security of mobile phones. They designed a static analysis tool that is able to find the embedded ad libs in an app and a dynamic analysis tool with three functions: identifying linking websites, detecting malware and fraud, and determining the source of an attack. Their experiment analyzed 242 ad libs from 600,000 apps, which were linked to a total of 1.5 million URLs. Some attacks originated through website connection, including malicious anti-virus software fraud, free iPad fraud, and Trojan horse attacks that used messages to distribute advertisements. Su *et al.* [21] noted that most ad libs have communication patterns when communicating with their ad servers. The pattern, although probably not unique, can help identify or group ad libs contained in an app.

3 Research Method

The proposed system herein has two functions: (1) analysis of permission misused by ad libs, and (2) risk assessment of URLs linked with apps. The proposed system is introduced as follows.

3.1 Analysis of Permission Misused by Ad Libs

Android's safety mechanism is to allow the use of its functionalities by declaring corresponding permissions in Android Manifest.xml. If a functionality is used without proper declaration, then errors cause the app's execution to end. Therefore, it is necessary to add an exception mechanism to the code of ad libs that attempt permission misuses, given that ad libs do not know which permissions are declared by the host app. In order to avoid execution errors, there are two ways for ad libs to deal with permission misuse. The first way is to use SecurityException in the java.lang package name to catch exceptions; i.e., the ad lib's code is wrapped with try-catch. When errors occur to an undeclared permission access, a new SecurityException() occurs for exception handling. The second way is to use ContextWrapper.checkPermission() in the android.content package name to confirm whether there is a specific permission declaration.

This study uses the Android emulator to analyze permission misuses of ad libs. This method inserts an ad lib into an empty project, MyAPP, which does not declare any other permission than those required by the ad lib. After execution, the Android SDK is monitored, and the permissions requested by the ad lib from the Android system is the output. If the ad lib only declares permission A and permission B, but permission C is requested, then permission C is determined to be a misuse. The proposed method then adds permission C to the AndroidManifest.XML for declaration and continues to find other misuses until no new ones can be found. The algorithm is given formally as follows.

Algorithm: Determine permission misused by an ad lib.

1) Create an empty project using Android Studio without any declaration of permissions. Insert a test ad lib into the project and add all permissions needed by the ad lib to the AndroidManifest.XML.

2) Set breakpoints for monitoring SecurityException() and ContextWrapper.checkPermission().

3) Execute the project in debug mode and trigger ad events to make sure the code of the ad lib can be executed.

4) Scan the monitored messages and retrieve attempts to access permissions.

5) Compare the permission access attempts obtained from Step 4 with the permissions declared in the AndroidManifest.XML to obtain permission misuses. If any new one exists, then add the declaration of the misused permission to the AndroidManifest.XML and go to Step 3; otherwise, no more permission misuses can be found, and thus terminate the procedure.

The idea behind this algorithm is basically that the misused permissions are obtained through the difference between the attempts to access permissions and the advertiser's declaration. In order to find all permission misuses, the first found one should be added to AndroidManifest.xml, and then Step 3 starts to find the next one until no more can be found. For example, the ad lib may misuse the READ_CONTACTS and SEND_SMS permissions in order. The proposed procedure will first find the attempt to access READ_CONTACTS. After adding the permission to the AndroidManifest.xml, the process returns to Step 3, and it will be able to find the attempt to access the SEND_SMS permission. All permission misuses can be found in this way. Applying the method to investigate different ad libs, a list of ad libs with declared and undeclared permission accesses can be obtained. The list is used in the proposed system to determine if a test APK contained ad libs committing permission misuse.

(a) (b) (c)

Figure 1: Obtaining URLs statically and dynamically. (a) Part of the decompiled program codes, (b) Part of the decompiled program codes when searching URLs from .smali after decompilation, (c) Part of the code to get URLs in real time.

3.2 Risk Assessment of URLs Linked with the App

This function consists of four phases. The first phase reads URLs statically after decompiling the APK. The second phase gathers linked URLs in the emulator. The third phase merges all URLs and sends them to Web-Of-Trust (WOT) [25] and VirusTotal [23] for inspection. The fourth phase receives responses from the two websites and provides security information about the test APK to users.

In the first phase, the user selects an APK and sends it to the proposed server for static analysis. In the server end, the uploaded APK is decompiled and parsed to get all URLs. Figure 1(a) shows the codes to decompile an APK using Apktool [3], in which decoderSetting(true, true) means to decompile source code and resources. Figure 1(b) is the code to parse the .smali files after decompilation in order to get URLs statically. In the second stage, the test APK is run in an emulator in order to record all the URLs with which it is linked. This study applies NOX emulator [6] and Monkeyrunner [1] to simulate user clicks. The URLs are then captured using Pcap4j [26]. Figure 1(c) is the code to get URLs in real time.

The third stage collects all obtained URLs, removes duplicates, and sends them to WOT [25] and VirusTotal [23], two trusted third-party security websites, for risk assessments. The json files sent back from the two websites are then parsed, sorted, and presented to the mobile phone user. This study uses the WOT Public API to send HTTP GET REQUEST, as shown in Figure 2(a), and determines a score for the test APK based on the returned results. Similarly, this study sends HTTP POST REQUEST via the VirusTotal Academic API, as shown in Figure 2(b), and determines a score for the test APK based on the returned results.

4 Experiment Results

The whole process of this study appears in Figure 3. The user selects a test APK on the smartphone and checks either one or both functions described in Section 3. The APK is then sent to the proposed server. After the analyses, the results are returned to the user. In the server, the APK file is decompiled in order to analyze permission misuse and/or linked URL risk. The former searches Package Name to identify the ad libs embedded in the APK and then compares the results with the list of ad lib with permission misuses obtained in Section 3.1. The latter includes a static search for URLs and a dynamic search for URLs from the traffic in the emulator. All collected URLs are sent to WOT [25] and VirusTotal [23] for risk analysis. Finally, the server integrates the results and sends them back to the user, as shown in Figure 3.

4.1 Results of Permission Misuse Analysis

This study tests 26 ad libs on the AppBrain website [4], presenting the results in Table 1. According to the list of ad libs obtained by the algorithm in Section 3.1, the declared permissions by the ad libs are marked as “O”, and the undeclared but requested permissions in run time are marked as “X”. Most of the ad libs have undeclared but requested permissions MODIFY_AUDIO_SETTINGS and BLUETOOTH. Ad libs P and R in Table 1 attempted by requesting ACCESS_FINE_LOCATION/ ACCESS_COARSE_LOCATION and READ_CONTACTS, respectively. In addition, ad libs A and U appear to have requested excessively dangerous permissions, although they did not commit permission misuse. The permissions in red in Table 1 are dangerous, according to the official Android website [2].

Figure 4 shows an example. After choosing an APK and sending it to the server, the results returned to the smartphone are shown in Figure 4(a). Here, the APK contains 9 ad libs, and all of the attempts to access per-

```
String APIRequestUrl = "http://api.mywot.com/0.4/public_link_json2?hosts=";
String APIKey = "&key=635a34a33f3b794681ebb03b2cf28ba1aed525e5";
Set <String> urlHostName;

URL obj = new URL(APIRequestUrl + url + "/" + APIKey);
HttpURLConnection connection = (HttpURLConnection) obj.openConnection();

// optional default is GET
connection.setRequestMethod("GET");

//add request header
connection.setRequestProperty("User-Agent", USER_AGENT);
this.responseCode = connection.getResponseCode();
```

(a)

```
String APIRequestUrl = "https://www.virustotal.com/vtapi/v2/url/report";
String APIKey = "2814e38e8e605d0bf32fdea258626ee2db43e2632cab27071cf7fb860f56585d";
String urlParameters = "apikey=" + APIKey + "&resource=";

URL obj = new URL(APIRequestUrl);
HttpURLConnection httpURLConnection = (HttpURLConnection) obj.openConnection();

httpURLConnection.setRequestMethod("POST");
httpURLConnection.setRequestProperty("User-Agent", USER_AGENT);

// set the status that it can be writen parameters
httpURLConnection.setDoOutput(true);

String parameters = urlParameters + url + "&scan=1";

DataOutputStream dataOutputStream = new DataOutputStream(httpURLConnection.getOutputStream());
dataOutputStream.write(parameters.getBytes());
dataOutputStream.flush();
dataOutputStream.close();
```

(b)

Figure 2: Program segments of communication between the proposed system and WOT and VirusTotal as well. (a) Program segment of WOT API call, (b) Program segment of VirusTotal API call.

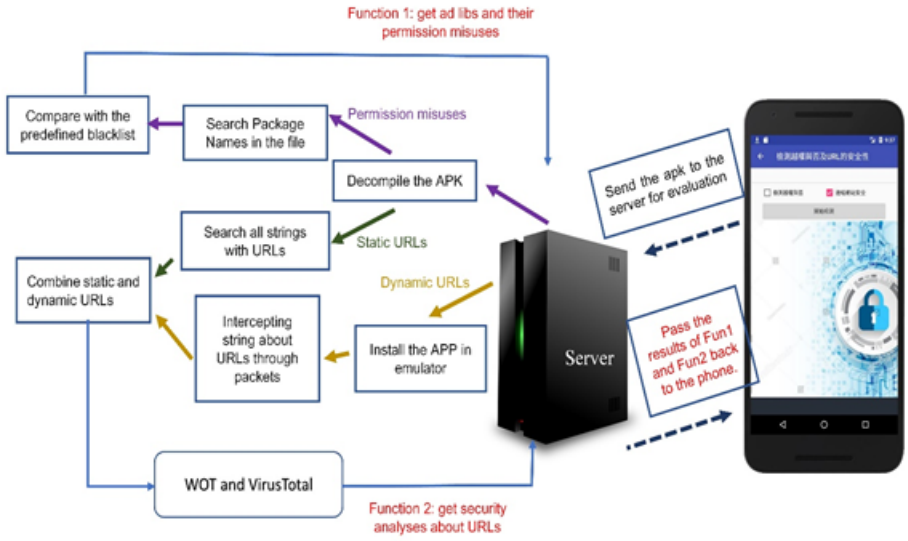
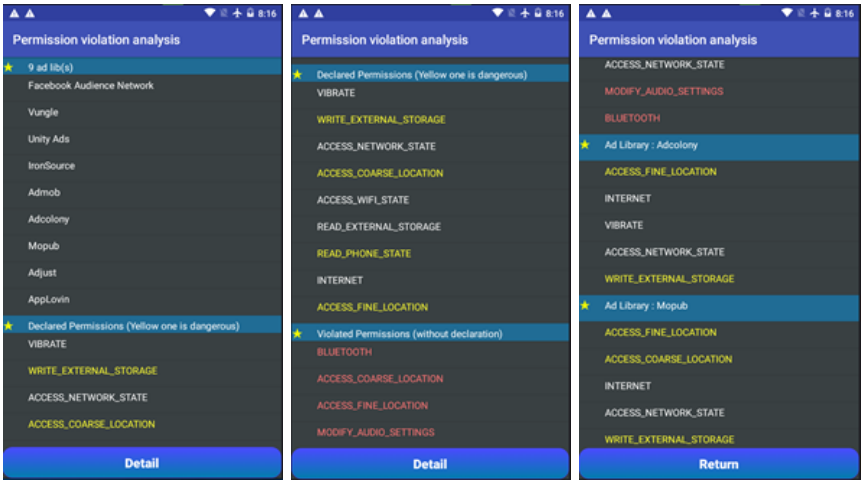


Figure 3: Flowchart of an APK analysis: Permission misuse and linked URL risk



(a) (b) (c)

Figure 4: Example: An APK with nine ad libs and their permission access attempts. (a) All ad libs and declared permissions, (b) Permission misuses in red, (c) Permission access of an ad lib.

Table 1: List of ad libs with permission misuses

Per. / Ad-Lib	INTERNET	ACCESS_NETWORK_STATE	READ_PHONE_STATE	CALL_PHONE	WRITE_EXTERNAL_STORAGE	READ_CALENDAR	WRITE_CALENDAR	READ_CONTACTS	RECEIVE_BOOT_COMPLETED	GET_ACCOUNTS	ACCESS_WIFI_STATE	ACCESS_FINE_LOCATION	ACCESS_COARSE_LOCATION	CHANGE_WIFI_STATE	INSTALL_SHORTCUT	VIBRATE	MODIFY_AUDIO_SETTINGS	BLUETOOTH	READ_EXTERNAL_STORAGE	PACKAGE_USAGE_STATS	BLUETOOTH_ADMIN	GET_TASKS	REAL_GET_TASKS	NFC	RECORD_AUDIO	ACCESS_COARSE_LOCATION	USE_CREDENTIALS	WAKE_LOCK	SYSTEM_ALERT_WINDOW
A	O	O			O		O	O			O	O		O			X	X											
B	O	O															X	X											
C	O	O										O	O																
D	O	O			O							O	O				X	X											
E	O	O			O							O	O				X	X											
F	O	O	O		O						O	O					X	X											
G	O	O	O		O						O																		
H	O	O	O		O														O										
I	O	O															X	X											
J	O	O	O		O						O	O	O				X	X		O	O	O	O						
K	O	O			O							O	O			O													
L	O	O	O								O																		
M	O	O			O												X	X											
N	O	O	O								O																		
O	O	O	O		O														O										
P	O	O			O						O	X	X				X	X											
Q	O	O			O						O		X			O	X	O	O						O	O			
R	O	O	O		O			X				O	O	O			X	X								O			
S	O	O			O							O	O				X	X											
T	O	O	O																										
U	O	O	O		O	O	O			O		O	O				X	X											
V	O	O	O		O					O	O						X	X											
W	O	O			O							O	O				X	X											
X	O	O									O	O	O																
Y	O	O	O		O						O																		O
Z	O	O							O		O	O	O				X	O											
O count	26	26	12	0	18	1	2	1	1	2	12	13	11	2	0	2	0	2	3	1	1	1	1	1	1	1	1	1	1
X count	0	0	0	0	0	0	0	1	0	0	0	1	2	0	0	0	16	14	0	0	0	0	0	0	0	0	0	0	0

mission are shown. Some permission misuses are shown in Figure 4(b) in red. Clicking the Detail button shows all permission access attempts from every ad lib, as seen in Figure 4(c). Permissions in yellow in Figure 4 are considered dangerous according to the official Android website [2].

4.2 Results of Linked URL Risk Assessment

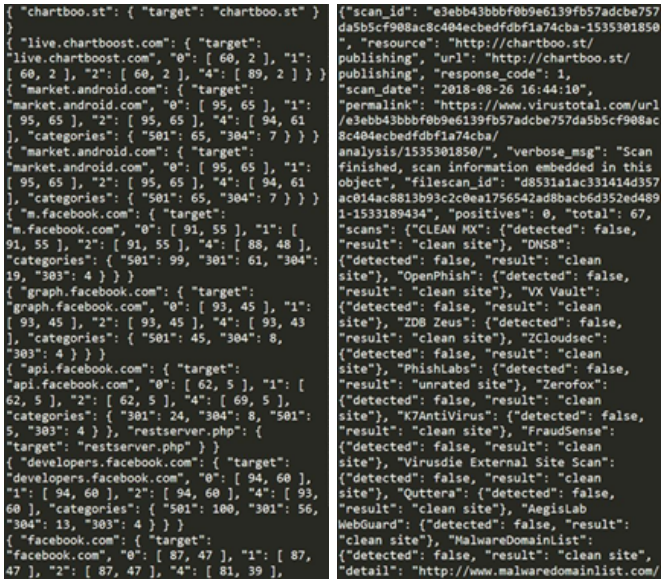
This process includes static and dynamic analyses. The static analysis decompiles the test APK and then parses the smali file of the classes.dex file to obtain the requested URLs by using regular expression. Dynamic analysis analyzes the linked URLs during execution in an emulator by inspecting the network traffic caused by the APK. Finally, all of the URLs obtained from static and dynamic analysis were sent to WOT [25] and VirusTotal [23] for risk analysis. The sites each returned a string in json format, as shown in Figure 5.

The server then computes the scores from the returned json files and presents them to the mobile phone user, as shown in Figures 6(a) and 6(b) for VirusTotal and WOT, respectively. URLs with a score of less than 70 are considered dangerous and are also shown in the lower parts of Figures 6(a) and 6(b). In this example, no URL received a score of less than 70 from WOT (see Figure 6(a)), but several URLs received scores of less than 70 from VirusTotal (see Figure 6(b)). Clicking the icon on the screens from either VirusTotal or WOT, in the middle of the figures, shows detailed information from the third-party risk analyses, as shown in Figure 6(c) and Figure 6(d) for VirusTotal and WOT, respectively.

From the first row of Figures 6(c) and 6(d), it is obvious that VirusTotal focused on checking malware/phishing/malicious/suspicious URLs, while WOT focused on evaluating URLs' trust/child-safety scores. In Figure 6(c), almost all URLs are clean, as each of them gets 1.0 (i.e., a score of 100), with the exception of the URL in the second row (https://api.appsflyer.com/install_data/v3/), which got 0.984375 (i.e., a score of 98.4375). Therefore, the average score is 99, as shown in Figure 6(a), with no URL scoring less than 70. Although the URLs have no malware or are not phishing/malicious/suspicious, it is still necessary to use WOT to evaluate their trust scores (reputation) and child-safety scores (whether the contents are suitable for children). In Figure 6(d), every URL's trust score and child-safety score are given. While the average trust score and child-safety scores are 73 and 70, respectively, as shown in Figure 6(b), those URLs with scores less than 70 (i.e., unsafe ones) can also be found at the lower part of Figure 6(b).

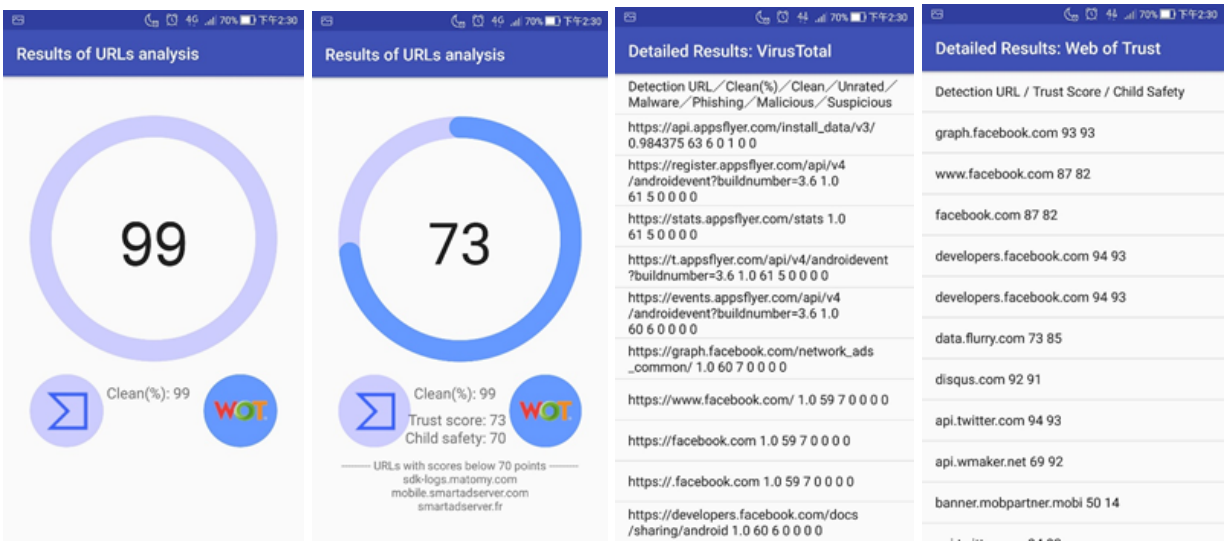
5 Conclusion

This study's findings present from the experiments conducted that most permission misuses involve MOD-



(a) (b)

Figure 5: Json files returned from WOT and VirusTotal. (a) WOT, (b) VirusTotal .



(a) (b) (c) (d)

Figure 6: Risk assessments of linked URLs of an app from VirusTotal and WOT. (a) Summary report from VirusTotal, (b) Summary report from WOT, (c) Detailed information from VirusTotal, (d) Detailed information from WOT.

IFY_AUDIO_SETTINGS and BLUETOOTH permissions, which are fortunately not dangerous according to the official Android website. Some ad libs did not commit permission misuse, but did declare excessive permissions beyond the needs of advertising. In addition, the linked URLs of an app were obtained through static and dynamic analyses. Through the evaluation of these URLs by impartial third-party websites, a quantitative security score is presented to mobile phone users. In other words, the system proposed in this paper informs users clearly whether the ad libs in an app access permissions beyond their declarations, and of the risk of URLs linked in that app. This work helps users achieve further awareness of the security level of an app before installation, in contrast to traditional anti-virus tools.

Acknowledgments

This research was partially supported by the Ministry of Science and Technology under grant numbers MOST 107-2221-E-130-003 and 109-2221-E-130-005.

References

[1] Android, *Monkeyrunner*, Feb. 15, 2021. (`https://developer.android.com/studio/test/monkeyrunner`)

[2] Android, *Manifest.permission*, Feb. 15, 2021. (`https://developer.android.com/reference/android/Manifest.permission`)

[3] Apktool, *A Tool for Reverse Engineering Android Apk Files*, Feb. 15, 2021. (`https://ibotpeaches.github.io/Apktool/`)

[4] AppBrain, *Android Ad Network Statistics And Market Share*, Feb. 15, 2021. (`http://www.appbrain.com/stats/libraries/ad`)

[5] E. Athanasopoulos, V. P. Kemerlis, G. Portokalidis, and A. D. Keromytis,"NaClDroid: Native code isolation for android applications," *Lecture Notes on Computer Sciences*, LNCS 9878, pp. 422-439, Springer, 2016.

[6] Bignox, *NOX Emulator*, Feb. 15, 2021. (`https://tw.bignox.com/`)

[7] V. Chebyshev, F. Sinitsyn, D. Parinov, B. Larin, O. Kupreev, E. Lopatin, *IT Threat Evolution Q1 2019. Statistics*, May 23, 2019. (`https://securelist.com/it-threat-evolution-q1-2019-statistics/90916/`)

[8] M. Diamantaris, E. P. Papadopoulos, and E. P. Markatos, "REAPER: Real-time app analysis for augmenting the android permission system," in *The Ninth ACM Conference on Data and Application Security and Privacy*, pp. 37-48, 2019.

[9] X. Gao, D. Liu, H. Wang, and K. Sun, "PmDroid: Permission supervision for android advertising," in *The IEEE Symposium on Reliable Distributed Systems*, pp. 120-129, 2015.

[10] D. Goodin, *Google Play's Malicious App Problem Infects 1.7 Million More Devices*, Mar. 25, 2020. (`https://arstechnica.com/information-technology/2020/03/found-malicious-google-play-apps-with-1-7-million-downloads-many-by-children/`)

[11] L. Grustniy, *Loapi — this Trojan is hot!*, Dec. 18, 2017. (`https://www.kaspersky.com/blog/loapi-trojan/20510/`)

[12] B. He, H. Xu, L. Jin, G. Guo, Y. Chen, and G. Weng, "An investigation into android in-app ad practice: Implications for app developers," in *The IEEE Conference on Computer Communications (IEEE INFOCOM'18)*, pp. 2465-2473, 2018.

[13] S. Khandelwal, *Pre-Installed Malware Found On 5 Million Popular Android Phones*, The Hacker News, Mar. 15, 2018. (`https://thehackernews.com/2018/03/android-botnet-malware.html`)

[14] S. Lee and S. Ryu, "Adlib: Analyzer for mobile ad platform libraries," in *The 28th ACM SIGSOFT International Symposium on Software Testing and Analysis*, pp. 262-272, 2019.

[15] X. Liu, J. Liu, S. Zhu, W. Wang, and X. Zhang, "Privacy risk analysis and mitigation of analytics libraries in the android ecosystem," *IEEE Transactions on Mobile Computing*, vol. 19, no. 5, pp. 1184-1199, 2020.

[16] I. J. M. Ruiz, M. Nagappan, B. Adams, T. Berger, S. Dienst, and A. E. Hassan, "Impact of ad libraries on ratings of android mobile apps," *IEEE Software*, vol. 31, no. 6, pp. 86-92, 2014.

[17] I. J. M. Ruiz, M. Nagappan, B. Adams, T. Berger, S. Dienst, and A. E. Hassan, "Analyzing ad library updates in android apps," *IEEE Software*, vol. 33, no. 2, pp. 74-80, 2016.

[18] R. Shao, V. Rastogi, Y. Chen, X. Pan, G. Guo, S. Zou, and R. Riley,"Understanding in-app ads and detecting hidden attacks through the mobile app-web interface", IEEE Transactions on Mobile Computing, vol. 17, no. 11, pp. 2675-2688, 2018.

[19] A. Shuba and A. Markopoulou, "Demo: AntWall-A system for mobile adblocking and privacy exposure prevention," in *The ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'18)*, 2018.

[20] R. Stevens, C. Gibler, J. Crussell, J. Ericksonand, and H. Chen, "Investigating user privacy in android ad libraries," in *The IEEE Mobile Security Technologies (MoST)*, 2012.

[21] M. Y. Su, H.-S. Wei, X.-Y. Chen, P.-W. Lin, D.-Y. Qiu, "Using ad-related network behavior to distinguish ad libraries," *Applied Sciences*, vol. 8, no. 10, pp. 1-18, 2018.

[22] X. Su, X. Liu, J. Lin, S. He, Z. Fu, and W. Li, "De-cloaking malicious activities in smartphones using HTTP flow mining," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 6, pp. 3230-3253, 2017.

[23] VirusTotal, *Virustotal*, Feb. 15, 2021. (https://www.virustotal.com/)
[24] X. Wei, I. Neamtiu, and M. Faloutsos, "Whom does your android app talk to?," in *The IEEE Global Communications Conference (GLOBECOM'15)*, pp. 1-6, 2015.
[25] WOT, *Protect Your Browsing with Web of Trust*, Feb. 15, 2021. (http://www.mywot.com/)
[26] K. Yamada, *Pcap4j*, Feb. 15, 2021. (https://github.com/kaitoy/pcap4j)
[27] Y. Yan, S. Cosgrove, V. Anand, A. Kulkarni, S. H. Konduri, S. Y. Ko, and L. Ziarek, "RTDroid: A design for real-time android," *IEEE Transactions on Mobile Computing*, vol. 15, no. 10, pp. 2564-2584, 2016.
[28] X. Zhu, J. Li, Y. Zhou, and J. Ma, "AdCapsule: Practical confinement of advertisements in android applications," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 3, pp. 479-492, 2020.

Biography

Ming-Yang Su received his B.S. degree in Computer Science and Information Engineering from Tunghai University, Taiwan in 1989, and received his M.S. and Ph.D. degrees from National Central University and National Taiwan University in 1991 and 1997, respectively. He is a professor of the Department of Computer Science and Information Engineering at Ming Chuan University, Taiwan. His research interests include network security, intrusion detection/prevention, malware detection, mobile security and wireless sensor networks.

Sheng-Sheng Chen received his B.S. degree in Computer Science and Information Engineering from Ming Chuan University, Taiwan in 2019. His research interests include mobile security and MANET.

Tsung-Ren Wu received his B.S. degree in Computer Science and Information Engineering from Ming Chuan University, Taiwan in 2019. His research interests include mobile security and MANET.

Hao-Sen Chang received his B.S. degree in Computer Science and Information Engineering from Ming Chuan University, Taiwan in 2019. His research interests include mobile security and MANET.

You-Liang Liu received his B.S. degree in Computer Science and Information Engineering from Ming Chuan University, Taiwan in 2019. His research interests include mobile security and MANET.

A Novel Chaotic Image Encryption Algorithm Based on Bit-level Permutation and Extended Zigzag Transform

Chunming Xu

(Corresponding author: Chunming Xu)

School of Mathematics and Statistical, Yancheng Teachers University, China
No. 50, Kaifang Avenue, Yancheng 224002, P. R. China
(Email: ycxcm@126.com)

(Received Oct. 30, 2019; Revised and Accepted Mar. 7, 2020; First Online Feb. 16, 2021)

Abstract

A novel color image encryption using bit-level permutation and extended zigzag transform is presented in this paper. Firstly, the comprehensive zigzag transform is utilized to scramble the bit matrix generated by the original plaintext image. Both the pixel value and the position of the pixel are changed simultaneously. Secondly, the chaotic sequences that have been pre-processed and can reduce the chaotic properties' degradation are used for encrypting the color image, and the encryption security is enhanced. Besides, the chaotic system's initial values are associated with the plaintext image to resist plaintext attack effectively. Experimentation is done on the classical Lena image, and the experiment results demonstrate the superior security and effectiveness of the proposed image encryption method.

Keywords:Bit-Level Permutation; Chaotic System; Color Image Encryption; Extended Zigzag Transform

1 Introduction

Image is an important source of information for human beings to understand the world. Digital image is the most widely used form for image storage, thus it is widely used in economy, military, industry and so on. In some special areas, such as military, legal and medical, digital image has high confidentiality requirements [3,4,6,9,24]. Therefore, image encryption has high research significance. Especially in recent years, with the rapid development of computer hardware conditions, how to achieve efficient and fast image encryption methods has become a new research hot spot.

As a pseudo-random sequence generation method, chaotic sequence has been widely used in image encryption due to its determinacy, sensitivity to initial values and long-term unpredictability. Fridrich brought forward the chaotic image encryption scheme in 1998. Since then, there were abundant study on it all over the world. For example, Mao introduced a novel fast image encryption scheme based on 3D chaotic baker maps [11]. A new hyperchaotic map named stochastic 2D-SHAM was used by Hayder Natiq to enhance the security of encrypted image [12]. Ghebleh used piecewise nonlinear chaotic sequence and least squares approximation to encrypt image [5]. By combining Henon map and Sine map, a novel 2D chaotic map called 2D-HSM was proposed and applied in digital image encryption by Wu [17]. Sodeif Ahadpour gave a chaos-based image encryption scheme based on chaotic coupled map lattices [1]. A novel color image encryption scheme using fractional-order hyperchaotic system was designed by Li [8]. Yin proposed an effective image encryption algorithm which has good sensitivity to initial value and anti-attack ability based on modified elliptic curve cryptography combining with homomorphic encryption for medical image encryption [22].

Many image encryption algorithms are based on pixel level. While in recent years, bit-based image encryption method has attracted the attention of researchers because that it can change both the pixel value and the position of the pixel simultaneously. Till now, a variate of bit-based image encryption algorithms have been proposed [16, 20, 23, 25, 26] by the researchers. Besides, some existing image encryptions can't resist the plaintext attack. Many researchers seeks to generate the secret keys based on the plainimage to resist plaintext attacks [7, 10].

Based on above discussions, a novel color image encryption method is proposed in this paper utilizing extended zigzag transform, bit-level permutation and chaotic system. The main advantages of the proposed algorithm are:

1) The Chaotic sequences adopted for encryption in this paper are plaintext-related and are resistant to chosen-plaintext attacks;

2) The R, G and B components of the color image are well confused by the bit-level permutation;

3) The chaotic sequences used for image encryption have been pre-processed, which will reduce the degradation of chaotic sequences and enhance the encryption quality.

The rest of the paper is organized as follows. In Section 2, we give a brief review of some fundamental knowledge. Section 3 introduces the proposed image encryption scheme. Section 4 presents the experimental results and the security of the algorithm. Finally, we conclude this paper in Section 5.

2 Fundamental Knowledge

2.1 Extended Zigzag Transform

Zigzag transform is a scanning method for matrix. In image encryption, zigzag can be used to change the position of the pixel value to achieve the purpose of confusing the pixels [19]. However, zigzag transform is generally only applicable to square matrices, and it can't be applied to matrices with unequal numbers of rows and columns. To solve this problem, an extended zigzag scan transform algorithm is proposed in [21]. The scanning method adopted by the extended zigzag transform is as Figure 1.

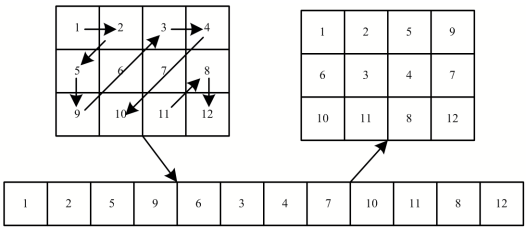


Figure 1: Extended zigzag transform

2.2 The Chaotic System

In 2019, the authors investigated the chaotic behaviors in a 3D autonomous analytic chaotic system with two quadratic terms and a nonlinear sine term, which is expressed by [14]:

$$\begin{cases} \dot{x_1} = x_2 x_1 + a\sin(x_3) + d \\ \dot{x_2} = x_1^2 - f x_2 \\ \dot{x_3} = -c x_1 \end{cases} \quad (1)$$

where x_1, x_2, x_3 are three state variables, and a, c, d, f are control parameters of the Chaotic System (1). When the control parameters are $a = -1.5, c = 7, d = 2, f = 1$, System (1) exhibits very complicated dynamics phenomenon. The three-dimensional view of the chaotic strange attractor and some dynamical behavior in different planes for System (1) are shown in Figure 2.

It can be seen from Figure 2 that System (1) has strong chaotic behavior. Because chaotic system has many strong points, such as random, unpredictability and initial state sensitivity, so that it is suitable for image encryption. Thus, System (1) is used to generate three random sequences for image encryption based on the fourth order Runge-Kutta method in this paper.

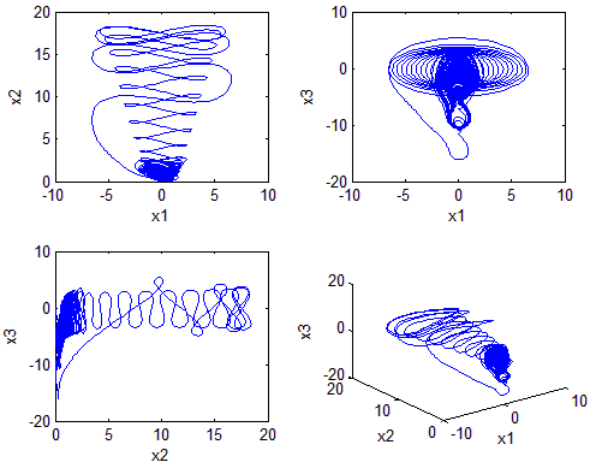


Figure 2: Typical dynamical behaviors of the 3D autonomous analytic chaotic system

3 The Encryption Method

The specific steps of the encryption algorithm can be described as follows:

Step 1: Suppose the size of the color plaintext image P_0 is $H \times W \times 3$, where H and W represent the height and width of the image respectively. Denote the color components of red, green and blue of P_0 as P_R, P_G and P_B, respectively.

Step 2: Integrate the three matrixes P_R, P_G and P_B together to form a $3H \times W$ gray image P_1.

Step 3: Transform each pixel value of P_1 into an 8-bit binary value, then we can get a binary image P_2 with size $3H \times 8W$.

Step 4: Use extended zigzag transformation to scramble the binary matrix P_3 then we can get a new binary matrix P_4.

Step 5: Convert the permutated bit matrix P_4 back to 2D pixel matrix and further convert it to a color image matrix denoted as P_5 using the inverse transformation of Step 3 and Step 2.

Step 6: Choose the system control parameters a, c, d, f of Chaotic System (1).

Step 7: Calculate the initial values x_0, y_0, z_0 of Chaotic System (1) by the following equations:

$$\begin{cases} x_0 = \frac{\sum_{ij} P_{Rij}}{H \times W} + 0.01 \\ y_0 = \frac{\sum_{ij} P_{Gij}}{H \times W} + 0.01 \\ z_0 = \frac{\sum_{ij} P_{Bij}}{H \times W} + 0.01 \end{cases}$$

Step 8: Iterate the Chaotic System (1) for $K + 1000$ times with the initial values x_0, y_0, z_0, remove the former 1000 values to avoid the transition effect of chaotic mapping and three chaotic sequences X_s, Y_s, Z_s of length K can be gotten, where $K = H \times W$.

Step 9: Calculate three encryption sequences S_R, S_G, S_B with X_s, Y_s, Z_s by

$$\begin{cases} S_R = [\frac{|X_s+Y_s-Z_s|}{3}] \times 10^{10} \mod 256 \\ S_G = [\frac{|Y_s+Z_s-X_s|}{3}] \times 10^{10} \mod 256 \\ S_B = [\frac{|Z_s+X_s-Y_s|}{3}] \times 10^{10} \mod 256 \end{cases}$$

Step 10: Denote the color components of red, green and blue of P_5 as $P_R^{'}$, $P_G^{'}$ and $P_B^{'}$, respectively. Encrypt the three components $P_R^{'}, P_G^{'}, P_B^{'}$ for each pixel to obtain their corresponding cipher values C_R, C_G, C_B as

$$\begin{cases} C_R = P_R^{'} \oplus S_R \\ C_G = P_G^{'} \oplus S_G \\ C_B = P_B^{'} \oplus S_B \end{cases}$$

Note that S_R, S_G, S_B are used as key streams instead of X_s, Y_s, Z_s in this step which can reduce the chaos degradation of chaotic properties caused by precision effects [13] and increase the randomness of the key streams.

Step 11: Combine these three image matrices C_R, C_G, C_B to get the ciphered color image C.

The decryption process is the inverse process of encryption and is omitted here for the sake of simplicity.

4 Test and Analysis of the Proposed Scheme

We use MATLAB 2017 as an experimental platform for experiments. Three classic color images i.e. Lena, Mandrill and peppers ($216 \times 216 \times 3$) are used for testing. The system parameters of the system are $a = -1.5, c = 7, d = 2, f = 1$. The plaintext images and their corresponding encrypted image and recovered images are shown in Figure 3.

4.1 Key Space Analysis

Key space refers to the range of the size of the encryption key. According to Kerckhoffs criterion, a good encryption algorithm should have sufficient key space to resist violent attacks. In the proposed algorithm, the secret keys are $K = x_0, y_0, z_0, a, c, d, f$. If the precision of the system parameters and initial values reaches 10^{15}, the total number of different keys in K is 10^{105}. Therefore the key space is large enough and has better security.

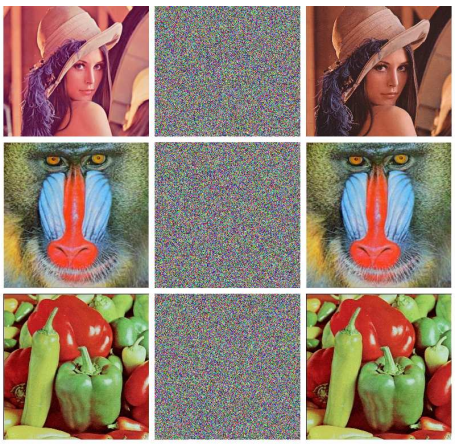


Figure 3: (a) Left column: Plaintext images; (b) Middle column: Ciphertext images; (c) Right column: Recovered images

4.2 Histogram Analysis

The histogram of digital image reflects the distribution of image pixel value. Attackers can use histogram to attack encrypted images. Figure 4 gives the histogram of R, G, B components of three plaintext images and their respective ciphertext images. As can be seen from Figure 4, the distributions of the pixel values of the plaintext images are very uneven, while the distributions of the pixel values of ciphertext images are very uniform, thus the distribution characteristics of the pixel values has been well concealed. As a result, the ciphertext images are resistant to statistical analysis.

4.3 Correlation Analysis

In digital image, there is a high correlation between each pixel and its adjacent pixels. After using an ideal image encryption algorithm to encrypt the image, there should be no correlation between the adjacent pixels of the cipher image. Therefore, the correlation coefficient of adjacent pixels can be used as an important index to evaluate the quality of an image encryption system. The formula for calculating the correlation of adjacent pixels is as follows [15]:

$$r_{xy} = \frac{\sum_{i=1}^{N}((x_i - E(x))(y_i - E(y)))}{\sqrt{(\sum_{i=1}^{N}(x_i - E(x))^2)(\sum_{i=1}^{N}(y_i - E(y))^2)}}$$

$$E(x) = \sum_{i=1}^{N} x_i$$

$$E(y) = \sum_{i=1}^{N} y_i$$

where x_i and y_i are gray-level values of the selected adjacent pixels, and N is the number of sample pixels.

Table 1: The results of correlation analysis

		Plaintext Image			Ciphertex Image		
		R	G	B	R	G	B
Lena	H	0.9375	0.9188	0.8705	-0.0160	0.0349	0.0334
	V	0.9486	0.9360	0.8961	-0.0173	0.0088	0.0212
	D	0.9164	0.9002	0.8589	-0.0182	0.0187	-0.0259
Mandrill	H	0.9487	0.9211	0.9533	0.0128	0.0022	0.0087
	V	0.9435	0.9129	0.9515	-0.0040	0.0251	0.0073
	D	0.9055	0.8545	0.9152	0.0002	-0.0122	0.0237
Peppers	H	0.9246	0.9631	0.9248	-0.0076	-0.0210	0.0083
	V	0.9279	0.9672	0.9314	0.0125	0.0115	-0.0151
	D	0.8751	0.9376	0.8780	0.0149	-0.0063	0.0144

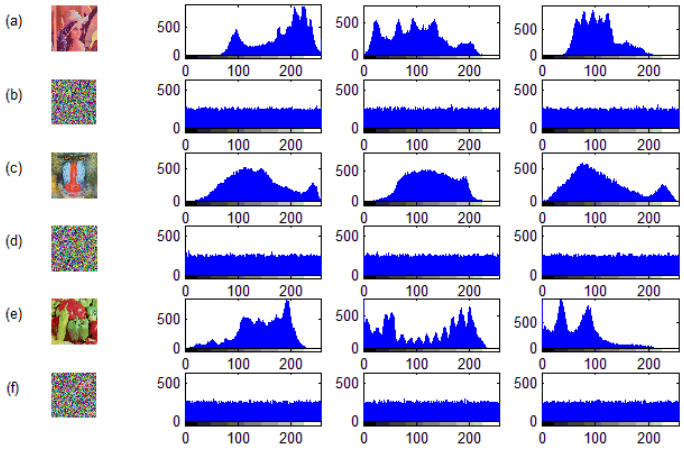


Figure 4: The histogram of the plaintext images and ciphertext images: (a) Plaintext image Lena; (b) Ciphertext image Lena; (c) Plaintext image Mandrill; (d) Ciphertext image Mandrill; (e) Plaintext image Peppers; (f) Ciphertext image Peppers

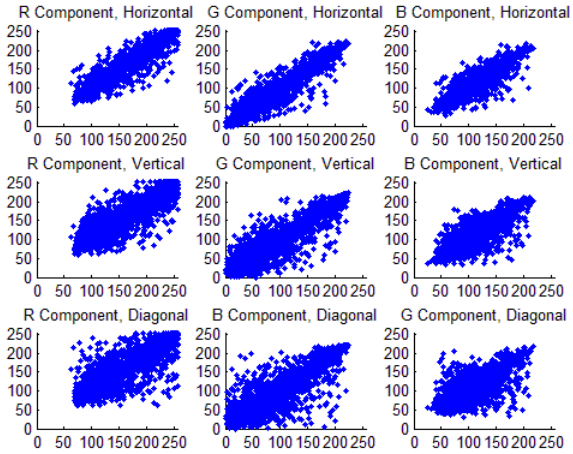


Figure 5: Correlation distributions of plaintext image of Lena in each direction

In order to evaluate the correlation of adjacent pixels in encrypted image, 5000 pixels and its adjacent pixels in horizontal, vertical and diagonal directions are randomly selected in plaintext image and corresponding ciphertext image respectively, and the correlation coefficients in horizontal, vertical and diagonal directions of each image are calculated. Table 1 gives the correlation coefficients of plaintext images and encrypted images in three directions. In addition, the distributions of r_{xy} of Lena are also plotted in Figures 5 and 6.

From Table 1 and Figures 5 and 6, we could find that the correlations of adjacent pixels in the plaintext images in three directions are very strong, while the correlation coefficients of the cipher images tend to zero, which indicates that the proposed algorithm can break the correlation between adjacent pixels.

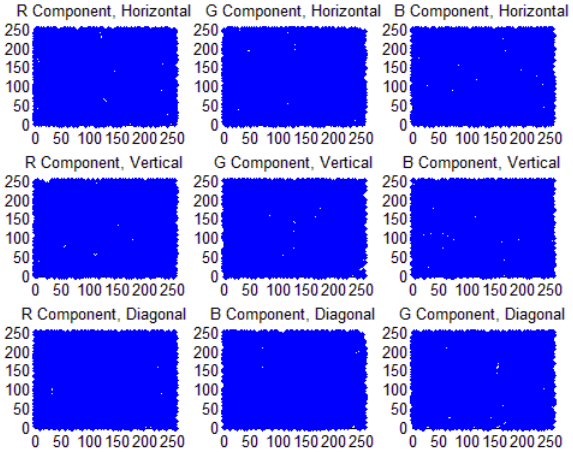


Figure 6: Correlation distributions of ciphertext image of Lena in each direction

4.4 Information Entropy Analysis

The information entropy can be used to measure the randomness and unpredictability of of an image [2]. In general, the more uniform distribution of gray-scale image, the greater the entropy is. For the gray image, the formula for calculating the information entropy is:

$$H(m) = -\sum_{i=0}^{255} P(m_i) \log_2 P(m_i)$$

where m_i is the i th gray level for the digital image and $P(m_i)$ represents the probability of m_i.

For a color image, we calculate the information entropy of R, G and B components respectively. Table 2 lists the results of three ciphertext images. The information entropy of ciphertext images are all very close to the ideal value 8. The results show that the gray value distribution of the encrypted image is very uniform, which also show that the proposed algorithm has a good ability to resist entropy attack.

Table 2: The results of entropy analysis

Images	Entropy		
	R	G	B
Lena	7.9974	7.9965	7.9968
Mandrill	7.9970	7.9972	7.9973
Peppers	7.9972	7.9972	7.9974

4.5 Analysis of Differential Attack Resistance

Differential attack attacks the cryptographic algorithm by comparing and analyzing the changes of image before and after encryption. If an encryption algorithm relies on the plaintext information, it will have better ability to resist differential attack. There are two mainly used differential attack metrics: The number of pixels change rate (NPCR) and the unified averaged changed intensity (UACI) [18]. Suppose there are two plaintext images and there is only one-pixel difference between them. The NPCR and UACI values can be calculated by

$$NPCR = \frac{\sum_{ij} D_{ij}}{W \times H} \times 100\%$$

$$UACI = \frac{1}{W \times H} \frac{\sum_{ij} (C_1(i,j) - C_2(i,j))}{255} \times 100\%$$

where $C_1(i,j)$ and $C_2(i,j)$ are the encrypted images for the plaintext images and D_{ij} is defined by

$$D_{ij} = \begin{cases} 0 & \text{if } C_1(i,j) = C_2(i,j) \\ 1 & \text{if } C_1(i,j) \neq C_2(i,j) \end{cases}$$

The ideal values of NPCR and UACI are 1 and 0.334. The results of NPCR and UACI tests for different color images in three channels are shown in Table 3 and Table 4 respectively. These results show that our encryption algorithm has good performance in resisting differential attack.

Table 3: The results of NPCR test for different color images in three channels

Images	NPCR		
	R	G	B
Lena	99.61%	99.63%	99.60%
Mandrill	99.61%	99.59%	99.60%
Peppers	99.62%	99.60%	99.62%

Table 4: The results of UACI test for different color images in three channels

Images	UACI		
	R	G	B
Lena	33.38%	33.48%	33.47%
Mandrill	33.50%	33.56%	33.43%
Peppers	33.53%	33.46%	33.48%

5 Conclusions

In this paper, an encryption algorithm based on bit-level permutation and extended zigzag transform is proposed. The proposed method can make the information of R, G and B components be fully fused and solve the problem of degradation of chaotic sequences used for image encryption. In addition, the presented method has high sensitivity to plaintext image, which makes the encrypted image be more secure. Experiments is tested on the classical Lena image and the results of encryption are evaluated using the histogram, correlation analysis, entropy, Number of Pixel Change Rate (NPCR) and Unified Average Change Intensity (UACI). The experimental results show the security and effectiveness of the presented algorithm.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (No. 11871417). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] S. Ahadpour, Y. Sadra, "A chaos-based image encryption scheme using chaotic coupled map lattices," *International Journal of Computer Applications*, vol. 49, no. 2, pp. 15-18, 2012.

[2] R. E. Boriga, A. C. Dascalescu, and A. V. Diaconu, "A new fast image encryption scheme based on 2D chaotic maps," *IAENG International Journal of Computer Science*, vol. 41, no. 4, pp. 249-258, 2014.
[3] T. Y. Chen, M. S. Hwang, J. K. Jan, "A feature-oriented copyright owner proving technique for still images," *International Journal of Software Engineering and Knowledge Engineering*, vol. 12, no. 3, pp. 317-330, 2002.
[4] T. Y. Chen, M. S. Hwang, J. K. Jan, "A secure image authentication scheme for tamper detection and recovery," *Imaging Science Journal*, vol. 60, no. 4, pp. 219-233, 2012.
[5] M. Ghebleh, A. Kanso, D. Stevanovi, "A novel image encryption algorithm based on piecewise linear chaotic maps and least squares approximation," *Multimedia Tools and Applications*, vol. 77, no. 6, pp. 7305-7326, 2018.
[6] H. M. Ghadirli, A. Nodehi, R. Enayatifar, "An overview of encryption algorithms in color images," *Imaging Science Journal*, vol. 164, no. 11, pp. 163-185, 2019.
[7] Y. Guo, S. Jing, Y. Zhou, X. Xu, L. Wei, "An image encryption algorithm based on logistic-fibonacci cascade chaos and 3D bit scrambling," *IEEE Access*, vol. 8, pp. 9896-9912, 2020.
[8] P. Li, J. Xu, J. Mou, F. F. Yang, "Fractional-order 4D hyperchaotic memristive system and application in color image encryption," *EURASIP Journal on Image and Video Processing*, vol. 26, no. 10, pp. 11-23, 2017.
[9] H. Liu, C. Jin, "A color image encryption scheme based on arnold scrambling and quantum chaotic," *International Journal of Network Security*, vol. 19, no. 3, pp. 347-357, 2017.
[10] Y. Liu, Z. Qin, J. H. Wu, "Cryptanalysis and enhancement of an image encryption scheme based on bit-plane extraction and multiple chaotic maps," *IEEE Access*, vol. 7, pp. 74070-74080, 2019.
[11] Y. B. Mao, G. R. Chen, S. G. Lian, "A novel fast image encryption scheme based on 3D chaotic Baker maps," *International Journal of Bifurcation & Chaos*, vol. 14, no. 10, pp. 3613-3624, 2004.
[12] H. Natiq, N. M. G. Al-Saidi, M. R. M. Said, A. Kilicman, "A new hyperchaotic map and its application for image encryption," *The European Physical Journal Plus*, vol. 133, no. 6, pp. 5-18, 2018.
[13] E. G. Nepomuceno, L. G. Nardo, J. Arias-Garcia, D. N. Butusov, A. Tutueva, "Image encryption based on the pseudo-orbits from 1D chaotic map," *Chaos*, vol. 29, no. 061101, 2019.
[14] X. Wang, G. R. Chen, "Constructing a new 3D chaotic system with any number of equilibria," *International Journal of Bifurcation and Chaos*, vol. 29, no. 5, pp. 11-23, 2019.
[15] X. Wu, H. Kan, and J. Kurths, "A new color image encryption scheme based on DNA sequences and multiple improved 1D chaotic maps," *Applied Soft Computing*, vol. 37, no. 12, pp. 24-39, 2015.
[16] J. H. Wu, X. F. Liao, B. Yang, "Cryptanalysis and enhancements of image encryption based on three-dimensional bit matrix permutation," *Signal Processing*, vol. 142, pp. 292-300, 2018.
[17] J. H. Wu, X. F. Liao, B. Yang, "Image encryption using 2D Hnon-Sine map and DNA approach," *Signal Processing*, vol. 153, no. 12, pp. 11-23, 2018.
[18] Y. Wu, J. P. Noonan, and S. Agaian, "NPCR and UACI randomness tests for image encryption," *Journal of Selected Areas in Telecommunications*, vol. 1, no. 2, pp. 31-38, 2011.
[19] X. Xu, J. Feng, "Research and implementation of image encryption algorithm based on zigzag transformation and inner product polarization vector," in *IEEE International Conference on Granular Computing*, pp. 556-561, 2010.
[20] L. Xu, Z. Li, J. Li, "A novel bit-level image encryption algorithm based on chaotic maps," *Optics and Lasers in Engineering*, vol. 78, pp. 17-25, 2016.
[21] Y. Q. Yang, T. F. Jiang, G. Liu, "Method of digital image scrambling based on extended zig-zag transformation," *Netinfo Security*, vol. 11, no. 10, pp. 57-58, 2011.
[22] S. L. Yin, J. Liu, L. Teng, "Improved elliptic curve cryptography with homomorphic encryption for medical image encryption," *International Journal of Network Security*, vol. 22, no. 1, pp. 155-160, 2020.
[23] W. Zhang, Y. Hai, Y. L. Zhao, Z. L. Zhu, "Image encryption based on three-dimensional bit matrix permutation," *Signal Process*, vol. 118, pp. 36-50, 2016.
[24] L. H. Zhang, X. F. Liao, X. B. Wang, "An image encryption approach based on chaotic maps," *Chaos Solitons and Fractals*, vol. 24, no. 3, pp. 759-765, 2005.
[25] Y. C. Zhou, W. J. Cao, L. P. Chen, "Image encryption using binary bitplane," *Singal Process*, vol. 100, pp. 197-201, 2014.
[26] Z. L. Zhu, W. Zhang, K. W. Wong, Y. Hai, "A chaos-based symmetric image encryption scheme using a bit-level permutation," *Information Sciences*, vol. 181, no. 6, pp. 1171-1186, 2011.

Biography

Chunming Xu is an associate professor at the mathematics and statistical from Yancheng Teachers University, PR China. His main research interests include image processing and artificial intelligence.

An Identity Authentication Scheme of Energy Internet Based on Blockchain

Xiuxia Tian, Xi Chen, and Siqian Li
(Corresponding author: Xiuxia Tian)

College of Computer Science and Technology, Shanghai University of Electric Power
No.2588 Changyang Road, Shanghai 200090, China
(Email: xxtian@shiep.edu.cn)

(Received July 22, 2019; Revised and Accepted Dec. 15, 2019; First Online Feb. 3, 2020)

Abstract

Energy Internet is becoming the development direction of energy system. In the future, Energy Internet will access a large number of intelligent terminals. Therefore, accessing to Energy Internet services through effective identity authentication is attracting the attention of many users. However, centralized identity authentication will bring great pressure to Energy Internet. In addition, some authentication schemes have serious security problems, such as privacy leakage. To solve these problems, we propose an Energy Internet identity authentication scheme based on blockchain and cryptographic accumulator. In the proposed scheme, we first design the system model of Energy Internet, and then a blockchain is formed by the cryptographic accumulator. Energy Routers and terminals can rely on blockchain to complete authentication. Our security analysis demonstrates that privacy and authentication are both achieved in the scheme. Experimental results illustrate that this scheme effectively improves authentication efficiency.

Keywords: Blockchain; Cryptographic Accumulator; Energy Internet; Energy Router; Identity Authentication

1 Introduction

The traditional power grid is an aggregation network. It is centered on large power plants and it combines power transmission and distribution to provide electricity to users. As the connection point of the user and the power grid, Smart Meter can complete the user's identity authentication and record the power consumption data. In general, Smart Meter only authenticates and communicates with Electric Power Company, and there is almost no communication need between different Smart Meters. Therefore, the authentication scheme for the traditional power grid is a centralized authentication scheme represented by PKI.

However, the centralized authentication scheme is difficult to apply to the future Energy Internet [9]. Compared with the traditional power grid, Energy Internet is a distributed network [18]. In Energy Internet, users are not only energy consumers, but also producers, they have a comprehensive energy utilization system. By connecting to form a regional microgrid, it can realize not only the power transmission and distribution but also energy exchange with the centralized power grid. In addition, users are free to trade energy, energy inflows and outflows will be very frequent. There have a large number of authentication and communication needs in Energy Internet. At the same time, the existing authentication schemes exist many security problems [17, 19]. It is difficult to protect data privacy, confidentiality and authenticity. To use the services in Energy Internet more securely and efficiently, this paper proposes an identity authentication scheme based on blockchain and cryptographic accumulator.

Blockchain is a technology developed independently of Bitcoin [16]. It has many features such as decentralization, distribution, and high security. There is a strong consistency between blockchain and Energy Internet. A cryptographic accumulator is an one-way membership function. It can answer the question about whether someone is a member of a collection without disclosing individual member in the collection [2, 4, 6, 12, 14]. The main contributions of this paper are as follows:

1) First, we built a system model of Energy Internet. Based on this model, an authentication scheme is proposed. To our best knowledge, we are the first to study Energy Internet with its identity authentication scheme, which will bring reference significance to the construction and deployment of Energy Internet.

2) Second, we realize the authentication under the premise of protecting users' privacy, where we utilize the cryptographic accumulator to achieve the construction of blockchain.

3) Third, we provide a comprehensive security analysis to show that the proposed scheme achieves the de-

sired security property. In addition, we conducted experiments on the proposed scheme. The experimental results show that our scheme can achieve efficient authentication requirements.

The remainder of this article is organized as follows: Related work is reviewed in Section 2. In Section 3, we introduce system model. Definition and preliminaries is included in Section 4. Section 5 presents our proposed scheme. We give performance and security analysis in Section 6. Finally, we conclude this paper.

2 Related Work

This section discusses existing authentication schemes.

Literature [8] proposed an access authentication scheme for Energy Internet. In this paper, the authentication request of terminals is handled by an authentication group which consist of a certain number of slave nodes selected by the master node. The authentication group uses PBFT consensus mechanism which is implemented with the Shamir threshold secret sharing mechanism to achieve a consensus with each other.

Literature [13] proposed a distributed authentication mechanism called Bubbles of Trust which is based on blockchain. By setting up a secure virtual area, devices can be securely authenticated to each other while maintaining data integrity and security. However, this scheme is less scalable and less flexible.

Literature [1] proposed a distributed public key infrastructure system based on Ethereum, controlled by the smart contract. It is meant to solve the problem that the centralized and opaque public key infrastructure is easy to be attacked by rogue certificates. The system is highly transparent and provides fine-grained attribute management of the web of trust.

Literature [10] proposed a distributed authentication scheme called Certcoin which is based on NameCoin. Certcoin is a PKI system that can replace CA and PGP Webs of Trust and provide effective key checking. However, this scheme runs the risk of privacy leaks.

Literature [3] proposed a privacy-aware blockchain authentication model: PB-PKI. It uses online and offline keys to protect user identity and reduce the risk of privacy leakage. The blockchain-based PKI can be built to provide varying levels of privacy-awareness.

At present, most of the authentication schemes are difficult to apply to the Energy Internet, there are few authentication schemes specifically proposed to the Energy Internet. At the same time, they are difficult to protect the security of users.

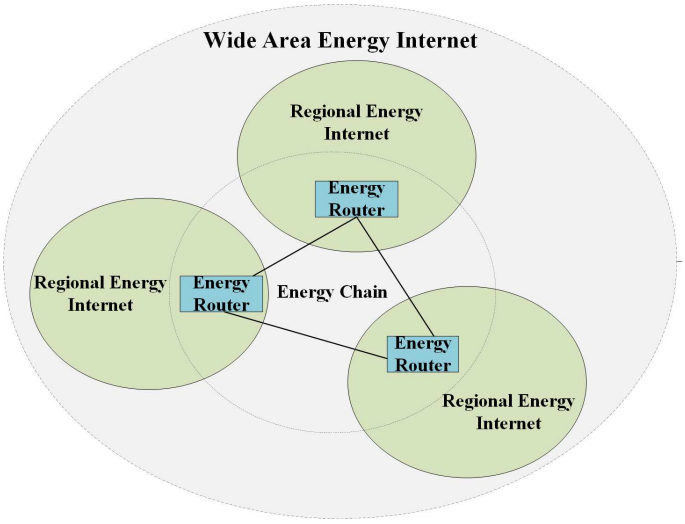


Figure 1: System mode

3 System Model and System Security Requirements

3.1 System Model

In this section, we design the system model of Energy Internet. As shown in Figure 1, it includes Wide Area Energy Internet (Wide EI), Regional Energy Internet (Regional EI), Energy Router (ER), Energy Chain (EC), and Users.

- *Wide EI*: The Wide EI refers to Energy Internet which covers a large region. It can transmit clean energy from remote region to cities over a long distance, thus enabling the wide utilization of renewable energy.
- *Regional EI*: The Regional EI is a comprehensive region energy system that provides energy such as cold, heat, and electricity to users in the region [20]. The region here can refer to the administrative regions in the country or city, such as townships, towns, villages, streets, etc., and can also refer to industrial parks, commercial parks, agricultural parks, residential areas, etc. The Regional EI includes a large number of users.
- *ER*: The ER is an infrastructure in Regional EI, who is responsible for the input, output, conversion, and storage of different energy. The ER connects Regional EI and Wide EI.
- *EC*: All ERs together form a blockchain called Energy Chain (EC). The ER will act as a physical node in the blockchain.
- *Users*: Users are intelligent terminals in Energy Internet, such as family users, electric vehicle, large

power plants and so on. They have a common feature: they are both energy producers and consumers.

3.2 System Security Requirements

First, we explain the authentication scenario in the system. For example, there is a user *Alice* in the region A. *Alice* needs some electrical energy, so Alice asks the energy router ER_A in the region A to get the energy she needs. After confirming her identity, ER_A finds that it can not provide the energy *Alice* needs. So ER_A begins to turn to the energy router ER_B in the neighbor region B for help. After receiving the request, ER_B first confirms the identity of ER_A. Then it checks whether the region B can provide the required energy. If there is enough energy, ER_B transfers the electrical energy to ER_A. If there is not, ask other regions.

In our system under consideration, the ER is honest but curious, that is, they don't change users' energy usage during communication, but they are curious about the specific electrical information of each user. However, the adversary in the region is malicious, namely, actively eavesdrop on communication between different departments, modify communication information or launch replay attacks. Therefore, our security requirements are as follows:

- *Privacy*: User's private information is not revealed to the adversary. ER should know nothing about the details of the user's usage.
- *Confidentiality*: The user's energy usage and bills should be protected against any adversary. Even if an adversary eavesdrops on data transmission links, no useful information can be extracted from them.
- *Authenticity*: The user and ER should be protected from spoofing attacks. An adversary could not falsify the identity of the ER and the user.

4 Definition and Preliminaries

This section reviews the principle and property of cryptographic accumulator.

The notations in Table 1 are used throughout this paper.

Table 1: Notations and definitions

Notation	Definition
EI	Energy Internet
Wide EI	Wide Area Energy Internet
Regional EI	Regional Energy Internet
ER	Energy Router
EC	Energy Chain
RA	Registration Authority
1^λ	security parameter
N	accumulation threshold
aux	auxiliary information
w	witness
z	accumulated value
ID	identity information
PU	public key
PR	private key
h	hash value
k	security parameter $k \leftarrow Gen(1^\lambda, N)$

4.1 One-Way Accumulators

In 1993, Benaloh and de Mare [5] first proposed *one-way accumulators*. It is a one-way hash function that satisfies *quasi-commutativeness*.

Definition 1 (One-way Hash Funtions)**.**

1) For any integer λ and any $h_k \in H_\lambda$, $h_k(\cdot,\cdot)$ is computable in time polynomial in λ.

2) For any probabilistic, polynomial-time algorithm $\mathcal{A}$:

$$Pr[h_k \overset{R}{\leftarrow} \mathcal{H}_\lambda; x \overset{R}{\leftarrow} X_k; y \overset{R}{\leftarrow} Y_k; (x', y') \leftarrow \mathcal{A}(1^\lambda, x, y) : y' \neq y \wedge h_k(x, y) = h_k(x', y')] < \mathbf{negl}(\lambda)$$

where the probability is taken over the random choice of h_k, x, y and the random coins of $\mathcal{A}$.

Definition 2 (Quasi-commutativeness)**.**

A function $f : X \times Y \rightarrow X$ is said to be quasi-commutative if:

$$(\forall x \in X)(\forall y_1, y_2 \in Y)[f(f(x, y_1), y_2) = f(f(x, y_2), y_1)].$$

4.2 Dynamic Accumulators

In 2002, Camenisch and Anna Lysyanskaya [7] proposed *dynamic accumulator*. A dynamic accumulator scheme is *a 7-tuple of polynomial time algorithms*(*Gen*, *Eval*, *Wit*, *Ver*, *Add*, *Del*, *Upd*), where:

- *Gen*: The key generation algorithm, is a probabilistic algorithm used to set up the parameters of the accumulator. *Gen* takes as input a security parameter 1^λ and an accumulation threshold N(an upper bound on the total number of values that can be securely accumulated) and returns an accumulator key k from an appropriate key space $K_{\lambda,N}$. When using an accumulator for a collection of more than N elements, security is not guaranteed.
- *Eval*: The evaluation algorithm, is a probabilistic algorithm used to accumulate a set $L \doteq \{y_1, ..., y_{N'}\}$ of $N' \preceq N$ elements from an efficiently-samplable domain Y_k, where k is some accumulator key from

$K_{\lambda,N}$. *Eval* receives as input$(k, y_1, ..., y_{N'})$ and returns an accumulated value (or accumulator) $z \in Z_k$ and some auxiliary information aux, which will be used by other algorithms. Notice that *Eval* on the same input$(k, y_1, ..., y_{N'})$ must return the same accumulated value, however the auxiliary information aux can differ.

- *Wit*: The witness extraction algorithm, is a probabilistic algorithm that takes as input an accumulator key $k \in K_{\lambda,N}$,a value $y_i \in Y_K$ and the auxiliary information aux previously output (along with the accumulator z) by *Eval* $(k, y_1, ..., y_{N'})$. If y_i is confirmed in L, a witness $\omega_i \in W_k$ is output to prove that y_i is accumulated into z, otherwise return special symbol $\perp$.

- *Ver*: The verification algorithm, is a deterministic algorithm for verifying the identity by witness. *Ver* input (k, y_i, w_i, z), prove that whether y_i is accumulated into z according to witness w_i, output yes or no.

- *Add*: The element addition algorithm, is a (usually deterministic) algorithm that given an accumulator key k, a value $z \in Z_k$ obtained as the accumulation of some set L of less than N elements of Y_k, and another element $y' \in Y_k$, returns a new accumulator z' corresponding to the set $L \cup \{y'\}$, along with a witness $\omega' \in W_k$ for y' and some update information aux_{Add} which will be used by the *Upd* algorithm.

- *Del*: The element deletion algorithm, is a (usually deterministic) algorithm that given an accumulator key k, a value $z \in Z_k$ obtained as the accumulation of some set L of elements of Y_k, and an element $y' \in L$, returns a new accumulator z' corresponding to the set $L \setminus \{y'\}$, along with some update information aux_{Del} which will be used by the *Upd* algorithm.

- *Upd*: The witness update algorithm, is a deterministic algorithm used to update the witness $\omega \in W_k$ for an element $y \in Y_k$ previously accumulated within an accumulator $z \in Z_k$, after the addition(or deletion) of an element $y' \in Y_k \setminus \{y\}$ in (or from) z. *Upd* takes as input(k, y, w, op, aux_{op})(where *op* is either *Add* or *Del*), and returns an updated witness ω' that "proves" the presence of y within the updated accumulator z'.

5 Proposed Scheme

In this section, we introduce our identity authentication scheme. To describe this scheme, we divide the process into three parts: system initialization, ER authentication and user authentication.

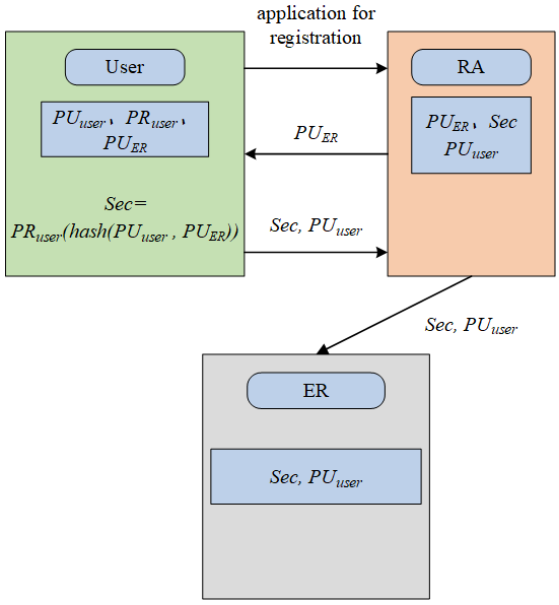


Figure 2: The registration of user

5.1 System Initialization

System initialization includes the generation and distribution of keys and the identity registration of ER and users.

5.1.1 Keys Generation and Distribution

In our scheme, ER calculates its own key according to the main key, and the user's key is generated by himself.

Firstly, RA generates a pair of key MPU and MPR, where MPU is the main public key, MPR is the main private key, and then RA sends MPU and MPR to the ER; Secondly, ER uses a random number γ and (MPU, MPR) to generate its own public key $PU_{ER} = f(MPU, \gamma)$ and private key $PR_{ER} = f(MPR, \gamma)$, where the f is the function that the master key to generate the subkey. Finally, ER uploads the public key PU_{ER} to the RA.

The user generates its own public and private key pairs. In the same way, it will send public key PU_{user} to the RA.

5.1.2 User Registration

When users join the EI, they should apply for registration with ER in their region. The registration process is shown in Figure 2.

After receiving the registration information, RA will check the authenticity of the user's identity, and then sends PU_{ER} to the user. The user calculates his own secret value according to $Sec = PR_{user}(hash(PU_{user}, PU_{ER}))$, which represents his affiliation with the region. At the same time, the user sends its public key PU_{user} to RA, and then RA sends (Sec, PU_{user}) of this user to the ER. ER uses merkle tree to calculate the hash value $MerkleRoot$ of all users' Sec

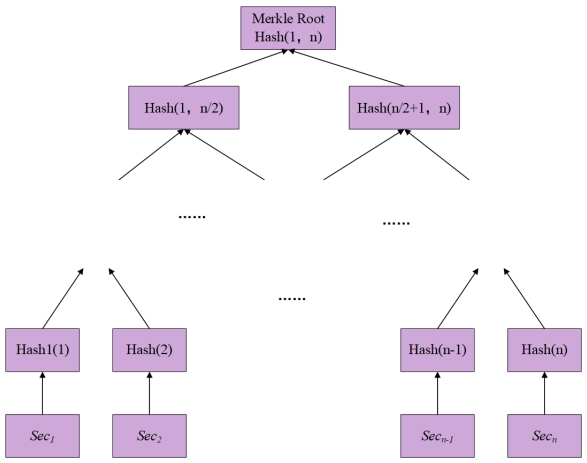


Figure 3: Merkle root

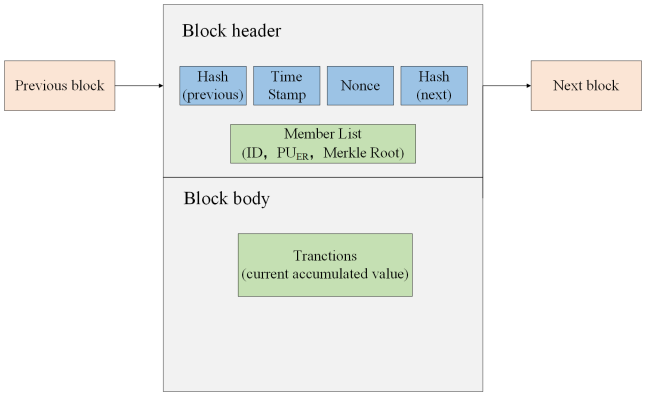


Figure 4: Block design

of this region. The accumulation process is shown in Figure 3.

5.2 ER Authentication

This part describes the process of building the EC and how ERs rely on the EC to complete authentication.

Compared with the normal blockchain, we add *Member List* to the genesis block header, and add the *Current Accumulated Value* to the transaction of the block body. The Member List includes ID, PU_{ER}, and $MerkleRoot$ of all ERs. The structure of the genesis block is shown in Figure 4.

5.2.1 Create

The initial build steps of the EC are as follows:

Step 1. When the EC is initially created, the security parameter $k \leftarrow Gen(1^{\lambda}, N)$ is first to be created.

Step 2. The *Member List* is added to genesis block header, which includes $(ID, PU_{ER}, MerkleRoot)$. The *Member List* here refers to the information of the ERs participating in the construction of EC. PU_{ER} refers to the public key of ER.

Step 3. The accumulated value Z_{init} is obtained by $(Z_{init}, aux) = Eval(k, y_1, y_2, ..., y_n)$, where $y_i = hash(ID, MerkleRoot)$.

Step 4. Broadcast Z_{init} to network, other nodes verify that the accumulated value is correct. If Z_{init} is correct, the block is recognized. If Z_{init} is not correct, discard the block.

Step 5. All nodes calculate the witnesses $w_i \leftarrow Wit(k, y_i, aux)$ of their own.

5.2.2 Node Join

When there is a new ER add to the EC, the steps are as follows:

Step 1. The node's $(ID, PU_{ER}, MerkleRoot)$ will be added to *Member List* first.

Step 2. Calculate new witnesses $(Z_{new}, w, aux_{Add}) = Add(k, Z_{old}, hash(ID, MerkleRoot))$.

Step 3. Broadcast Z_{new} to network, other nodes verify that the accumulated value is correct. If Z_{new} is correct, the block is recognized. Then, the node's $(ID, PU_{ER}, MerkleRoot, aux, w)$ will be written in new block. If Z_{new} is not correct, discard the block.

Step 4. Other nodes update their witnesses $w_i' \leftarrow Upd(k, hash(ID, MerkleRoot), w_i, aux_{Add})$.

5.2.3 Node Delete

When a node is deleted from the EC, the steps are as follows:

Step 1. Verify that this node is in the EC by calculate $Ver(k, hash(ID, MerkleRoot), w, z) = 1$.

Step 2. If the verification is successful, the node will be removed and the accumulated value is recalculated by $(Z_{new}, aux_{Del}) \leftarrow Del(k, Z_{old}, hash(ID, MerkleRoot))$. Then the new accumulated value will be add to the new block. If the verification fails, it will terminate.

Step 3. All nodes verify that the Z_{new} is correct. If the accumulated value is updated correctly, the witness of each node will be updated. Otherwise, it will terminate.

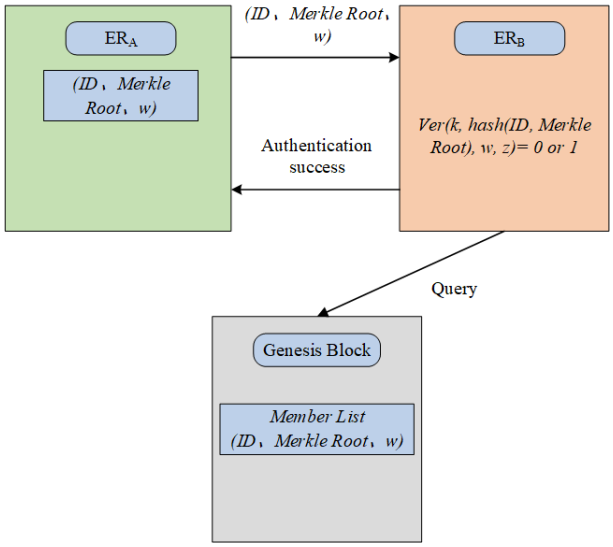


Figure 5: ER authentication

5.2.4 Authentication

Take ER_A and ER_B for example, we explain their authentication process as shown in Figure 5. The premise is that they are already members of the EC. We refer to the literature [15] to complete simple authentication.

Firstly, ER_A sends $(ID, MerkleRoot, w)$ to ER_B. Secondly, ER_B query the genesis block to verify that the information of ER_A is correct. Finally, ER_B calculates $Ver(k, hash(ID, MerkleRoot), w, z) = 0$ *or* 1 to confirm the identity of ER_A.

5.2.5 Mining and Consensus

In our scheme, EC is a consortium blockchain, and the ER, as a node in the chain, has high reliability. Therefore, we use *reputation value* to select the miner node and record the ledger, which was proposed in our previous work [11].

Each ER has its initial *reputation value* C_0. After ER trades with the user, the user needs to evaluate this transaction, and the result will be used for the calculation of the *reputation value*. If the user approves this transaction, the returned evaluation result $C_{approval}$ is a positive value. The user encrypts the evaluation result with its private key and adds the system timestamp T, and then sends the $PR_{user}(C_{approval}, T)$ to ER. ER broadcasts ciphertext to the network. If the user does not approve this transaction, the returned evaluation result C_{blam} is a negative value. The user encrypts the evaluation result with its private key and adds the system timestamp T, and then sends $PR_{user}(C_{blam}, T)$ to ER. ER broadcasts ciphertext to the network. The user encrypts the evaluation result with his private key so that the ER can not forge the evaluation result. System timestamp can prevent outdated evaluation result from replacing the current.

Suppose the cycle is 10 minutes, the mining and billing authority is obtained by the ER who has the highest *reputation value* in this cycle. When a cycle arrives, the ER decrypts all the evaluation results and calculates the total evaluation result D according to the following formula.

$$D = \frac{1}{n}(k \cdot C_{approval} + (n-k) \cdot C_{blam}), k \in \{0, 1, ..., n\},$$

where n is the number of transactions. k is the number of times that the evaluation result is $C_{approval}$.

In order to reduce the impact of previous behavior on the current *reputation value*, we introduce a decreasing function that reduces the weight of the previous *reputation value* with the change of time, then we get the final *reputation value*. Assuming that the *reputation value* of the ER at the ith cycle is C_i, the *reputation value* of the tth cycle can be calculated by the formula.

$$C_t = \sum_{i=0}^{t-1} C_i e^{(-t-i)} + D.$$

ER broadcasts its current *reputation value* in the EC and verifies the *reputation value* of the ER who claims to have the highest reputation value. After the verification is passed, the ER with the highest reputation value obtains the current accounting authority to complete the cycle. If the *reputation value* of the ER is found to be forged, the accounting right of the node will be cancelled.

5.3 User Authentication

In order to ensure that the data sent by the user is safe and confidential, we complete the authentication in the following ways.

The user sends $PU_{ER}(Sec, PU_{user})$ to the ER of his region. The ER decrypts the sent information using its own private key PR_{ER}, and then gets the Sec and PU_{user}. Finally, the ER compares the (Sec, PU_{user}) with the stored (Sec, PU_{user}). If successful, the authentication is completed.

6 Performance and Security Analysis

6.1 Security Analysis

In this section, we analyze the security of the proposed scheme. According to the security requirements proposed in Section 3, we discuss whether the proposed scheme meets the requirements.

- *Privacy*: In the scheme, the RA produces the main public key MPU and the main private key MPR, each ER of the EI calculates its own public and private key based on (MPU, MPR). In this way, the private key is known only to himself, which can avoid the problem of key exposure. From the user's point

of view, the user completes the registration at the RA, and the ER can only obtain the user's Sec and PU_{user}. Using Sec and PU_{user} to complete the identity authentication without knowing the user's true identity, so it cannot specifically associate the user's personal information. This can protect the privacy of users very well.

- *Confidentiality*: The data of users are sent to the ER after being encrypted by the ER's public key PU_{ER}. The ER's private key PR_{ER} is generated according to main private key MPR, which is known to himself. The user's data have been transmitted in ciphertext formats, and the attacker can not get any information about the data.

- *Authenticity*: We build the system through the blockchain, which can ensure that the authenticity and traceability of data. At the same time, our scheme can avoid forgery very well. Take the Sybil Attack as an example, Sybil Attack is a common security problem in consortium blockchain. In some consensus plugins represented by PBFT, the number of nodes directly affects the consensus result. A malicious node will destroy the blockchain network by disguising itself as multiple good nodes. Our scheme uses *reputation value* to reach consensus in the EC. The node which has the highest *reputation value* will obtain mining and recording rights. The *reputation value* is not only calculated by the user's evaluation of transactions but also add a timestamp. So, it cannot be forged. The malicious node cannot implement a Sybil Attack on our scheme.

The advantages and disadvantages of our scheme and other schemes are shown in Table 2, where "yes" means that it has the advantage or it can resist the attack, "no" means that it does not have this advantage or it cannot resist the attack, "/" means that it does not exist this attack.

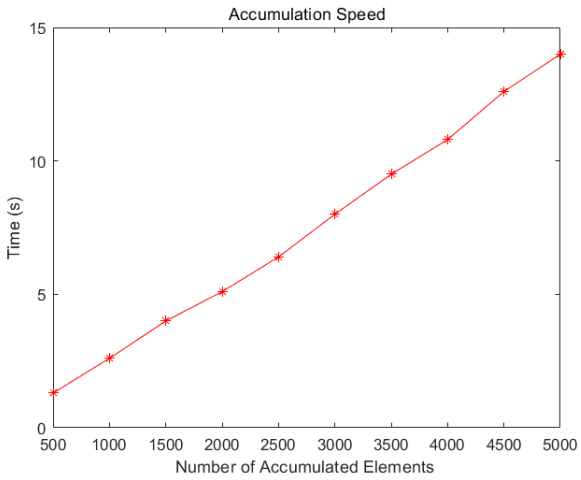


Figure 6: Accumulation speed

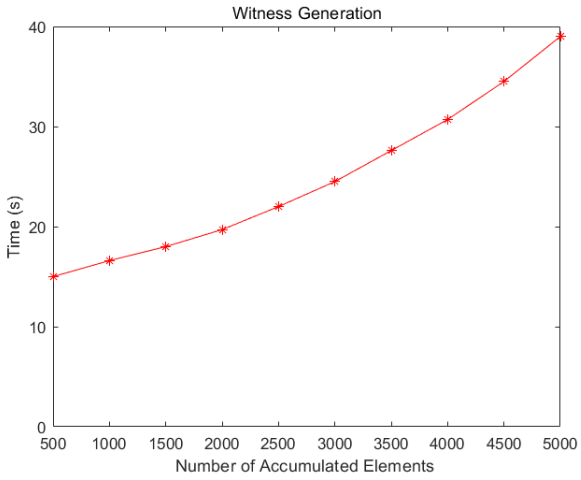


Figure 7: Witness generation speed

6.2 Performance Analysis

In order to evaluate the proposed identity authentication scheme with energy privacy preservation, we conduct the simulations on a 64 bit computer with Intel(R) Core(TM)i7-7700HQ 2.80GHz CPU and 8G RAM, using Python.

We use blockchain instead of CA in traditional PKI authentication scheme. The cryptographic accumulator used in this paper is the RSA accumulator. We built about 5000 pairs of 512-bit public-private key pair data sets to test the authentication efficiency of two schemes. We first tested the performance of the RSA accumulator, mainly the speed of accumulation and witness generation. It can be seen in Figure 6 and Figure 7. The result shows that cryptographic accumulator has good performance, not only accumulation speed but also witness generation speed. The EC can be built quickly without spending too much time.

Then we test the authentication efficiency of the cryptographic accumulator. In the blockchain, the traditional authentication scheme needs to traverse all the blocks to complete the authentication. In our scheme, the identity authentication is independent of the length of the blockchain. It only needs to calculate whether the accumulated value Z is equal.

As can be seen from the Figure 8, in the traditional scheme, with the number of block increases, the authentication time increases and the efficiency becomes lower. In our scheme, the time increases slowly. The reason is that time is only spent in running the verification algorithm. Therefore, compared to the traditional scheme, our scheme is more efficient.

Table 2: Notations and definitions

Schemes	Malicious User	Spoofing Attack	Man in The Middle	Distributed	Lightweight
PKI	No	/	No	No	Yes
Chen *et al.* [8]	Yes	No	No	Yes	Yes
Hammi *et al.* [13]	Yes	Yes	Yes	Yes	No
Fromknecht *et al.* [10]	No	Yes	Yes	Yes	Yes
Proposed Scheme	Yes	Yes	Yes	Yes	Yes

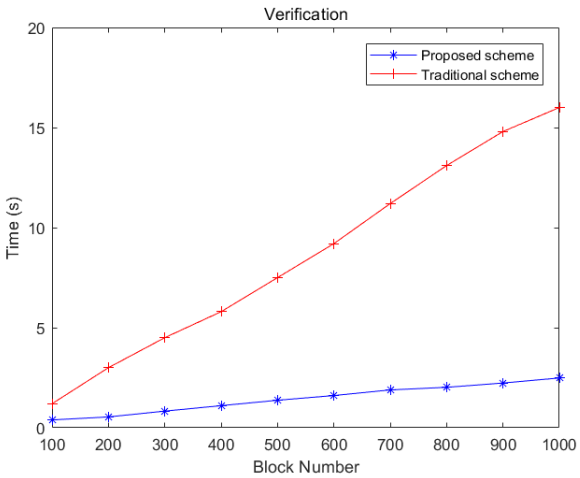


Figure 8: Verification time

7 Conclusion

In this paper, we have proposed an identity authentication scheme of Energy Internet Based on Blockchain. First, we give a system model of the Energy Internet, and then a secure identity authentication scheme is proposed. ERs build a blockchain with cryptographic accumulator to accomplish efficient authentication. Moreover, user's authentication depends on the ER in its region. In addition, the security analysis shows that our scheme achieves privacy and confidentiality, as well as authenticity. The experimental results show that the authentication efficiency is indeed better than traditional PKI authentication scheme. For our future work, we intend to explore the better way to combine blockchain and Energy Internet to improve our current scheme.

Acknowledgments

This work was supported by NSFC Grants (No. 61772327 No. 61532021). We would like to express our gratitude to the anonymous reviewers for their valuable comments.

References

[1] M. Al-Bassam, "Scpki: A smart contract-based PKI and identity system," in *Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts*, pp. 35–40, 2017.

[2] M. H. Au, Q. Wu, W. Susilo, and Y. Mu, "Compact e-cash from bounded accumulator," in *Cryptographers'Track at the RSA Conference*, pp. 178–195, 2007.

[3] L. M. Axon and M. Goldsmith, "PB-PKI: A privacy-aware blockchain-based PKI," in *The 14th International Conference on Security and Cryptography*, 2016. DOI: 10.5220/0006419203110318.

[4] N. Barić and B. Pfitzmann, "Collision-free accumulators and fail-stop signature schemes without trees," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 480–494, 1997.

[5] J. Benaloh and M. D. Mare, "One-way accumulators: A decentralized alternative to digital signatures," in *Workshop on the Theory and Application of of Cryptographic Techniques*, pp. 274–285, 1993.

[6] P. Camacho, A. Hevia, M. Kiwi, and R. Opazo, "Strong accumulators from collision-resistant hashing," in *International Conference on Information Security*, pp. 471–486, 2008.

[7] J. Camenisch and A. Lysyanskaya, "Dynamic accumulators and application to efficient revocation of anonymous credentials," in *Annual International Cryptology Conference*, pp. 61–76, 2002.

[8] X. Chen, X. Hu, Y. Li, X. Gao, and D. Li, "A blockchain based access authentication scheme of energy internet," in *The 2nd IEEE Conference on Energy Internet and Energy System Integration*, pp. 1–9, 2018.

[9] P. Fan, Y. Liu, J. Zhu, X. Fan, and L. Wen, "Identity management security authentication based on blockchain technologies," *International Journal Network Security*, vol. 21, no. 6, pp. 912–917, 2019.

[10] C. Fromknecht, D. Velicanu, and S. Yakoubov, "CertCoin: A namecoin based decentralized authentication system 6.857 class project," *Unpublished class project*, 2014. (`https://courses.csail.mit.edu/6.857/2014/files/19-fromknecht-velicann-yakoubov-certcoin.pdf`)

[11] F. L. Tian, X. X. Tian, X. Chen, "Blockchain-based smart meter authentication scheme (in chinese)," *Journal of East China Normal University (Natural Science)*, vol. 5, pp. 135–143, 2018.

[12] G. G. Wan, S. J. Zhou, and Z. G. Qin, "An overview of the one-way accumulator technology (in chinese)," *Computer Science*, vol. 32, no. 8, pp. 57–59, 2005.

[13] M. T. Hammi, B. Hammi, P. Bellot, and A. Serhrouchni, "Bubbles of trust: A decentralized blockchain-based authentication system for IoT," *Computers & Security*, vol. 78, pp. 126–142, 2018.

[14] J. Li, N. Li, and R. Xue, "Universal accumulators with efficient nonmembership proofs," in *International Conference on Applied Cryptography and Network Security*, pp. 253–269, 2007.

[15] T. Ling-Tao, X. Min, and J. Yu-Rong, "Research on methods of improving efficiency of identity authentication based on blockchain (in chinese)," *Application Research of Computers*, vol. 36, no. 10, 2019.

[16] S. Nakamoto, *et al.*, *Bitcoin: A Peer-to-Peer Electronic Cash System*, 2008. (`https://bitcoin.org/bitcoin.pdf`)

[17] X. Tian, F. Tian, A. Zhang, and X. Chen, "Privacy-preserving and dynamic authentication scheme for smart metering," *International Journal Network Security*, vol. 21, no. 1, pp. 62–70, 2019.

[18] K. Wang, J. Yu, Y. Yu, Y. Qian, D. Zeng, S. Guo, Y. Xiang, and J. Wu, "A survey on energy internet: Architecture, approach, and emerging technologies," *IEEE Systems Journal*, vol. 12, pp. 2403–2416, Sep. 2018.

[19] W. Wang and Z. Lu, "Cyber security in the smart grid: Survey and challenges," *Computer networks*, vol. 57, no. 5, pp. 1344–1371, 2013.

[20] S. You and P. Song, "A review of development of integrated district energy system in denmark," *Distribution & Utilization*, vol. 34, no. 12, pp. 2–7, 2017.

Biography

Xiuxia Tian received the MS degree in applied cryptography-based information security from Shanghai Jiaotong University in 2005, and the PhD degree in database security and privacy preserving in cloud computing from Fudan University in 2011. She is currently a professor in the College of Computer Science and Technology, Shanghai University of Electric Power. She is a visiting scholar of two years at UC Berkeley working with groups of SCRUB and SecML. She has published more than 40 papers and some papers are published in international conferences and journals such as DASFAA, ICWS, CLOUD, and SCN. Her main research interests include database security, privacy preserving (large data and cloud computing), applied cryptography, and secure machine learning.

Xi Chen Graduate. College of Computer Science and Technology in Shanghai University of Electric Power. He research interests mainly focus on the blockchain and security identity authentication (Email:chenjss@yeah.net).

Siqian Li Graduate. College of Computer Science and Technology in Shanghai University of Electric Power.

Enhanced Deduplication Protocol for Side Channel in Cloud Storages

Jie Ouyang[1], Huiran Zhang[1,2,3], Hongqing Hu[1], Xiao Wei[1,2,3], and Dongbo Dai[1]
(Corresponding author: Dongbo Dai)
School Computer Engineering and Science, Shanghai University, Shanghai 200444, China[1]
Shanghai Institute for Advanced Communication and Data Science, Shanghai 200444, China[2]
Materials Genetics Institute, Shanghai University, Shanghai 200444, China[3]
(Email: dbdai@shu.edu.cn)
(Received July 31, 2019; Revised and Accepted Dec. 3, 2019; First Online Feb. 3, 2020)

Abstract

Cloud storage usually adopt client-based deduplication, which can achieve considerable savings in both storage and bandwidth. However, an attacker can carry out a steal-file-content (SFC) attack, exposing data privacy. In this paper, a simple yet effective scheme, called double bytes transport protocol (DBTP), is proposed. In this scheme, the client-side requests deduplication checks of the double chunks simultaneously, and the server-side receives the deduplication request and response with a reasonable value. The result demonstrates that DBTP can significantly mitigate the side channel's risk while maintaining the high bandwidth efficiency of deduplication.

Keywords: Cloud Storage; Data Privacy; Deduplication Check; Side Channel

1 Introduction

As a convenient and popular service model, cloud storage services enable users to store and access various resources in the cloud on demand. Major cloud storage providers offer multiple file types of storage. According to a report of an International Data Corporation (IDC), the total amount of digital data in global data centers is exploding rapidly [16]. According to the latest information, the volume of global digital data created and replicated is up to 33 Zettabytes in 2017, while with the advent of the era of big data, it will exceed 175 Zettabytes in 2025. IDC analysis also shows that 49% of the data will be stored in a public cloud environment and close to 75% data has a copy. This similar phenomenon also exists in the research report of Microsoft Research Institute [10]. Based on the above surveys, a large amount of redundant data will storage in the cloud. If the cloud storage services cannot process these redundant data, it will cause waste of cloud storage space and increase network bandwidth when users upload files to cloud storage.

Nowadays, there are two main ways to implement the deduplication check. They are service-side deduplication check and client-side deduplication check respectively. In the former, the users must upload the data to the server-side first, then server-side deletes the redundant data again. This approach can reduce storage overhead, but cannot save bandwidth. In the latter, client-side divide the file into multiple chunks and send chunks hash signatures to cloud and check for the existence or inexistence status of chunks. It can not only delete redundant data but also save bandwidth. Therefore, the commercial cloud storage services mostly adopt the cross-user client-side deduplication check to maintain a specified number of copies of files [13]. It can achieve the maximize the utilization of storage space.

Although cross-user client-side deduplication check can bring the above benefits, it will also lead to the threat of side channel [4]. According to the traditional deduplication check, if fingerprint matching of a chunk does not exist, it means that there is no such chunk on the server-side. The server-side will feed back to the client-side and request to upload the file chunk. If fingerprint matching is successful, the file chunk does not need to be uploaded. Based on the above observation, the client-side needs to receive the exact feedback from the server-side and decides whether to upload file chunk. However, the users can always obtain the determined existence or inexistence information of the chunk by observing the data traffic transmitted between the client-side and the server-side. Thus, existence or inexistence response from server-side products the threat of data privacy leakage. and creates a side channel [4]. This privacy leakage can lead to the following potential attacks.

Identifying files: In order to identify the existence or the inexistence status of a specific file, an attacker performs a deduplication check by using well-designed file template.Violent file cracking can be seen as the most straightforward privacy leak.

Stealing the file content: Normally, an attacker can check whether a particular file is stored in a cloud stor-

age. However, what is more serious is that the attacker might apply this attack to multiple versions of the same file and obtain the secret information of files by brute force. As shown in Figure 1, the file T is the target file to be attacked. The attacker has obtained other information about file T except x. In order to steal the secret content of x, the attacker try to use a way of brute force. It uses n files with all possible values of x (x_1, x_2, ..., x_n) to probe deduplication respectively. If only file F_n does not upload the file chunk, the attacker will be very sure that the secret information is x_n. Because the server-side identifies the existence of the unique chunk x_n, the content of x is stolen by the attacker. If the number of possible versions of the target file is moderate [5], the success rate of this attack is very high.

Covert Channel: As long as the two parties reach a consensus, a covert channel can bypass the censorship and communicate with each other [12, 22]. It means that file existence status can be used as the medium of communication.

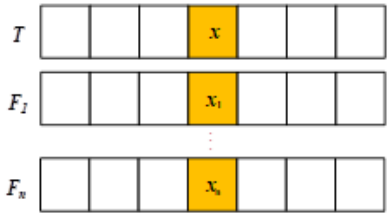


Figure 1: The attack in chunk-level deduplication

The obvious drawback of the traditional deduplication check is that the attacker can continuously use templates of file chunks to violently verify the sensitive information of the file which are stored in server-side, but attacker usually blocks the upload of file chunks. Because the templates of these file chunks can be used repeatedly, thus this is one of the key factors that leakage of user data privacy. However, the server-side does not take effective measures to against this threat.

In order to address these challenges, we propose double byte transport protocol (DBTP), a simple yet effective strategy to ensures a balance between privacy security and deduplication check benefit. It achieves the two-side privacy (existence privacy and inexistence privacy). Since deduplication check of single chunk make the user clearly knowing whether the chunk exists on the cloud, the strategy uses auxiliary chunk to perform the deduplication check on double chunks at once and return the reasonable value. This means that there exist enough room to confuse the attacker's judgment when client-side performs deduplication check. In particular, in order to solve side channel, there is set a list D for recording dirty chunks (file chunks that need to be uploaded but not uploaded under normal deduplication check). Chunk list H for recording chunks that no longer needs to be checked. It helps save bandwidth and storage overhead. As can be seen from Table 1, compared to existing solutions, the DBTP protocol has its own advantages, such as no redundant parameter configuration, no additional hardware, two-side privacy protection and storage space savings.

Table 1: Comparisons between DBTP and other schemes

	NoArgument	NoHardware	Privacy	Save
RT	No	Yes	No	No
ZEUS	Yes	Yes	No	Yes
ZEUS$^+$	No	Yes	Yes	No
RARE	Yes	Yes	Yes	No
Mozy	No	Yes	No	No
Shin	Yes	No	Yes	Yes
Heen	Yes	No	Yes	Yes
DBTP	Yes	Yes	Yes	Yes

The scheme achieves the two-side privacy and maintains the deduplication rate of the original deduplication check. Specifically, if a double chunks combination is confirmed not in the cloud by deduplication check, the client-side will only upload one of the chunks first. The remaining chunk will be combined with the other chunk that is selected from the remaining list chunks. It is different from RARE [5] which implements privacy security with excessive redundant chunks upload. Other similar programs, just like ZEUS [15] can't satisfy the inexistence privacy and RT [4] only offer the inexistence privacy. ZEUS$^+$ [15] is based on ZEUS and RT to enhance the privacy security, but the setting of the random threshold parameter severely reduces the deduplication rate.

We conducted a detailed security analysis about the return values of the server-side in the DBTP scheme. It demonstrates that DBTP can effectively resist side channel attack.In addition, this method only involves the interactions between the client-side and server-side and does not need the extra hardware [23]. It is only minimally modified on the original deduplication check technology.

The DBTP implementation does not require additional parameter configuration. Most solutions [4, 12] are to ensure privacy security through random threshold parameters. ZEUS$^+$ [15] adopt a random threshold chosen uniformly in a range [2, d] to maintain an inexistence privacy.

2 Background and Related Work

2.1 Deduplication in Cloud Storages

Data deduplication is an effective technique to eliminate the redundant data which results in storage saving directly [8, 20]. For example, as shown in Figure 2, the client-side of user Aaron first uploads the file fm, then the file f_m will be divided into A, B, C, D, E, and F chunks by the dicing algorithm. When the hash values of these chunks are matched, it is found that these chunks are not in the cloud.Therefore, they are all uploaded to the server-side. Later, another client-side of user Beck

wants upload file fn to own cloud folder. After deduplication check, he or she founds that chunks E, F, and C already exist in the cloud. Thus, the user Beck only needs to upload H, I and J, which are three chunks that did not exist in the cloud. The result is that server-side saves storage and respective bandwidth. It is foreseeable that as the size of users increases, the amount of redundant data is bound to increase more. At that time, the benefits of deduplication check must be considerable. Xia *et al.* [18] present a P-Dedupe system. It uses pipeline and parallelization techniques to accelerate the deduplication process. In our work, we focus on cross-user client-side data deduplication. In addition, different chunk segmentation algorithms can use different slice sizes for files. For example, Dropbox [3] performs the deduplication check and the file is partitioned to some chunks with a determined value of 4MB size.

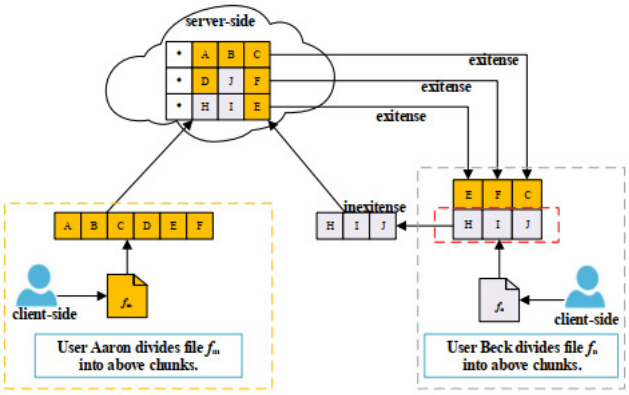


Figure 2: An example of deduplication check

The most mainstream scheme of deduplication check is the cross-user client-side data deduplication. Cross-user means that deduplication check is executed in the storagespace shared by all users. Compared with deduplication check under single-user, redundant data deletion rate based on cross-user has increased a lot. Client-side means that the deletion of redundant data is performed on the client. The process is that the client-side sends the hash value of the chunks to the server-side. According to the hash matching results of the server-side, the client-side determines whether the chunks are to be uploaded to the server-side. It is illustrated in Figure 3, where the client-side sends direction request and server-side gives direction response. It is used to tell user whether the chunk need to be uploaded to the cloud.

2.2 Related Work for the Side Channel

Especially, after performing hash matching of file chunk,the cloud needs to deterministically return the existence status.The problem of side channel which is appearing in the traditional deduplication check. It is defined by Harnik *et al.* [4]. In order to address the problem of side

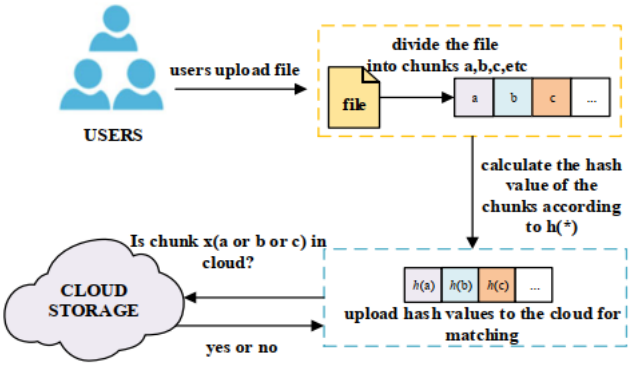


Figure 3: Deterministic response of data deduplication

channel,they recommend using a random threshold solution (RT). The server-side assigns a threshold t_x which is belongs to [2, d]. Thus, it is only known by the server-side. When a file chunk c_x is uploaded and the number of existing copies is greater or equal to t_x, the system will perform client-side deduplication. The problem of this scheme is the uncontrollability of the d value. If the d is too large, the number of copies will be too much. If the d is too small, it will lead to lower security. Thus, the choice of this parameter is difficult. However, this random threshold solution also has privacy issues. If the number of chunk copies recorded in the cloud exceeds the specified threshold t_x, the deduplication check will expose the privacy of file existence. Compared to the global threshold solution of Harnik, Lee and Choi [11] adopted a random threshold t_i at each uploaded chunk. It is claimed to show stronger privacy than Harnik *et al.'s* solution. Armknecht *et al.* [1] recently proved that deduplication thresholds uniformly sampled from [1, B] achieve the optimal defense for the natural privacy measure. In particular, Wang *et al.* [9] designed the deduplication thresholds based on a gametheoretic approach. Unfortunately, all of the above proposals are based on the RT category, thus have the same weaknesses.

The idea of Mozy [21] is that only small-size files contain sensitive information, but large-size files such as music and movie are not sensitive. Based on such assumptions, the scheme designs a threshold x for the file size. If the size of the file is smaller than x, it is treated as the small file, otherwise it is regarded as the large file. Therefore, when the size of the file is larger than x, the deduplication check will be performed, otherwise it will not. Obviously, this lack of theoretical support, because there is no necessary connection between the size of the file and the importance of sensitive information.

Many previous studies focus on deduplication for periodical backup streams [7, 12]. Spatial or temporal locality has been exploited in [17]. Yu *et al.* [19] propose ZEUS which can achieve weak existence privacy but cannot achieve inexistence privacy. In order to implement two-side privacy, Yu *et al.* further propose ZEUS$^+$. It is to combine the use of ZEUS and RT.

In addition to above solutions, there exist some researchers recommend using the extra hardware between the client-side and server-side to enhance the privacy of data security. S. Li *et al.* [2] propose a secure and efficient client-side encryption deduplication scheme (CSED). This solution introduces a private key server to generate MLE keys to resist brute force attacks. Shin and Kim [12] propose a deduplication protocol. It implements differential privacy check based on an independent server bridging between the client-side and server-side. Similar methods is that Heen *et al.* [22] considered to set trusted gateway bridges between the users and cloud. Based on the above methods of using additional hardware, the deterministic relationship that between client-side requests and server-side responses can be broken.

3 Design and Implementation

As the methods mentioned above, there exist defects in the processing of channel problems. In this section, the DBTP algorithm is presented. It is used to solve the channel problem and keep the benefits of deduplication check. To ease reading, we summarize major notations, they are used to construct the DBTP in Table 2.

Table 2: List of notations used in the DBTP

Symbols	Description
f_x	The file that will to be uploaded
c_n	The chunks in which the file f_x is divided
φ	Chunk size
H	chunks that no longer need to be checked
K	K is a collection of all file chunks
tag	assists in deduplication checking
h(x)	Hash function

3.1 The Simple Introduction of DBTP

The biggest problem with the traditional deduplication protocol is that deduplication check with only one chunk at a time. The server-side explicitly returns the state of a chunk's existence. Thus, the attacker can judge whether the chunk has been transmitted by observing network traffic. In DBTP, the basic idea is to use the flag chunk(tag) for auxiliary transmission under different transmission conditions. When the user runs the local client program, the client-side will generate a flag chunk according to the agreement with the cloud. According to the agreement, the cloud defaults the flag chunk already exists. The combination method of double chunks is only [x_i, x_{i+1}] or [x_i, tag]. The deduplication check on the other double chunk combination is based on the specific implementation details of the DBTP algorithm.

3.2 Scheme Design in Detail

The cloud returns 0 to client indicate that the chunk does not exist in the cloud. If cloud returns 1 means the opposite. As you know from the Figure 4, t_i denotes the number of chunks that the cloud requires client-side to upload. The t_1 represents both chunks c_1 and c_2 are not in the cloud storage, the t_2 represents c_1 is not but c_2 is in the cloud storage, and so on.

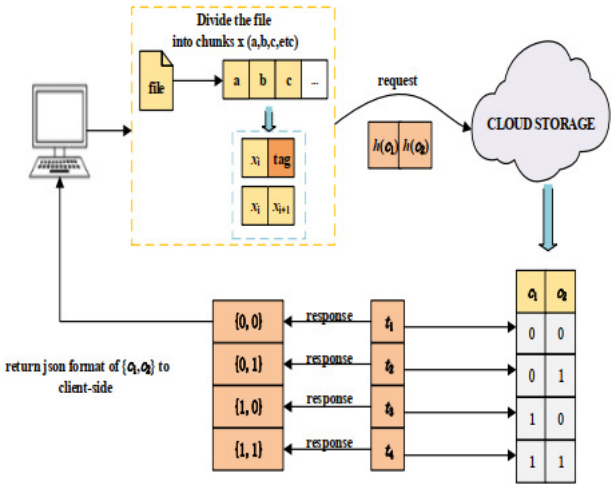


Figure 4: The process of the scheme

When the uploaded file is split, there are only two types of combinations. One is the combination of ordinary chunks like [x_1, x_2], neither x_1 nor x_2 is a tag. The other is the combination of ordinary and tag chunks like [x_i, tag]. Deduplication check for a hash list of two chunks at the same time. Obviously, the number of chunks required to be uploaded can only be selected in 0, 1, and 2. First, except for t_4, the others cannot equal 0. Otherwise, the chunk will be missing. Thus, DBTP needs to meet $t_1 \neq 0$, $t_2 \neq 0$, and $t_3 \neq 0$. Specifically, if the attacker is interested in chunk c_2, he or she can upload two chunks of [c_1, n] first. Thus, c_1 must exist in the cloud. Then, the attacker can upload [c_1, c_2]. Obviously, if $t_3 \neq t_4$ and the state value of c_1 both are 1, the existence state of c_2 can be easily determined. In this sense, $t_3 = t_4$ needs to be satisfied. Similarly, it can be obtained that $t_2 = t_4$ must also be satisfied.

It is easy to draw the following conclusions that $t_2 = t_3 = t_4$ and the values must be selected from 1 and 2. Thus, the value of [t_1, t_2, t_3, t_4] can be [1,1,1,1], [2,1,1,1], [1,2,2,2] and [2,2,2,2]. Nonetheless, if the result returned by the serverside is [2, 2, 2, 2], it can offer the strongest privacy, but deduplication will be ineffective. However, [2,1,1,1] clearly exposed the privacy of inexistence which c_1 and c_2 do not exist in the cloud. Therefore, we can only choose from [1,1,1,1] and [1,2,2,2]. Obviously, [1,2,2,2] is not conducive to bandwidth savings. Based on the above analysis, the most suitable values of t are [1, 1, 1, 1]. It means that result will be returned to client-side in Table 3.

Table 3: Design for DBTP

t	[c_i,c_j]	the chunk is uploaded by client-side
$t_1 = 1$	[0,0]	c_i or c_j
$t_2 = 1$	[0,1]	c_i
$t_3 = 1$	[1,0]	c_j
$t_4 = 1$	[1,1]	c_i or c_j

3.3 DBTP Algorithm Description

The algorithm shows the details of DBTP. First,in order to prevent the illegal behavior of malicious attackers, the system needs to verify the user's identity information. In the registration stage, the user needs to submit some personal information to the server. After receiving the information, the server will issue a smart card to the client, which contains some security parameters for later authentication [2, 18, 19]. After the registration phase, users can access the server in the authentication phase. Only users with valid smart cards and corresponding passwords can be successfully verified by the server. In that case, users without permission will not be able to access the system for malicious attacks. Besides this, we can also adopt an efficient certificateless conditional privacy preserving authentication scheme [2, 14] or a privacy preserving public auditing scheme [6]. Chunk list H store chunks that have been uploaded. K is a collection of all file chunks. When the user attempts to upload a file f_x to the cloud. First of all, the file f_x will be divided into chunks by size φ. The number of file chunks may be odd or only the last a chunk is left to be checked. Thus, it need use the tag chunk to assist the deduplication check (line 02-03). It should be emphasized that check of tag will return 1 by default.

Specifically, the client-side selects two different chunks, h(c_i) and h(c_j), to carries out deduplication check (line 05-09). If i is equals to j means that only the last chunk needs to be checked, so it will be matched to the tag chunk for deduplication check (line 05-07). For the two kind of combinations above, client-side performs deduplication check and cloud sends the values of chunk status according to Figure 4 (lines 10-21). Depending on Table 2, client-side receives values of JSON format from cloud and decides which chunk will be uploaded (lines 22-37). Importantly,client-side save the chunk that is uploaded and exist in the cloud already to the H list. When two chunks are combined next time, the chunks in the H list will be ignored. It will speeds up the deduplication check.

Algorithm 1 DBTP

```
1: Begin
2: client partitions f_x into chunks c_1,...,c_n.
3: create a chunk tag = c_{n+1}.
4: while i,j ∈ K and i,j ∉ H do
5:    random selection of two chunks [c_i ,c_j] and i ≠ j.
6:    if i == j then
7:       h(c_j) = h(tag).
8:    end if
9:    client performs deduplication on [h(c_i) ,h(c_j)].
10:   if c_i and c_j not in cloud then
11:      cloud responses [0,0] according to Figure 4.
12:   end if
13:   if c_i not in cloud and c_j in cloud then
14:      cloud responses [0,1] according to Figure 4.
15:   end if
16:   if c_i in cloud and c_j not in cloud then
17:      cloud responses [1,0] according to Figure 4.
18:   end if
19:   if c_i in cloud and c_j in cloud then
20:      cloud responses [1,1] according to Figure 4.
21:   end if
22:   if client receives [c_i ,c_j]=[0,0] then
23:      client uploads c_i or c_j for Table 2.
24:      H = H ∪ c_i or c_j
25:   end if
26:   if client receives [c_i ,c_j]=[0,1] then
27:      client uploads c_i for Table 2.
28:      H = H ∪ c_i and c_j
29:   end if
30:   if client receives [c_i ,c_j]=[1,0] then
31:      client uploads c_j for Table 2.
32:      H = H ∪ c_i and c_j
33:   end if
34:   if client receives [c_i ,c_j]=[1,1] then
35:      client uploads c_i or c_j for Table 2.
36:      H = H ∪ c_i and c_j
37:   end if
38: end while
39: End
```

4 Performance Evaluation

4.1 Experiment Settings

Based on Linux system, we have built a preliminary programming and testing environment. The scheme is tested on centos-release-6-8 on Intel(R) Xeon(R) CPU E5-26xx v4. DBTP is implemented by Python 3.7.6 platform and MySQL database.

4.2 Experiment Results

DBTP can realize two-side privacy to avoid side channel attack. The privacy experiment uses 10M, 20M and 30M files as the target test object. The cut size φ of each file is set to 2K, 3K, 4K, 5K respectively. The experimental results are shown in Figures 5, 6 and 7.

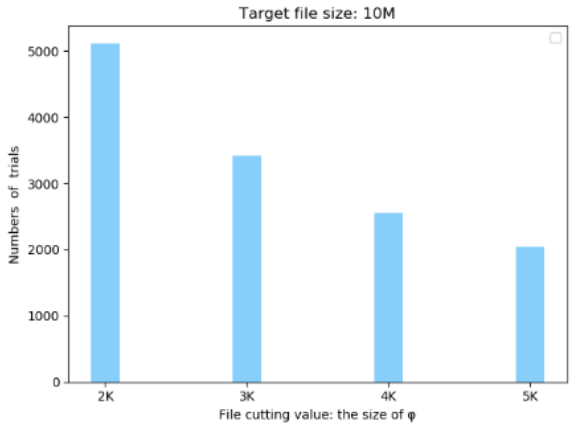


Figure 5: Target file size(10M)

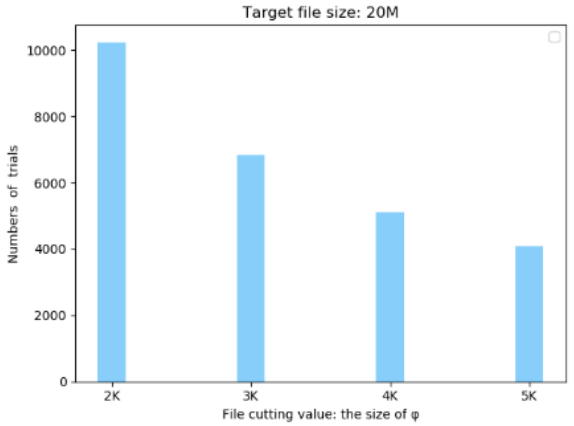


Figure 6: Target file size(20M)

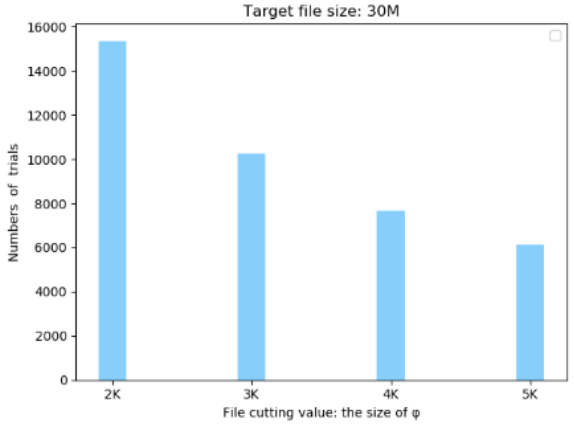


Figure 7: Target file size(30M)

Deduplication ratio is an important indicator to measure the efficiency of deduplication check. Results are shown in Figures 8, 9 and 10. The deduplication ratio is defined as follows $\alpha = A/S = (S\text{-}B)/S$. The formula is explained as follows, α indicates the deduplication rate, and A represents the deleted file size for performing deduplication check operation. The value of A is equal to the difference between S and B, where S represents the total size of the file thatwas not subjected to the deduplication process at the time of uploading, and B represents the file size that has been uploaded to the cloud. The deduplication rate is not only affected by the deduplication strategy, but also by chunk size.

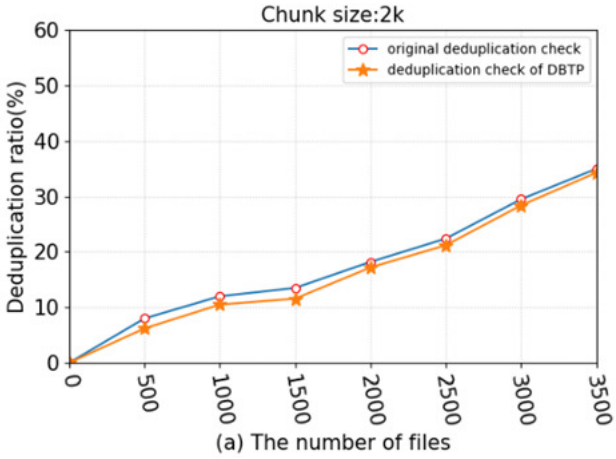


Figure 8: Deduplication ratio comparison(2k)

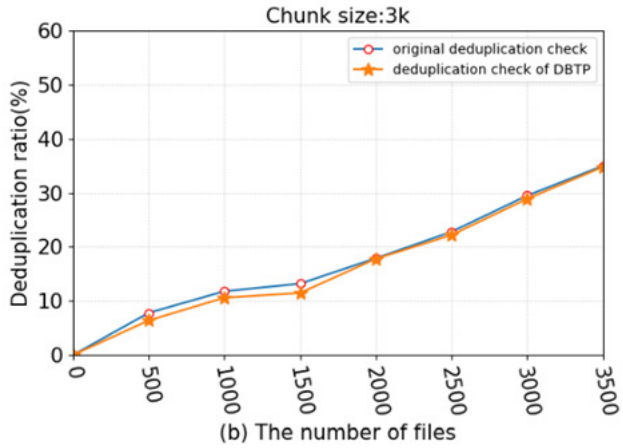


Figure 9: Deduplication ratio comparison(3k)

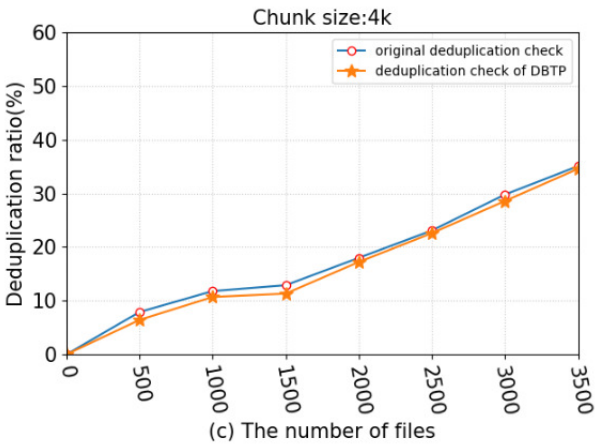


Figure 10: Deduplication ratio comparison(4k)

4.3 Result Analysis

In this section, we have completed two experiments. The former is the experiment of DBTP security privacy for avoiding side channel attack and the latter is a comparison experiment between DBTP and the original schema deduplication rate. Based on Figures 5, 6, and 7, we examine files of sizes 10M, 20M, and 30M, respectively. For example, the target file T, the attacker does not know about a small part of the file, but this part is the privacy part that the attacker is more interested in. Now, we use a computer B to play an attacker to upload different file templates. These templates only modify the imaginary privacy part. The number of probes for each file upload is shown on the y-axis. Through traffic analysis, whether it is a target file of size 10M, 20M or 30M, when violent trials are cracked on them, file blocks are uploaded every time. Because the attacker cannot observe the transmission of zero traffic, it is impossible to judge whether the modified data part matches the target file. Based on the above analysis, the DBTP protocol protects the privacy of data.

From the Figures 8, 9 and 10, compared the deduplication ratios of the two methods. It is obvious that the original deduplication check has relatively higher deduplication ratio compared to the DBTP method. Because the original deduplication check finds the maximum opportunity to eliminate the redundant chunk. Fortunately, deduplication check of DBTP guarantees the two-side privacy of data, but the deduplication ratio is only a little reduced, and the gap between them is within an acceptable range. In general, DBTP precisely guarantees a good balance between deduplication efficiency and data privacy protection.

5 Conclusions

Although cloud storage service providers have widely adopted cross-user client-side deduplication check to reduce redundant data and communication costs, it leaks the privacy of the chunk existence or inexistence status, resulting in more threats like side channel. In this paper, we propose a solution, DBTP, based on tag chunk for auxiliary transmission. This scheme leaks zero-knowledge for side channel. In other words, it can prevent the attacker from gaining the existence or inexistence status information from deduplication check. DBTP implements a stronger two-side privacy and performance guarantee based on minimal modification of the ordinary deduplication mechanism.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (No.2018YFB0704400).

References

[1] A. Chiniah, J. A. D. Dhora, and C. J. Sandooram, "Erasure-coded network backup system (ECNBS)," in *International Conference on Information, Communication and Computing Technology*, pp. 35–43, 2017.

[2] Z. Guo, "Cryptanalysis of a certificateless conditional privacy-preserving authentication scheme for wireless body area networks," *International Journal of Electronics and Information Engineering*, vol. 11, no. 1, pp. 1–8, 2019.

[3] Z. Han, W. Xia, Y. Hu, D. Feng, Y. Zhang, Y. Zhou, M. Fu, and L. Gu, "Dec: An efficient deduplication-enhanced compression approach," in *IEEE 22nd International Conference on Parallel and Distributed Systems (ICPADS'16)*, pp. 519–526, 2016.

[4] D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Side channels in cloud services: Deduplication in cloud storage," *IEEE Security & Privacy*, vol. 8, no. 6, pp. 40–47, 2010.

[5] S. Lee and D. Choi, "Privacy-preserving cross-user source-based data deduplication in cloud storage," in *International Conference on ICT Convergence (ICTC'12)*, pp. 329–330, 2012.

[6] C. Li and Z. Liu, "A secure privacy-preserving cloud auditing scheme with data deduplication," *International Journal of Network Security*, vol 21, no. 2, pp. 199–210, 2019.

[7] S. Li, C. Xu, and Y. Zhang, "CSED: Client-side encrypted deduplication scheme based on proofs of ownership for cloud storage," *Journal of Information Security and Applications*, vol. 46, pp. 250–258, 2019.

[8] J. Liu, N. Asokan, and B. Pinkas, "Secure deduplication of encrypted data without additional independent servers," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 874–885, 2015.

[9] D. Meister and A. Brinkmann, "Multi-level comparison of data deduplication in a backup scenario," in *Proceedings of SYSTOR: The Israeli Experimental Systems Conference*, pp. 8, 2009.

[10] D. T. Meyer and W. J. Bolosky, "A study of practical deduplication," *ACM Transactions on Storage (TOS'12)*, vol. 7, no. 4, pp. 14, 2012.

[11] J. Paulo and J. Pereira, "A survey and classification of storage deduplication systems," *ACM Computing Surveys (CSUR'14)*, vol. 47, no. 1, pp. 11, 2014.

[12] Z. Pooranian, K. C. Chen, C. Mu Yu, and M. Conti, "Rare: Defeating side channels based on data-deduplication in cloud storage," in *IEEE INFOCOM IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS'18)*, pp. 444–449, 2018.

[13] P. Puzio, R. Molva, M. Önen, and S. Loureiro, "Cloudedup: Secure deduplication with encrypted data for cloud storage," in *IEEE 5th International Conference on Cloud Computing Technology and Science*, vol. 1, pp. 363–370, 2013.

[14] S. Shan, "An efficient certificateless signcryption scheme without random oracles," *International Journal of Electronics and Information Engineering*, vol. 11, no. 1, pp. 9–15, 2019.

[15] Y. Shin and K. Kim, "Differentially private client-side data deduplication protocol for cloud storage services," *Security and Communication Networks*, vol. 8, no. 12, pp. 2114–2123, 2015.

[16] V. Turner, J. F. Gantz, D. Reinsel, and S. Minton, "The digital universe of opportunities: Rich data and the increasing value of the internet of things," *IDC Analyze the Future*, 2014. (https://www.emc.com/leadership/digital-universe/2014iview/index.htm)

[17] D. Wang, D. He, P. Wang, and C. H. Chu, "Anonymous two-factor authentication in distributed systems: Certain goals are beyond attainment," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 4, pp. 428–442, 2014.

[18] D. Wang and P. Wang, "Two birds with one stone: Two-factor authentication with security beyond conventional bound," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 708–722, 2016.

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

[20] W. Xia, H. Jiang, D. Feng, F. Douglis, P. Shilane, Y. Hua, M. Fu, Y. Zhang, and Y. Zhou, "A comprehensive study of the past, present, and future of data deduplication," *Proceedings of the IEEE*, vol. 104, no. 9, pp. 1681–1710, 2016.

[21] W. Xia, H. Jiang, D. Feng, and Y. Hua, "Silo: A similarity-locality based near-exact deduplication scheme with low ram overhead and high throughput," in *USENIX Annual Technical Conference*, pp. 26–30, 2011.

[22] W. Xia, Y. Zhou, H. Jiang, D. Feng, Y. Hua, Y. Hu, Q. Liu, and Y. Zhang, "Fastcdc: A fast and efficient content-defined chunking approach for data deduplication," in {*USENIX*} *Annual Technical Conference ({USENIX}{ATC} 16)*, pp. 101–114, 2016.

[23] C. M. Yu, S. P. Gochhayat, M. Conti, and C. S. Lu, "Privacy aware data deduplication for side channel in cloud storage," *IEEE Transactions on Cloud Computing*, 2018. (https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8260900)

Biography

Jie Ouyang received the bachelor degree in engineering from Nanchang HangKong University, Nanchang, Jiangxi Province,China in 2014. Now, he is a master student in software engineering at the School of Computer Engineering and Science,Shanghai University. His research interests include Network Security, Cloud computing platform, Data mining and analysis. Email:oyjcoding@shu.edu.cn.

Huiran Zhang received the B.S. degree and the Ph.D. degree from the University of Toyama, Japan. He is currently an Associate Professor with the Shanghai University, Shanghai, China. His research interests include Cloud Computing, Machine Learning, and big data semantic processing.

Hongqing Hu has been learning at Shanghai University for her master's degree from September 2017 to the present.She is focusing on the research of cloud computing and machine learning.

Xiao Wei received the B.S. degree from the Shandong University, China, and the Ph.D. degree from Shanghai University, China, all in computer science. He is currently an Associate Professor with the Shanghai University, Shanghai, China. His research interests include NLP, machine learning, textual semantic analysis, and big data semantic processing.

Dongbo Dai is a lecturer at Shanghai University. His main research interests are computational theory and algorithmic game theory.

Two Lightweight Authenticated Key Agreement Protocols Using Physically Unclonable Function with Privacy Protection

Dan Zhu[1], Liwei Wang[2], and Hongfeng Zhu[2*]

(Corresponding author: Hongfeng Zhu)

School of Foreign Languages, Shenyang Jianzhu University[1]
No.9, HunNan East Street, HunNan District, Shenyang 110168, China
(zhudan413@163.com)
Software College, Shenyang Normal University[2]
No.253, HuangHe Bei Street, HuangGu District, Shenyang 110034, China
(1696751943@qq.com; zhuhongfeng1978@163.com)

(Received July 31, 2019; Revised and Accepted Dec. 3, 2019; First Online Apr. 6, 2020)

Abstract

With the continuous development of the information age, people's demand for information security also increases. In recent years, some people combine passwords with input Physically unclonable function (PUF) to authenticate unsafe communication more effectively. The PUF described in this authentication method has a hardware-based embedding function and has significant physical inconsistency. In this paper, we will continue to propose more effective authentication protocols based on the PUF algorithm. In published articles, authentication involves only one user and one server. Today, we discuss a text authentication protocol involving three parties: a three-party key agreement protocol based on the PUF algorithm. This key agreement protocol has more practical functions. The two users in the protocol and the server have different public keys and their private keys. After a series of calculations and the server's message transmission, the information in the two users' hands has matched again. Based on continuous experiments, we find that this key agreement protocol is effective.

Keywords: Key Exchange; Mutual Authentication; Physically Unclonable Function; Privacy Protection

1 Introduction

In today's society, many people already love browsing on various websites. At the same time, with the development of the network, information security has been involved in people's production and life. At present, the most popular technology is authentication protocol based on some algorithm. Two password-based PUF authentications have been introduced in literature [1, 6]. First, a physical unclonable function (PUF) is physically embedded in a hardware of device and always outputs an unpredictable noise y for an input x depending on unique hardware characteristics. The PUF [5] also has an unclonable property that any attempt to clone or reproduce makes itself unrecoverable from an original one. Owing to its unpredictability and unclonability, in recent years, the PUF has been extensively integrated into devices over wireless communication environment.

From the most practical point of view, human memory password is one of the most common authentication methods in wireless personal communication, because this kind of password is the most convenient and simple authentication tool in practice. But in many cases, passwords in user memory are easily guessed or stolen, and they are inherently vulnerable to well-known online and online dictionary attacks [9]. Because this simple and simple memory password has been attacked and stolen, two factor authentication (smart card and password) have been designed [6]. Recently, due to the emergence of new hardware technologies such as PUF, some scholars have studied the combination of password and PUF [1] in order to conduct more effective authentication [7] in unsafe communications, and hope to prevent the loss of password and personal information in this way. Protecting personal privacy [3]. They are all based on a same assumption that the PUF is initially integrated with a fuzzy extractor (FE) for converting a PUF's unpredictable output into a stable output. It first takes a password input password from a user and outputs an unpredictable secret s through FE. The secret s later serves as a main authentication factor.

In this way, network adversaries can prevent guessing attacks on user's personal passwords. At present, the published literature on key agreement by combining password and PUF involves only a single user and server Two ex-

isting schemes are based on an IUF, for example: PUF + PAKE Scheme, PUF + ZKPK Scheme [6].

The key agreement protocol based on PUF, which has been written into the literature above, has been solved. However, we still need to think about the next issues.

This paper attempts to design a new protocol, which can be established in a more practical environment under the existing PUF algorithm [4]. This protocol breaks the tradition and is no longer a single key agreement between users and servers. It is upgraded to a three-party key agreement protocol based on PUF algorithm [11, 12]. To this end, we will review the key authentication process based on PUF algorithm published in the literature. We will refer to the published literature [5–7]. On the basis of the elaboration, this paper discusses the definition, steps, practical uses, advantages and disadvantages of the three-party key agreement protocol based on PUF, and improvements.

Generally speaking, the purpose of this paper is as follows:

1) We design two key agreement protocols based on PUF algorithm, one is tripartite key agreement protocol, the other is group key agreement protocol. This scheme can support mutual authentication and ensure the security of authentication.

2) The two protocols designed by us can resist off-line password guessing attacks, and have strong practicality and security.

3) The two key agreement protocols designed by us can also protect perspicacious.

The arrangement of this paper is as follows: We will outline the preparatory knowledge in the second section. In the third section, three-party and group key agreement protocols based on PUF algorithm are introduced. Section 4 gives the analysis of security and efficiency. Section 5 gives a summary of this paper.

2 Related Work

Nowadays, there are two types of authentication protocols for PUF algorithm in published literature [1, 2]. One is PUF+PAKE scheme and the other is PUF+ZKPK scheme. The basic idea of these two protocols is to convert unpredictable PUF output into a unified random key that can be used as a key directly on the integrated PUF by using a fuzzy processor [6]. The most important thing is that these two authentication protocols are under the integrated PUF algorithm, and after dealing with noise through the fuzzy processor, they can always get a clearer output on the same input.

2.1 PUF and Password Combination

Firstly, we need to review the process of using PUF and password to generate keys for registration published in the literature. They all execute the registration protocol in the secure channel and then run the authentication protocol. The registration protocol of PUF+PAKE scheme [6] is showed in Figure 1.

The Server S randomly selects a random number c_i and sends it to the user U. Next, the user U calculates $d_i \leftarrow H(c_i \| pwd)$ with his own memory password.User U use input to calculate PUF and get (s_i, w_i) from the formula $Gen(PUF(d_i))$. The registration protocol of PUF+ZKPK scheme [6] is showed in Figure 2.

The Server S randomly selection of a random Value c, and chooses $\{G_q\}$. And send them to users U. U evaluates PUF for an input $H(H(c||pwd, \{G_q\})$ and calculates s, w from $Gen(PUF(H(H(c||ped), \{G_q\}))$ User U computes $u(= g^s mod q)$ and sends (u, w) to Server S, its corresponding list $(c, \{G_q\}, u, w)$ is maintained in DB.

2.2 Framework of Our Scheme

In Figure 3, we further illustrate the steps of our tripartite key agreement protocol. In this protocol, we define two user names: U_1 and U_2, and define the server as S. U_1 and S have an common key k_1, U_2 and S have an common key k_2. U_1 and U_2 choose their temporary private key, $x_1, y_1 = g^{x_1}$,$x_2, y_2 = g^{x_2}$. Then, U_1 computes $E_{k_1}(U_1||y_1)$, $E_{k_1}(U_1||U_2||y_2)$ and U_2 computes $E_{k_2}(U_2||y_2)$, $E_{k_2}(U_2||U_1||y_1)$. U_1 and U_2 uses S to distribute and transmit messages. At last, U_1 and U_2 computes $SK_{u_1u_2} = y_2^{x_1}$, $SK_{u_1u_2} = y_1^{x_2}$. The two formulas are equal or not.

3 The Improved Two-Party PAKA Protocol with Privacy Protection

3.1 Notations

The concrete notations used hereafter are shown in Table 1.

3.2 Authenticated Key Agreement Phase

Figure 4 illustrates the user Authenticated phase. When two users, one The server or three parties conduct key agreement, two users hold the shared key with the server respectively k_1, k_2.These two users and The servers will complete the authentication process on the secure information channel.

Step 1: User U_1, U_2 and The Server Pass Shared Key k_1, k_2 , U_1 randomly selection x_1 to compute private key $y_1 = g^{x_1}$, U_2 randomly selection x_2 to compute private key $y_2 = g^{x_2}$.

Step 2: After receiving the message k_1, k_2 from U_1, U_2, Two users using $E_{k_m}()$ encryption method to compute $E_{k_1}(U_1||y_1)$,$E_{k_2}(U_2||y_2)$. And pass the calculation result to the server. The server will trans-

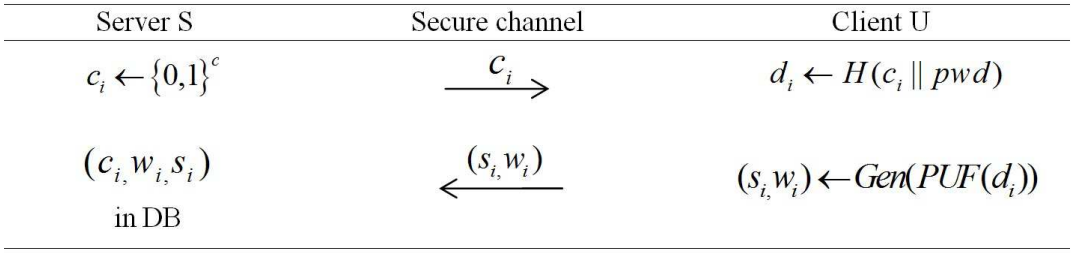


Figure 1: Enrollment phase in PUF + PAKE scheme

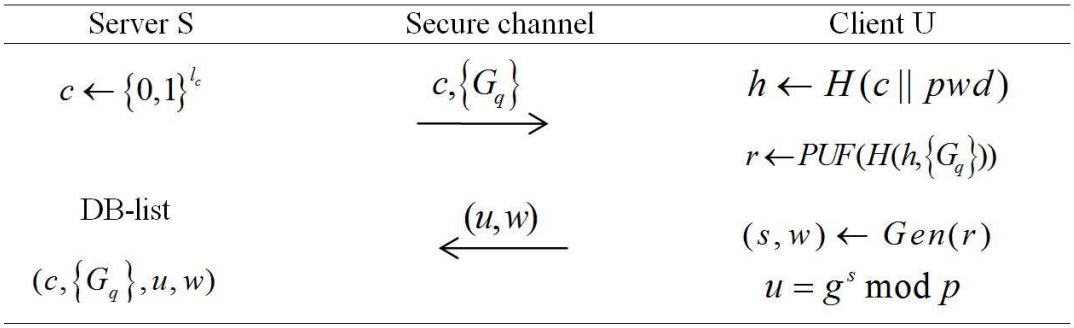


Figure 2: Enrollment phase in PUF + ZKPK scheme

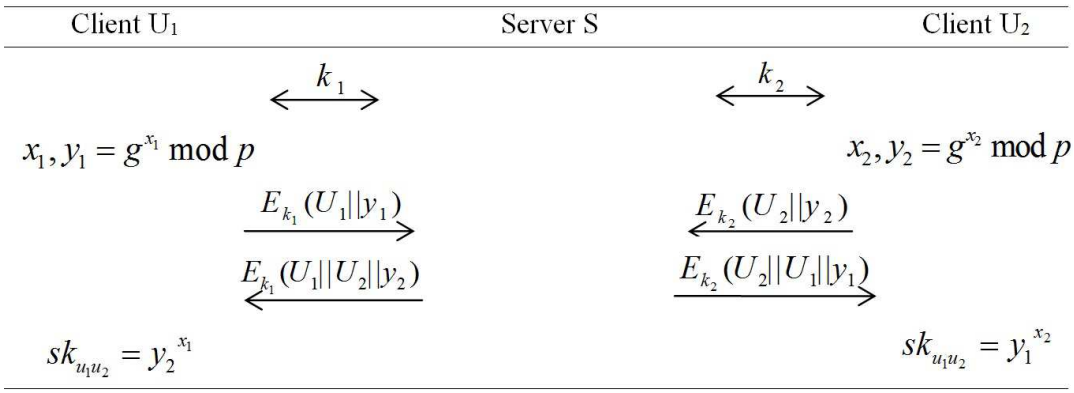


Figure 3: Authenticated key exchange phase of tripartite key agreement protocol

Table 1: Notations

Symbol	Definition
U_1, U_2,U_n	User name
S	Server
K_1,K_2,K_n	Shared Key between User and Server
x_1,x_2,y_1,y_2	private key
$E_{K_m}()$	Use $K_m$to symmetrically encrypt, m is a nonzero integer
$SK_{U_1U_2}$	Session key between U_1 and U_2
SK_{Group}	Group session key
$\|\|$	concatenation operation
$\oplus$	exclusive or operation
$H()$	Hash Functions

mit the results $E_{k_1}(U_1||U_2||y_2)$ and $E_{k_2}(U_2||U_1||y_1)$ to two users.

Step 3: In the above two steps, the server acts as a messaging role, Users decrypt and verify hash functions. They calculate the two results separately with the information they receive. $sK_{u_1u_2} = y_2^{x_1}$, and $sK_{u_1u_2} = y_1^{x_2}$. Then the two results are equal, the authentication is completed. If any authenticated process does not pass, the protocol will be terminated.

3.3 Authentication Phase of Multiparty Key Agreement Protocol

Based on the tripartite key agreement protocol, we continue to propose a group key agreement protocol scheme. The registration process of group key agreement protocol scheme is similar to that of tripartite key agreement protocol, which is based on PUF+ZKPK scheme. Figure 2 illustrates the user registration phase. The flow of group key agreement is illustrated in Figure 5 below.

Step 1: The Server S and User U_i extract K_i using PUF algorithms. Meanwhile, The Server S shares the key K_i with U_1,U_2,U_3 ……,U_n.

Step 2: The server S use shared keys with each user to compute, $T_i = K_1 \oplus K_2 \oplus K_3 \cdots \oplus K_{i-1} \oplus K_{i+1} + Timestamp$,and $MAC = H(U_1||U_2|| \ldots ||U_n||K_1 \oplus K_2 \oplus K_3 \cdots \oplus K_n||Timestamp)$. Then, the server broadcasts MAC and time stamp(K_i keep the same number with Time stamp) to the user remove U_i. And send T_i and MAC to U_i separately.

Step 3: U_i needs to compute $SK_i = K_i \oplus T_i$ (K_i keep the same number with T_i). The Server needs compute MAC_i locally. So next U compare MAC_i and MAC. If the two results are equal, the group session key is that $SK_{Group} = H(SK_i||Timestamp)$.

4 Security and Efficiency Analysis

In this section, we will describe a security model of three-party key agreement scheme and group key agreement scheme based on PUF algorithm. And we will prove that the computation of these two schemes is secure in random oracle and ideal cryptography models[8-10].

4.1 Provable Security of Tripartite Key Agreement Protocol

We use the following security model [8] to define the security requirements of authentication schemes for tripartite and group key agreement protocols.

Players. We define a server S and a user U who can participate in the authentication scheme certification of the key agreement protocol. Each of them has different instances. We call them oracles.

Queries. Adversary A can interact with participants and try to break Key or authentication of a user or server. For this purpose, we can use multiple queries.

1) Execute(U,S): This query simulates a passive attack in which the Adversary A will eavesdrop on the authentication communication process between the user U and the server S.

2) Reveal(I): This query simulates the abuse of session keys between instances I. Queries are available only if the attacked the instance I actually holds the session key and releases it.

3) Send(I,m): This query adversary A models the message sent to the instance I. Adversary A receives the response generated when the message m is processed according to the agreement p. In our scheme, adversary emphA query sending (S, start) initializes the key exchange algorithm, so adversary A receiving server should send the stream to the client.

4.2 Security Proof

The following theorem shows that the proposed scheme can safely distribute session keys under the assumption that it is reasonable and well-defined and more difficult to handle [9].

Theorem 1. *Let P be the above agreement and password be a finite dictionary of size N equipped with a uniform distribution. Let A be an adversary against the AKE security of P within a time bound t, with less than q_s interactions with the parties and q_p passive eavesdropping, and asking q_h hash-queries and qe encryption/decryption-queries. Then, we have* $\mathrm{Adv}_p^{ake}(A) \leqslant \frac{q_s}{N} + 4q_h \mathrm{Succ}_G^{cdh}(t') + \frac{(q_s+q_p)^2}{q-1} + \frac{(q_h+4q_s+q_p+q_e)^2}{2^\ell}$ *where* $t' \leqslant t + (q_s + q_p + 1) \cdot \Gamma_G$ *withℓ denoting $\ell_1, \ell_2, \ell_3, \ell_4$ and Γ_G denoting the computational time for an exponentiation in* G.

Proof. **Stage:** In this proof, for simplicity, we do not consider forward secrecy. We incrementally define sequence of games starting at the real game G_0 and G_1, G_2. For each G_n (n = 0, 1, 2) we define the following events:

1) S_n occurs if A correctly guesses the bit b involved in the *Test-query*.

2) $Encrypt_n$ occurs if A submits data it has encrypted by itself using the password.

3) $Auth_n$ occurs if A submits an authenticator $Auth$ that is accepted by the server and that has been built by the adversary itself.

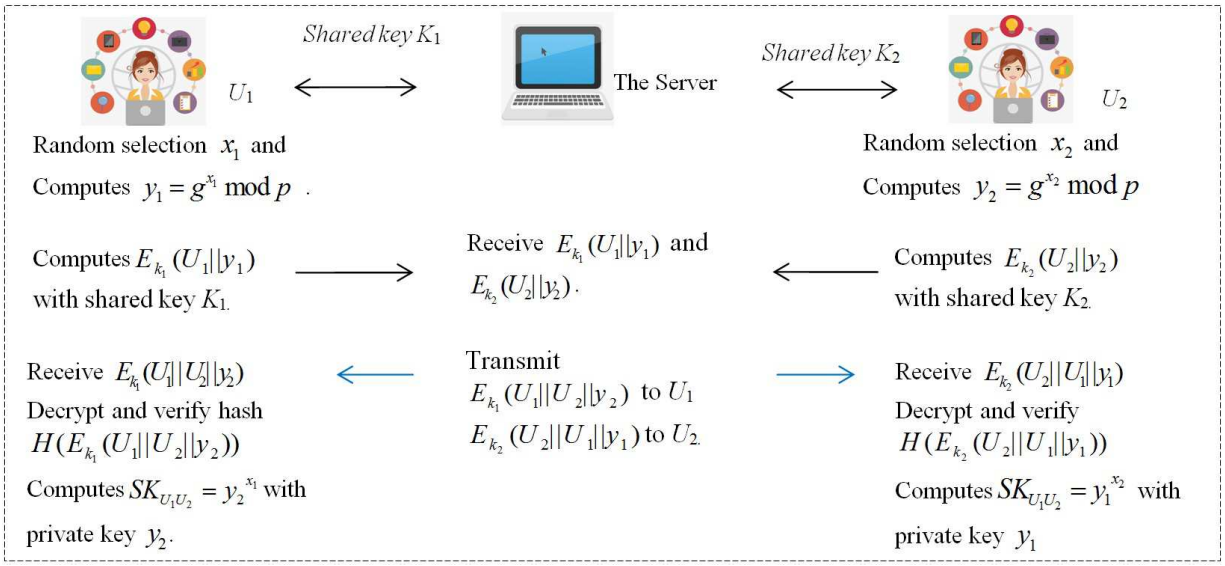


Figure 4: Authenticated key agreement phase

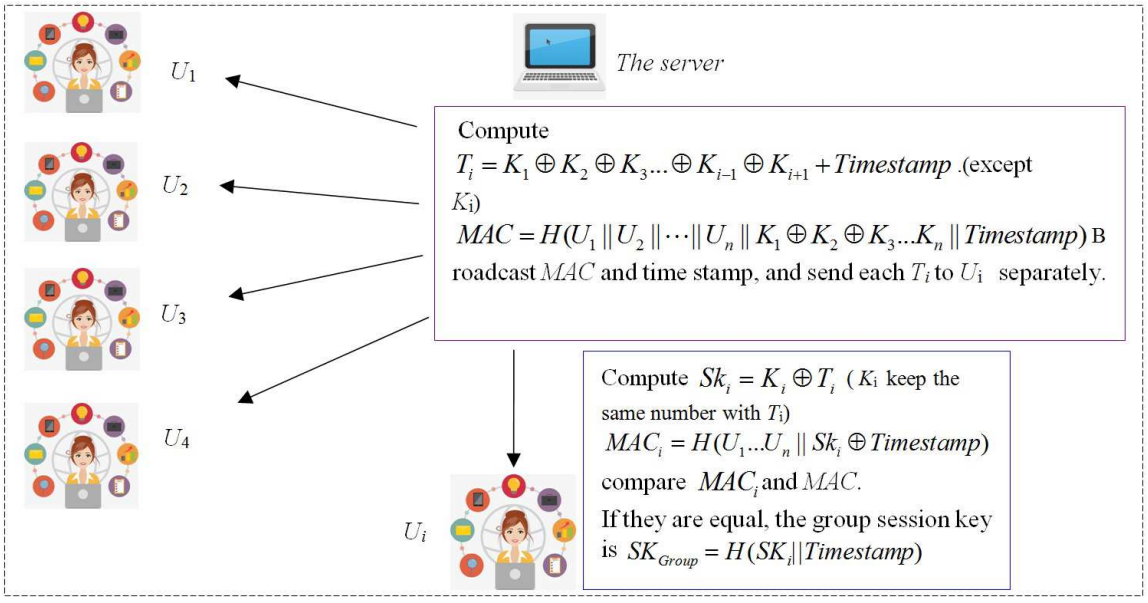


Figure 5: Authentication phase of multiparty key agreement protocol

Game. G_0 : This is the real agreement in the random oracle and ideal-cipher models. Several oracles are thus available to the adversary A: one hash oracles ($H()$) and all the instances U and S (in order to cover concurrent executions). We have

$$\mathrm{Adv}_p^{ake}(A) = 2P_r[S_0] - 1.$$

In the game below, we further assume that when the game aborts or stops with no answer b' outputted by adversary A we choose this bit b' at random, which in turn defines the actual value of the event S_K, Moreover, if the adversary A has not finished playing the game after q_s Send-queries or lasts for more than time t, we stop the game (and choose a random bit b'), where q_s and t are predetermined upper bounds.

Game. G_1 we simulate the hash oracles, $H()$, and five additional hash functions $H()$ and the encryption/decryption oracles and an encryption list.We also simulate all the instances, as the real players would, for the Send-queries and for the *Execute*, *Reveal* and *Test-queries*. From this simulation, it is clear that the game is perfectly. Thus, we have

$$P_r[S_1] = P_r[S_0].$$

Game. G_2 :In this game, the opponent guesses the session key without asking the corresponding Oracle h, so that it exists separately from the password and the temporary key, which are protected by the PUF algorithm. Otherwise, we will use the early game change method. Thus, we have

$$|P_r[S_1] - P_r[S_2]| \leqslant \frac{q_h^2}{2^{\ell+1}} + \frac{q_\varepsilon}{2^{\ell_4+1}} + \frac{(q_s+q_p)^2}{2(q-1)}$$

□

4.3 Further Security Discussion

1) The scheme could resist password guessing attack.

Proof. This attack means that adversary A will try to guess the password of the legitimate user based on the transmitted information. Password guessing attacks can only crack functions with a low-entropy variable (password), so we need to insert at least one large random variable that can withstand such attacks. In our protocol, when there is no transmission information, adversary A can only launch an online password guessing attack, using the password as the input value. Even if the opponent gets secret information, he does not have any comparative data to verify whether the password guess is correct without the help of the server. In other words, an adversary will not be able to construct tables. On the other hand, the maximum number of permissible invalid attempts for online password guessing attacks is only a few, and the account will be locked by the registered server [9, 10]. □

2) The scheme could support mutual authentication.

Proof. The Registration Server S verifies the authenticity of user U's request through validating the condition $MAC_i = MAC$ during the proposed phase. To compute $MAC_i = H(U_1 \ldots U_n || SK_i \oplus Timestamp)$, the attacker must has the password. Furthermore, MAC_i includes a large random nubmer *Timestamp*, the adversary cannot replay the old messages in the protocol. So, mutual authentication can successfully achieve in our scheme. □

3) The scheme could resist replay attack.

Proof. Validation messages include temporary random numbers, such as timestamps. More importantly, all such temporary random numbers are protected by corresponding information. Only legitimate users with secret keys and passwords can find these problems. □

4) The user-privacy protection can be provided in the proposed scheme.

Proof. There is no clear text in the authentication message sent in our proposed protocol. But authentication messages include overwritten ciphertext, which can send any important information to the other party or to a designated place using the public key of the peer, such as the identity in the scheme. Another message is to verify ciphertext using a one-way secure hash function. The other message is transmitted dynamically through channels and cannot be cloned. In addition, there is no duplicate message section in continuous communication. This shows that our scheme implements the attributes of user privacy. □

4.4 Efficiency Analysis

Table 2 shows some basic calculation processes of the three-party key agreement protocol and the group key agreement protocol based on the PUF algorithm written in this paper. In addition, compared with the two-key agreement protocol based on the PUF algorithm, this new protocol is more powerful and more computationally intensive. It is safer and more reliable in the safe transmission of information. Among them, $Ours_1$: three-party key agreement protocol based on PUF algorithm, $Ours_2$: PUF algorithm based group key agreement protocol. Yes/No: Support/Not support , T_{hash}: Time for executing the hash function.

5 Conclusion

This paper presents a tripartite and group key agreement protocol based on PUF algorithm. It extends on the basis of two-party key agreement protocol. This protocol not only inherits the advantages of the two-party key

Table 2: Comparisons between our proposed scheme and the related literatures

	PUF+PAKE [7]	PUF+ZKPK [7]	$Ours_1$	$Ours_2$
Shared secret key	No	No	Yes	Yes
Communication round	3round	2round	2round	1Broadcast
Vulnerable to replay attack	No	No	Yes	Yes
Formal security proof	No	No	Yes	Yes
Privacy protection	No	No	Yes	Yes
Authentication	Mutual	Mutual	Mutual	Mutual

agreement protocol, but also innovates. In addition to being able to authenticate between users and servers, the two-party key agreement protocol can also resist guessing attacks. This new protocol has the characteristics of resisting off-line password attack, supporting mutual authentication and providing privacy protection for users. The protocol also has strong security and enforce-ability, and enriches the types of key agreement protocols.

Acknowledgments

This work was supported by the Liaoning Provincial Natural Science Foundation of China (Grant No. 2019-MS-286), and Shenyang Science & Technology Innovation Talents Program for Young and Middle-aged Scientists (2019).

References

[1] D. Amelino, M. Barbareschi, E. Battista, A. Mazzeo, "How to manage keys and reconfiguration in WSNs exploiting sram based PUFs," in *Intelligent Interactive Multimedia Systems and Services*, pp. 109-119, 2016.

[2] F. Afghah, B. Cambou, M. Abedini, S. Zeadally, "A reram physically unclonable function (ReRAM PUF)-based approach to enhance authentication security in software defined wireless networks," *International Journal of Wireless Information Networks*, 2018. DOI: 10.1007/s10776-018-0391-6.

[3] M. Barbareschi, "Notions on silicon physically unclonable functions," in *Hardware Security and Trust*, pp. 189-209, 2017.

[4] J. W. Byun, I. R. Jeong, "Comments on physically unclonable function based two-factor authentication protocols," *Wireless Personal Communications*, vol. 106, no. 3, pp. 1243-1252, 2019.

[5] N. Chikouche, P. L. Cayrel, E. M. Mboup, *et al.*, J. Supercomput, "A privacy-preserving code-based authentication protocol for internet of things," *The Journal of Supercomputing*, vol. 75, no. 12, pp. 8231-8261, 2019.

[6] J. Delvaux, D. Gu, R. Peeters, and I. Verbauwhede, "A survey on lightweight entity authentication with strong PUFs," *ACM Computing Surveys (CSUR'15)*, vol. 48, no. 2, 2015.

[7] B. Halak, "Physically unclonable functions," *Electronics & Electrical Engineering*, 2018. ISBN 978-3-319-76804-5.

[8] P. Mall, M. Z. A. Bhuiyan, R. Amin, "A lightweight secure communication protocol for IoT devices using physically unclonable function," in *International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage*, vol. 11611, pp. 26-35, 2019.

[9] D. I. Moon, A. Rukhin, R. P. Gandhiraman, B. Kim, M. Meyyappan, "Physically unclonable function by an all-printed carbon nanotube network," *ACS Applied Electronic Materials*, 2019. (`https://doi.org/10.1021/acsaelm.9b00166`)

[10] A. C. D. Resende, K. Mochetti, D. F. Aranha, "PUF-based mutual multifactor entity and transaction authentication for secure banking," in *Lightweight Cryptography for Security and Privacy*, vol. 9542, pp. 77 -96, 2015.

[11] D. P. Sahoo, "Design and analysis of secure physically unclonable function compositions," *Design and Analysis of Secure Physically Unclonable Function Compositions*, 2017. (`http://www.idr.iitkgp.ac.in/xmlui/handle/123456789/8243`)

[12] D. P. Sahoo, A. Bag, S. Patranabis, D. Mukhopadhyay, R. S. Chakraborty, "Fault-tolerant implementations of physically unclonable functions on FPGA," in *Security and Fault Tolerance in Internet of Things*, pp. 129-153, 2019.

Biography

Dan Zhu obtained her master degree in English Language and Literature from Liaoning Nornal University. Dan Zhu is a teacher at Shenyang Jianzhu University. She has research interests in Big Data, e-learning, and translation. Mrs. Zhu has published more than 10 international journal papers on the above research field.

Liwei Wang a postgraduate studying at Shenyang Normal University. She has researched interests in network security and quantum cryptography. Under the guidance of the teacher, she has published one article in

EI journals.

Hongfeng Zhu obtained his Ph.D. degree in Information Science and Engineering from Northeastern University. Hongfeng Zhu is a full professor of the Kexin software college at Shenyang Normal University. He is also a master's supervisor. He has research interests in wireless networks, social networks, network security and quantum cryptography. Dr. Zhu had published more than 60 international journal and international conference papers on the above research fields.

Security Bound of Biclique Attacks on AES-128

Xiaoli Dong[1] and Jie Chen[2]
(Corresponding author: Xiaoli Dong)
School of Cyberspace Security, Xi'an University of Posts and Telecommunications[1]
Xi'an 710121, China
School of Telecommunication Engineering, Xidian University[2]
Xi'an 710071, China
(Email: dxl_xaut@163.com)
(Received Aug. 24, 2019; Revised and Accepted Dec. 6, 2019; First Online Feb. 3, 2020)

Abstract

For two future possible improvements of AES-128: enhanced subkey diffusion property or increased encryption rounds, this paper evaluates the security bound of R-Round AES-128 ($R > 10$) and 10-Round AES-IND-128 (AES-128 with independent of key schedule) against biclique attack. For attacking R-round AES-128($R > 10$), with the increase of several rounds R, the time complexity increases gradually, but it never reaches $2^{127.86}$, reduced by about 10% compared with brute force. For attacking 10-Round AES-IND-128, a 1-round biclique is firstly constructed, and then the attack is proposed. The time complexity is no more than $2^{127.42}$, reduced by about 33% compared with brute force.

Keywords: AES; Biclique; Block Cipher; Cryptanalysis

1 Introduction

Block ciphers are the central tool in the design of protocols for symmetric encryption [16]. Since Rijndael [4] is the winner of Advanced Encryption Standard (AES) in 2000, it has become one of the most widely used Block Ciphers. AES supports 128-bit block size with three different key lengths of 128, 192 and 256 bits, which are denoted as AES-128, AES-192 and AES-256 respectively.

Due to the popularity of AES, cryptology researchers have increased their focus on its security, such as square attack [5, 10], collision attack [12],impossible differential attack [14, 15], meet-in-the-middle attack [6–9], biclique attack [3], integral attacks [18],polytopic attacks [19], subspace trail cryptanalysis [11], structural-differential attacks [13], yoyo tricks [17],Grassi's Attack [1] and so on.

There are many cryptanalytic results on AES-128. In 1997, the 6-round AES-128 was broken with the square attack by the designer of AES [5]. Since 2000, 7-round AES-128 has been broken successively by a series of attacks as follows. The square attack in [10] requires a data complexity of $2^{127.997}$ and a time complexity of 2^{120}. The collision attack in [12]costs a data complexity of 2^{32} and a time complexity of 2^{128}.The impossible differential attacks were proposed in [14,15] and the fastest needs a time complexity of $2^{117.2}$. The meet-in-the-middle attack was proposed in [7–9] and the fastest needs a time complexity of 2^{99}. In 2011, 10 rounds were successfully broken with a data complexity 2^{88} and a time complexity of $2^{126.18}$ when the biclique attack [3] is applied to AES-128. In 2014, the biclique attack [2] was once again improved with a data complexity of 2^{64} and a time complexity of $2^{126.12}$.

The biclique attack [3] can be divided into two steps: biclique exploration from the independent related key differentials, and then key recovery. In the key recovery phase of the attack, the recomputation technique is usually adopted to reduce the time complexity. There are two parts separate components of recomputation: State recomputation and subkey recomputation. In [3], subkey recomputation is not considered because it is negligible compared to state recomputation. This paper considers both the two parts of recomputation: State recomputation and subkey recomputation, this is because the subkey recomputation cannot be ignored for R-Round AES-128($R > 10$) and 10-Round AES-IND-128 anymore.

The principle of a biclique attack is to test all the keys to discover which is the correct key based on the biclique structure. Therefore, the complexity of the biclique attack on the full rounds of any block cipher will never exceed the complexity of the exhaustive key search. If the encryption rounds are increased for AES-128, or if the diffusion of the key schedule is enhanced, how will the complexity of the biclique attack change?

In this paper,we study the biclique attack for R-Round AES-128($R > 10$) and 10-Round AES-IND-128, where AES-IND-128 denotes AES-128 independent of the key schedule.

1) For attacking R-round AES-128 ($R > 10$),with the increasing number of rounds R, the time complexity increases gradually, yet it will never reach $2^{127.86}$. In the attack,both state and subkey recomputation are consided.Prior to performing the attack, properties of the recomputation of R-round subkeys are discov-

ered.

2) For attacking 10-round AES-IND-128,the time complexity is $2^{127.42}$. AES-IND-128 includes the property: the diffusion property is enhanced so that any subkey byte in any round affects all the subkey bytes in the other rounds in term of the difference. When the diffusion of key schedule is enhanced or the number of encryption rounds is increased, biclique attacks remain effective to AES-128.

The remainder of this paper is organized as follows. The AES block cipher is described in Section 2 and the biclique attack is introduced in Section 3. In Sections 4 and 5, we apply the biclique attacks to R-round AES-128 and 10-round AES-IND-128, respectively. Finally in Section 6, the conclusions are drawn from the results.

2 Description of AES-128

AES-128 [4] encrypts or decrypts data blocks of 128 bits by using keys of 128-bits. The number of rounds is 10, denoted as successively. A 128-bit plaintext and the interme-diate state are treated as byte matrices of size. The round transformation of AES consists of the four basic transformations:

SubBytes (SB): Applying the same 8-bit to 8-bit invertible S-box 16 times in parallel on each byte of the state.

ShiftRows(SR): Cyclically shifting each row (the i'th row is shifted by i bytes to the left).

MixColumns (MC): Multiplication of each column by a 4×4 matrix over the field $GF(2^8)$.

AddRoundKey (ARK): XORing the state and a subkey.

The MC operation is omitted in the last round, and an additional ARK operation using a Whitening SubKey (WSK) is performed before the first round.

The Key Schedule(KS) of AES-128 takes the 128-bit user key and transforms it into 11 subkeys of 128-bits each.The subkey array is denoted by $W_0, W_1, \cdots, W_{43}$, where each word of consists of 32 bits. $K = (W_0, W_1, W_2, W_3)$ is the user supplied keys. $K_r = (W_{4(r+1)}, W_{4(r+1)+1}, W_{4(r+1)+2}, W_{4(r+1)+3})(r = 0, \cdots, 9)$ is updated according to the following rule:
If

$$i \equiv 0 \bmod 4$$

then

$$W_i \equiv W_{i-4} \oplus SB(RotByte(W_{i-1})) \oplus Rcon(i/4)$$

else

$$W_i \equiv W_{i-4} \oplus W_{i-1}.$$

K_r denotes the r-round subkey. $K_{-1} = K$ as a WSK. $RotByte(\cdot)$ denotes the rotation of the word by one byte.

In this paper, $\#2r$ and $\#(2r+1)$ are addressed as the states before SB and the state after MC in round r, respectively.

Property 1. *For r-round AES-128, the ratio of one SubBytes operation to the full AES is $\sigma = 4/5r$.*

3 Chosen-Ciphertext Biclique Attack

In this section, we introduce the biclique attack proposed by Bogdanov *et al.* [3]. Before the description of the biclique attack, the biclique structure must be denoted: Let f be subcipher that maps an internal state S to the ciphertext $C : f_K(S) = C$.The 3-tuple $[\{C_i\}, \{S_j\}, \{K[i,j]\}]$ is called a d-dimensional biclique, if $C_i = f_{K[i,j]}(S_j)$ $(, i, j \in \{0, 1, \cdots, 2^d - 1\})$.

The biclique attack can be divided into two steps: biclique exploration and key recovery. The following sections of this paper will describe both biclique exploration and key recovery in further detail.

3.1 Biclique Exploration from the Independent Related-Key Differentials

The d-dimensional biclique can be achieved from the independent related-key differentials. There are two stages: key partition and then biclique construction.

3.1.1 Key Partition

2^n keys can be divided into 2^{n-2d} groups $K^{(m)}(m = 0, 1, \cdots, 2^{n-2d} - 1)$, where $K^{(m)}$ is defined to be the m-th key group with 2^{2d} keys $K^{(m)}[i,j](i, j = (0, 1, \cdots, 2^d - 1))$. These keys can be obtained as follows:

- Look for the key differentials Δ_i^K, ∇_j^K, such that $\Delta_i^K \cap \nabla_j^K = \{0\}$.
- With Δ_i^K, ∇_j^K, determine the base key $K^{(m)}[0,0]$ in $K^{(m)}$.

Therefore: $K^{(m)}[i,j] = K^{(m)}[0,0] \oplus \Delta_i^K \oplus \nabla_j^K$.

3.1.2 Biclique Construction

In each $K^{(m)}$, construct the d-dimension biclique as follows:

- Base computation: $S_0^{(m)} \xrightarrow[f]{K^{(m)}[0,0]} C_0^{(m)}$.
- Based on Δ_i^K, ∇_j^K, construct differentials

$$\Delta_i^{(m)} - differentials : 0 \xrightarrow[f]{\Delta_i^K} \Delta_i^{(m)},$$

$$\nabla_j^{(m)} - differentials : \nabla_j^{(m)} \xrightarrow[f]{\nabla_j^K} 0.$$

Such that they do not share the active nonlinear component, where $\Delta_0^K = 0, \Delta_0^{(m)} = 0, \nabla_0^K = 0, \nabla_0^{(m)} = 0$.

Therefore it is denoted:

$$\begin{aligned} S_j^{(m)} &= S_0^{(m)} \oplus \nabla_j^{(m)} \\ C_i^{(m)} &= C_0^{(m)} \oplus \Delta_i^{(m)} \\ K^{(m)}[i,j] &= K^{(m)}[0,0] \oplus \Delta_i^K \oplus \nabla_j^K \end{aligned}$$

and obtain the definition of a d-dimensional biclique $[\{C_i^{(m)}\}, \{S_j^{(m)}\}, \{K^{(m)}[i,j]\}]$.

3.2 Key Recovery under the Chosen-Ciphertext Attack

3.2.1 Key Recovery

For each $K^{(m)}$, based on the d-dimension biclique $[\{C_i^{(m)}\}, \{S_j^{(m)}\}, \{K^{(m)}[i,j]\}]$, the key recovery is as follows:

- Data Collection: The adversary obtains the 2^d plaintexts from the ciphertexts $P_i^{(m)}$ through the decryption oracle: $C_i^{(m)} \xrightarrow[e^{-1}]{decryption \quad oracle} P_i^{(m)}$.
- Key Testing: The block cipher E can be decomposed into $E : P \underset{h}{\rightarrow} V \underset{g}{\rightarrow} S \underset{f}{\rightarrow} C$.For the testing key $K^{(m)}[i,j]$, the adversary checks whether

  $$P_i^{(m)} \xrightarrow[g]{K^{(m)}[i,j]} \vec{v}^{(m)} \stackrel{?}{=} \overleftarrow{v}^{(m)} \xleftarrow[h]{K^{(m)}[i,j]} S_j^{(m)}$$

  If the equation holds, the testing key is the secret key K_{secret}.

The full time complexity of the attack is:

$$C_{full} = 2^{n-2d} C^{(m)} = 2^{n-2d}[C_{biclique}^{(m)} + C_{match}^{(m)} + C_{falsepos}^{(m)}]$$

where $C_{biclique}^{(m)}$ is the complexity of constructing biclique; $C_{match}^{(m)}$ is the complexity of the computation of the internal variable v 2^d times in each direction; $C_{falsepos}^{(m)}$ is the complexity generated by false positives.

3.2.2 The Recomputation Technique

In fact, in order to decrease the time complexity in the key testing, adopt the recomputation technique. In fact, there are two parts of recomputation as follows.

- State Recomputation.
  - Precomputation: The adversary computes and stores 2×2^d computations: $\forall\ i(P_i^{(m)} \xrightarrow[g]{K^{(m)}[i,0]} \vec{v}^{(m)})$ and $\vec{v}^{(m)} \xleftarrow[h]{K^{(m)}[0,j]} S_j^{(m)}$.
  - Recomputation: For particular i and j, the adversary recomputes the states which differ from the stored ones.
- Subkey Recomputation. First, the adversary computes and stores computations $K^{(m)}[i,0]$ and $K^{(m)}[0,j]$. Then, for other $K^{(m)}[i,j]$, it is recomputed only those parts that differ from the stored ones.

  So the full time complexity of the attack is: $C_{full} = 2^{l-2d} C^{(m)} = 2^{l-2d}[C_{biclique}^{(m)} + C_{precomp}^{(m)} + C_{recomp_1}^{(m)} + C_{recomp_2}^{(m)} + C_{falsepos}^{(m)}]$, where $C_{precomp}^{(m)}$ is the complexity of the precomputation in the key recovery; $C_{recomp_1}^{(m)}$ and $C_{recomp_2}^{(m)}$ are the complexities of the state and the subkey recomputation in the matching stage,respectively.

Remark 1. *Subkey recomputation is ignored in [3],but cannot be ignored on attacks on R-Round AES-128 ($R > 10$) and 10-Round AES-128-IND.*

4 Biclique Attack on R-Round AES-128($R > 10$)

The encryption rounds of AES-128 can be improved to exceed 10 rounds, therefore biclique attacks are investigated for R-Round AES-128($R > 10$). For the purpose of this research a recomputation technique was adopted within section III. With the rounds increased, $C_{recomp_2}^{(m)}$ becomes increasingly very critical and complex, so we firstly calculate the recomputation of the subkeys.Then perform the biclique attack on $R-$Round AES-128($R > 10$).

4.1 Recomputation of the Subkeys

The superscript (m) of $K_{(r)}^{(m)}[i,j]$ is not reflected in the following lemmas, so it is omitted and denoted as $K_{(r)}[i,j]$.

If we know r-round subkeys $K_{(r)}[0,0], \Delta_i^K, \nabla_j^K$ and $K_{(r)}[i,j] = K_{(r)}[0,0] \oplus \Delta_i^K \oplus \nabla_j^K$, evaluate recomputation of the $(r-1)$-round subkeys $K_{(r-1)}[i,j]$ by the following lemmas.

Lemma 1. *If the value of r-round subkeys $K_{(r)}[i,j]$ is known, the evaluatation of the recalculation of the $(r-1)$-round subkeys $K_{(r-1)}[i,j]$ is given as follows.*

1) $K_{(r-1)}$ can be expressed as the linear function of $K_{(r)}$ and $\Delta K_{(r)}$, so the recomputation for $K_{(r-1)}$ can be ignored, where $\Delta K_{(r)} = K_{(r)}[i,j] \oplus K_{(r)}[0,0]$.

2) $K_{(r-1)}$ cannot be expressed as the linear function of $K_{(r)}$ and $\Delta K_{(r)}$, so the recomputation for $K_{(r-1)}$ cannot be ignored.

Proof. $K_{(r),h}(h = 0, 1, \cdots, 15)$ denote h-th byte of $K_{(r)}$. If the subkeys in round r are known, the subkeys in round

$r-1$ can be achieved as follows by the key schedule.

$$\begin{aligned}
K_{(r-1),0} &= K_{(r),0} \oplus S(K_{(r),9} \oplus K_{(r),13}) \\
K_{(r-1),1} &= K_{(r),1} \oplus S(K_{(r),10} \oplus K_{(r),14}) \\
K_{(r-1),2} &= K_{(r),2} \oplus S(K_{(r),11} \oplus K_{(r),15}) \\
K_{(r-1),3} &= K_{(r),3} \oplus S(K_{(r),8} \oplus K_{(r),12}) \\
K_{(r-1),4} &= K_{(r),0} \oplus K_{(r),4} \\
K_{(r-1),5} &= K_{(r),1} \oplus K_{(r),5} \\
K_{(r-1),6} &= K_{(r),2} \oplus K_{(r),6} \\
K_{(r-1),7} &= K_{(r),3} \oplus K_{(r),7} \\
K_{(r-1),8} &= K_{(r),4} \oplus K_{(r),8} \\
K_{(r-1),9} &= K_{(r),5} \oplus K_{(r),9} \\
K_{(r-1),10} &= K_{(r),6} \oplus K_{(r),10} \\
K_{(r-1),11} &= K_{(r),7} \oplus K_{(r),11} \\
K_{(r-1),12} &= K_{(r),8} \oplus K_{(r),12} \\
K_{(r-1),13} &= K_{(r),9} \oplus K_{(r),13} \\
K_{(r-1),14} &= K_{(r),10} \oplus K_{(r),14} \\
K_{(r-1),15} &= K_{(r),11} \oplus K_{(r),15}.
\end{aligned}$$

The equivalent but concise expressions are as follows:

$$\begin{aligned}
K_{(r-1),h} &= K_{(r),h} \oplus S(K_{(r),8+(h+1) \bmod 4} \\
&\quad \oplus K_{(r),12+(h+1) \bmod 4}) \ (h = 0, 1, 2, 3) \\
K_{(r-1),h} &= K_{(r),h-4} \oplus K_{(r),h} \ (h = 4, 5, \cdots, 15).
\end{aligned}$$

Remark 2. *For clarity, Rcon(•) is ignored in the key schedule above as it does not affect the final results.*

If the values of r-round subkeys are known: $K_{(r)}[i,j]$, $\Delta K_{(r)} = K_{(r)}[i,j] \oplus K_{(r)}[0,0]$. There are 2 cases in the recalculation of the $r-1$ round subkeys $K_{(r-1)}[i,j]$.

Case 1: For $0 \le h \le 3$, we consider the computation of $K_{(r-1),0}, K_{(r-1),1}, K_{(r-1),2}, K_{(r-1),3}$. As $\Delta K_{(r)} = K_{(r)}[i,j] \oplus K_{(r)}[0,0]$, we have

$$\begin{aligned}
K_{(r),0}[i,j] &= K_{(r),0}[0,0] \oplus \Delta K_{(r),0} \quad (1) \\
K_{(r),8+(h+1) \bmod 4}[i,j] &= K_{(r),8+(h+1) \bmod 4}[0,0] \\
&\quad \oplus \Delta K_{(r),8+(h+1) \bmod 4} \quad (2) \\
K_{(r),12+(h+1) \bmod 4}[i,j] &= K_{(r),12+(h+1) \bmod 4}[0,0] \\
&\quad \oplus \Delta K_{(r),12+(h+1) \bmod 4} \quad (3) \\
K_{(r-1),h}[0,0] &= K_{(r),h}[0,0] \quad (4) \\
&\quad \oplus S(K_{(r),8+(h+1) \bmod 4}[0,0] \\
&\quad \oplus K_{(r),12+(h+1) \bmod 4}[0,0]) \\
K_{(r-1),h}[i,j] &= K_{(r),h}[i,j] \quad (5) \\
&\quad \oplus S(K_{(r),8+(h+1) \bmod 4}[i,j] \\
&\quad \oplus K_{(r),12+(h+1) \bmod 4}[i,j]).
\end{aligned}$$

By Equations (1)-(5), we have

$$\begin{aligned}
K_{(r-1),h}[i,j] &\overset{(5)}{=} K_{(r),h}[i,j] \\
&\quad \oplus S(K_{(r),8+(h+1) \bmod 4}[i,j] \\
&\quad \oplus K_{(r),12+(h+1) \bmod 4}[i,j]) \\
&\overset{(2)(3)}{=} K_{(r),h}[i,j] \oplus S(K_{(r),8+(h+1) \bmod 4}[0,0] \\
&\quad \oplus \Delta K_{(r),8+(h+1) \bmod 4} \\
&\quad \oplus K_{(r),12+(h+1) \bmod 4}[0,0] \\
&\quad \oplus \Delta K_{(r),12+(h+1) \bmod 4}) \\
&\overset{(1)}{=} K_{(r),h}[0,0] \oplus \Delta K_{(r),h} \\
&\quad \oplus S(K_{(r),8+(h+1) \bmod 4}[0,0] \\
&\quad \oplus \Delta K_{(r),8+(h+1) \bmod 4} \\
&\quad \oplus K_{(r),12+(h+1) \bmod 4}[0,0] \\
&\quad \oplus \Delta K_{(r),12+(h+1) \bmod 4}) \\
&\overset{(4)}{=} K_{(r-1),h}[0,0] \oplus \Delta K_{(r),h} \\
&\quad \oplus S(K_{(r),8+(h+1) \bmod 4}[0,0] \\
&\quad \oplus K_{(r),12+(h+1) \bmod 4}[0,0]) \\
&\quad \oplus S(K_{(r),8+(h+1) \bmod 4}[0,0] \\
&\quad \oplus \Delta K_{(r),8+(h+1) \bmod 4} \\
&\quad \oplus K_{(r),12+(h+1) \bmod 4}[0,0] \\
&\quad \oplus \Delta K_{(r),12+(h+1) \bmod 4}).
\end{aligned}$$

Therefore,

$$\begin{aligned}
K_{(r-1),h}[i,j] &= K_{(r-1),h}[0,0] \oplus \Delta K_{(r),h} \\
&\quad \oplus S(K_{(r),8+(h+1) \bmod 4}[0,0] \\
&\quad \oplus K_{(r),12+(h+1) \bmod 4}[0,0]) \\
&\quad \oplus S(K_{(r),8+(h+1) \bmod 4}[0,0] \\
&\quad \oplus \Delta K_{(r),8+(h+1) \bmod 4} \\
&\quad \oplus K_{(r),12+(h+1) \bmod 4}[0,0] \\
&\quad \oplus \Delta K_{(r),12+(h+1) \bmod 4}). \quad (6)
\end{aligned}$$

Therefore, there are 2 Conditions 1), 2) from Equation (6).

1) $\Delta K_{(r),8+(h+1) \bmod 4} \oplus \Delta K_{(r),12+(h+1) \bmod 4} = 0 \Rightarrow K_{(r-1),h}[i,j] = K_{(r-1),h}[0,0] \oplus \Delta K_{(r),h}$. $K_{(r-1)}[i,j]$ can be expressed as the linear function of $K_{(r-1)}[0,0]$ and $\Delta K_{(r)}$.
2) $\Delta K_{(r),8+(h+1) \bmod 4} \oplus \Delta K_{(r),12+(h+1) \bmod 4} \neq 0 \Rightarrow K_{(r-1),h}[i,j] \neq K_{(r-1),h}[0,0] \oplus \Delta K_{(r),h}$. $K_{(r-1)}[i,j]$ cannot be expressed as the linear function of $K_{(r-1)}[0,0]$ and $\Delta K_{(r)}$.

Case 2: For $4 \le h \le 15$,we consider the computation of $K_{(r-1),4}, K_{(r-1),5}, \cdots K_{(r-1),15}$. Because $\Delta K_{(r)} = K_{(r)}[i,j] \oplus K_{(r)}[0,0]$, we have

$$\begin{aligned}
K_{(r),h-4}[i,j] &= K_{(r),h-4}[0,0] \oplus \Delta K_{(r),h-4} \quad (7) \\
K_{(r),h}[i,j] &= K_{(r),h}[0,0] \oplus \Delta K_{(r),h}. \quad (8)
\end{aligned}$$

By the schedule above when $h = 4, \cdots, 15$, the equations are:

$$\begin{aligned}
K_{(r-1),h}[i,j] &= K_{(r),h-4}[i,j] \oplus K_{(r),h}[i,j] \quad (9) \\
K_{(r-1),h}[0,0] &= K_{(r),h-4}[0,0] \oplus K_{(r),h}[0,0]. \\
&\quad (10)
\end{aligned}$$

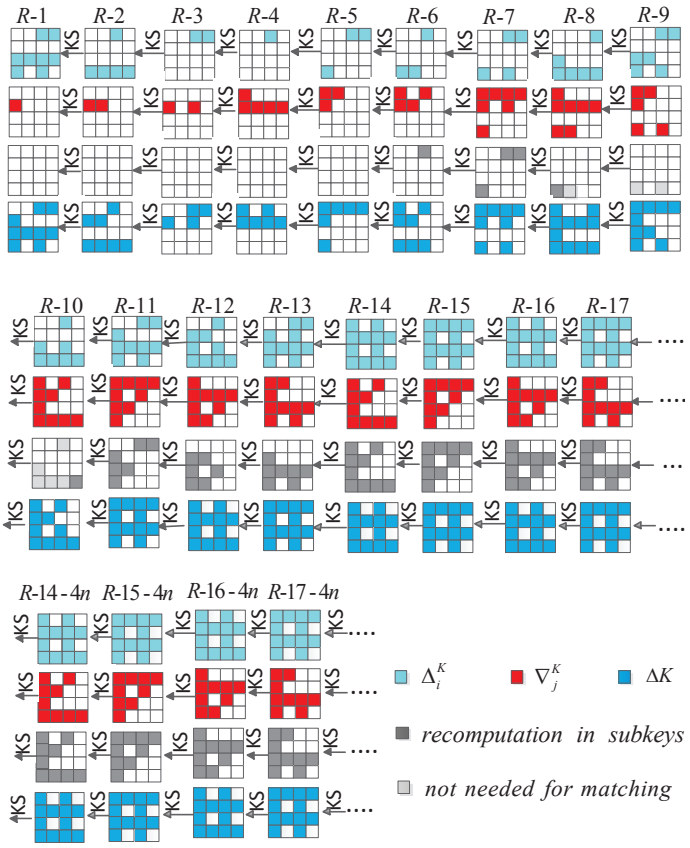


Figure 1: Computation in the subkeys form round -1 to round R-1

By Equations (7)-(10), we have:

$$\begin{aligned} K_{(r-1),h}[i,j] &\overset{(9)}{=} K_{(r),h-4}[i,j] \oplus K_{(r),h}[i,j] \\ &\overset{(7)(8)}{=} K_{(r),h-4}[0,0] \oplus \Delta K_{(r),h-4} \\ &\quad \oplus K_{(r),h}[0,0] \oplus \Delta K_{(r),h} \\ &\overset{(10)}{=} K_{(r-1),h-4}[0,0] \oplus \Delta K_{(r),h-4} \\ &\quad \oplus \Delta K_{(r),h}. \end{aligned} \tag{11}$$

From Equation (11), there is

$$\begin{aligned} K_{(r-1),h}[i,j] &= K_{(r-1),h-4}[0,0] \oplus \Delta K_{(r),h-4} \\ &\quad \oplus \Delta K_{(r),h}. \end{aligned}$$

So $K_{(r-1)}[i,j]$ can be expressed as the linear function of $K_{(r-1)}[0,0]$ and $\Delta K_{(r)}$.

□

Corollary 1. *The column 1,2,3 of $(r-1)$-round subkeys $K_{(r-1)}[i,j]$ are linear expressions of r-round subkeys $K_{(r-1)}[0,0]$ and $\Delta K_{(r)}$.*

Proof. Based on Lemma 1 in Equation (11). □

Corollary 2. *Only column 0 of $(r-1)$-round subkeys is a nonlinear expression of r-round subkeys $K_{(r-1)}[0,0], K_{(r)}[0,0]$ and $\Delta K_{(r)}$.*

Proof. Based on Lemma 1 in Equation (6). □

Corollary 3. *With the subkey recomputation, for $r \in \{-1, 0, \cdots, R-1\})$, if we know $K_{(r)}[0,0], \Delta K_{(r)}$ and $K_{(r-1)}[0,0]$, the recomputation for $(r-1)$-round subkeys $K_{(r-1)}[i,j]$, only appears in byte $0,1,2,3$, and they are only related to the last two columns of r-round subkeys $K_{(r)}[0,0]$ and $\Delta K_{(r)}$.*

Proof. By Corollary 2 and Lemma 1 in Equation (6). □

Lemma 2. *If*

$$\Delta_i^K = \begin{bmatrix} 0 & 0 & i & i \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \nabla_j^K = \begin{bmatrix} 0 & 0 & 0 & 0 \\ j & 0 & j & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$K^{(m)}[0,0] \in \begin{bmatrix} N & N & N & 0 \\ 0 & N & N & N \\ N & N & N & N \\ N & N & N & N \end{bmatrix}$$

in round $R-3$, computation $K^{(m)}[i,j]$ in the subkeys form round -1 to round $R-1$ is depicted in Figure 1 ($i,j \in GF(2^8)$, N denotes non-zero byte and 0 denotes zero byte).

Proof. The first, the second and fourth lines of Figure 1 above can be derived by the computer experiments based on the key schedule. The third line of Figure 1 can be derived by Lemma 1 and Corollarys 1, 2 and 3.

1) In the first line of Figure 1, Δ_i^K is depicted, and $K^{(m)}[i,0] = K^{(m)}[0,0] \oplus \Delta_i^K$.

2) In the second line of Figure 1, ∇_j^K is depicted, and $K^{(m)}[0,j] = K^{(m)}[0,0] \oplus \nabla_j^K$.

3) In the third line of Figure 1, the recomputation in the subkeys of $K^{(m)}[i,j]$ is depicted, where white cells need no recomputation because they are only related to Δ_i^K or ∇_j^K; Dark gray cells need recomputa-tion based on the Lemma 1 and Corollarys 1, 2 and 3; light gray cells are not required for matching.

4) In the fourth line of Figure 1, $\Delta K = K^{(m)}[i,j] \oplus K^{(m)}[0,0]$ is depicted based on the first line and the second line, where white cells denote zero difference.

□

Remark 3. *Our key partition in Round $R-3$ is the same as key partition in Round 8 [3], the computation of the subkeys in Figure 1 from $R-11$ to R is the same as Figure 9 [3] from -1 to 10.*

Lemma 3. *In Figure 1, the current equations are:*

1) $Pos_u(1) = Pos_{u-4v}(1)(u \le R-14, v = 0,1,\cdots)$ in Δ_i^K and ∇_j^K,where $Pos_u(1)$ denotes the position of non-zero differential in Round u.

2) $Pos_u(2) = Pos_{u-4v}(2)$ in recomputation of subkeys in Round u,where $Pos_u(2)$ denotes the position of recomputation in Round u.

Proof.

1) In the first, second and the fourth lines of Figure 1,it can be seen, the regularity from Round $R-4$ to Round -1, the position of non-zero differential byte in Δ_i^K, ∇_j^K and ΔK repeats every 4 rounds.

2) Similarly, in the third line of Figure 1, it can be shown (2) holds.

□

Theorem 1. *With the subkey recomputation technique, for $r \in \{-1,0,\cdots,R-1\}$), if we know r-round subkeys $K_{(r)}[0,0], \Delta K_{(r)}$, $K_{(r)}[i,j] = K_{(r)}[0,0] \oplus \Delta K_{(r)}$ and $(r-1)$-round subkeys $K_{(r-1)}[0,0]$, there are $(2R-16)$ S-boxes to be recomputed at most for the $(r-1)$-round subkeys $K_{(r-1)}[i,j]$ (Figure 1).*

Proof. We assume that we know r-round subkeys $K_{(r)}[0,0], \Delta K_{(r)}$ and $K_{(r)}[i,j] = K_{(r)}[0,0] \oplus \Delta K_{(r)}$.

When r is from round $R-1$ to round $R-8$,it is obviously there are all 2 S-boxes required to be recomputed for $K_{(r-1)}[i,j]$.

When $r \in \{-1,\cdots,R-9\}$, in the fourth line in Figure 1 based on Lemmas 2 and 3.

$$\Delta K_{r-1} = \begin{pmatrix} N & 0 & N & 0 \\ N & N & N & N \\ N & 0 & N & 0 \\ N & N & N & N \end{pmatrix} \text{ or } \begin{pmatrix} N & N & N & N \\ N & 0 & N & 0 \\ N & N & N & N \\ N & 0 & N & 0 \end{pmatrix}$$

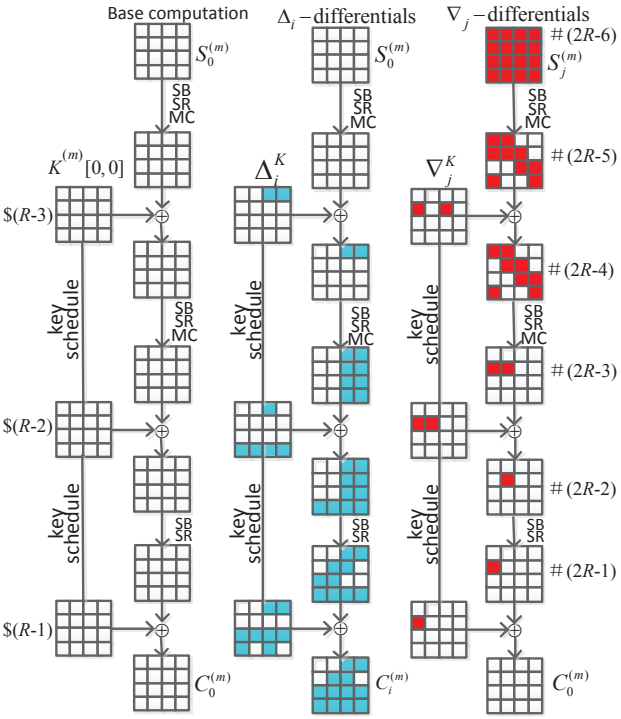


Figure 2: Biclique for R-round AES-128(R >10) [3]

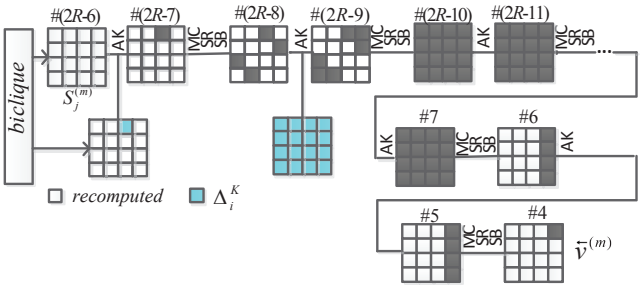


Figure 3: Recomputation in the forward direction for R-round AES-128(R>10)

where N denotes non-zero byte and 0 denotes zero byte. Then by the key schedule,

$$\Delta K_r = \begin{pmatrix} N & N & D_1 & D_2 \\ N & N & N & N \\ N & N & D_3 & D_4 \\ N & N & N & N \end{pmatrix} \text{ or } \begin{pmatrix} N & N & N & N \\ N & N & D_5 & D_6 \\ N & N & N & N \\ N & N & D_7 & D_8 \end{pmatrix}$$

It can therefore be demonstrated that: $D_1 = D_2, D_3 = D_4, D_5 = D_6, D_7 = D_8$. Based on Corollary 2 (*i.e.* Lemma 1 in Equation (4.6)), 2 S-boxes required to be recomputed at most in each round r.

So, there are $2(R-9)+2$ S-boxes be recomput-uted at most from round -1 to $R-1$ for $K_{(r-1)}[i,j]$.

Therefore, for the subkeys $K_r^{(m)}[i,j](r \in \{-1,0,\cdots,R-1\})$, there are $2R-16$ S-boxes to be recomputed at most. □

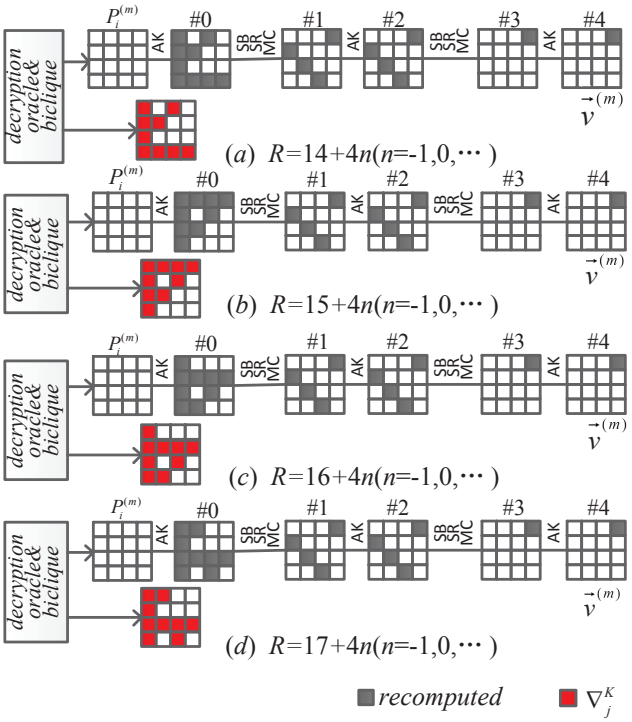


Figure 4: Recomputation in the forward direction for R-round AES-128(R>10)

4.2 Biclique Attack

Theorem 2. *For attacking R rounds of AES-128($R \geq 10$), the complexity of biclique attack*

$$C_{full} \leq 2^{121}(-\frac{752.4}{R} + 2^{6.86}).$$

Proof. For R-Round AES-128, biclique attack can be described in the following two steps: 3-round biclique construction and key recovery.

1) 3-round biclique construction. The 3-round biclique of dimension 8 $[\{C_i^{(m)}\}, \{S_j^{(m)}\}, \{K^{(m)}[i,j]\}]$ is constructed for AES-128 as shown in Figure 2,similar with Figure 4 [3].

2) Key recovery. We recover the secret key with recomputation technique.

Precompute $P_i^{(m)} \xrightarrow[g]{K^{(m)}[i,0]} \vec{v}_i^{(m)}$ and $\vec{v}_j^{(m)} \xleftarrow[h]{K^{(m)}[0,j]} S_j^{(m)}$. Then, store intermediate states and subkeys, where $\vec{v}_i^{(m)}$ and $\overleftarrow{v}_j^{(m)}$ are state #4.

- The amount of state recomputation in both directions is evaluated, where $S_j^{(m)}$ is the input of the round $R-3$.

  Backward direction: Recompute $\overleftarrow{v}^{(m)} \xleftarrow[h]{K^{(m)}[i,j]} S_j^{(m)}$ which is different from $\overleftarrow{v}_j^{(m)} \xleftarrow[h]{K^{(m)}[0,j]} S_j^{(m)}$. As shown in Figure 3, $41 + 16(R-10)$ S-boxes should be recomputed.

  Forward direction: Recompute $P_i^{(m)} \xrightarrow[g]{K^{(m)}[i,j]} \vec{v}^{(m)}$ which is different from $P_i^{(m)} \xrightarrow[g]{K^{(m)}[i,0]} \vec{v}_i^{(m)}$, therefore at most 13 S-boxes should be recomputed because the whitening subkeys of $K^{(m)}[i,j]$ and $K^{(m)}[i,0]$ differ in at most 9 bytes as shown in the line 2 of Figure 1. As shown in Figure4, for whitening subkeysK_{-1}, there are 4 kinds of difference between $K^{(m)}[i,j]$ and $K^{(m)}[i,0]$ when $R = 14+4n, R = 15+4n, R = 16+4n, R = 17+4n(n = -1, 0, \cdots)$.At most 13 S-boxes should be recomputed because the whitening subkeys of $K^{(m)}[i,j]$ and $K^{(m)}[i,0]$ differ in at most 9 bytes in four cases.

- The amount of subkey recomputation is evaluated.

  With the round increased, the recomputation of the subkeys cannot be ignored. First, the adversary computes and stores computations $K^{(m)}[i,0]$ and $K^{(m)}[0,j]$. Then, for other $K^{(m)}[i,j]$, he recomputes only those parts that differ from the stored ones.

  According to the Theorem 1, for the subkeys $K_r^{(m)}[i,j](r \in \{-1, 0, \cdots, R-1\})$, there are $2R-16$ S-boxes to be recomputed at most.

  In summary, for each $K^{(m)}$, $41 + 16(R-10) + 9 + (2R-16) = 18R-126$ S-box should be recomputed at most. The full time complexity of attacking R-round AES-128 is

$$C_{full} \leq 2^{112}\{2^7+2^7+2^{16}\times[18R-126]\times\frac{1}{16}\times\frac{4}{5R}+2^8\}$$
$$= 2^{121}(-\tfrac{752.4}{R} + 2^{6.86}) < 2^{127.86}$$

Theorem 2 shows: For attacking R rounds of AES-128 ($R \geq 10$), with the increase of number of rounds R, the complexity of biclique attack increases gradually, but it can never reach $2^{127.86}$. □

5 Biclique Attack on 10-Round AES-IND-128

AES-IND-128 denotes AES-128 independent of the key schedule. The AES key scheme can also be improved, so we need consider the AES-IND-128. In this paper, we assume the key schedule of AES-IND-128 satisfies: The diffusion property is enhanced so that any subkey byte in any round affects all the subkey bytes in the other rounds in terms of the difference.

Theorem 3. *The time complexity of the biclique attack on 10-round AES-IND-128 is* $2^{127.42}$.

Proof. For 10-round AES-IND-128, the biclique attack can be described in the following two steps: biclique construction and key recovery.

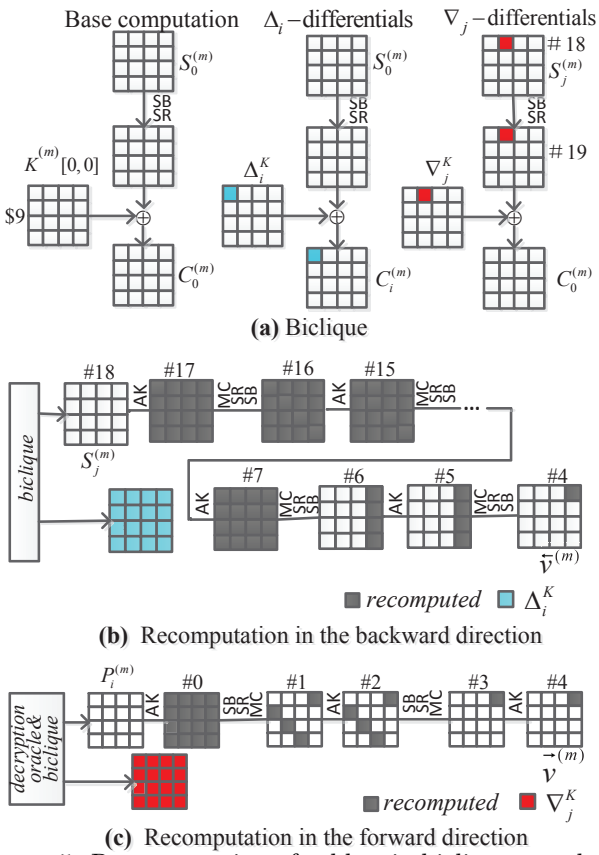


Figure 5: Recomputation of subkey in biclique attacks on 10-round AES-128-IND

1) 1-Round Biclique of Dimension 8. For 10-round AES-IND-128, it may fail to construct the 3-round or even 2-round biclique due to the diffusion properties of the internal rounds. Fortunately, it is feasible to construct 1-round biclique. In fact, what we should do is to find $\Delta_i^K \cap \Delta_j^K = \{0\}$ in round-9 subkey space.

- Key Partition. With the strategy of the key partition inside the biclique, define the key groups with respect to round-9 subkey space and enumerate the 2^{112} groups of 2^{16} keys: $K^{(m)}[i,j] = K^{(m)}[0,0] \oplus \Delta_i^K \oplus \nabla_j^K$ $(i,j = 0,1,\cdots,2^8-1; m = 0,1,\cdots 2^{112}-1)$, where

$$\Delta_i^K = \begin{bmatrix} i & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\nabla_j^K = \begin{bmatrix} 0 & j & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$K^{(m)}[0,0] \in \begin{bmatrix} 0 & 0 & N & N \\ N & N & N & N \\ N & N & N & N \\ N & N & N & N \end{bmatrix}$$

- 1-Round Biclique Construction. For each $K^{(m)}$, construct the biclique (Figure 5(a)). Fix $C_0^{(m)} = 0$ and computes $S_0^{(m)} = f^{-1}_{K^{(m)}[0,0]}(C_0^{(m)})$. Then, construct $\Delta_i^{(m)}$−differentials and $\nabla_j^{(m)}$−differentials. Finally, get the 1-round biclique of dimension 8.

  Because Δ_i-differentials influence the 1 bytes of the ciphertext, all the ciphertexts share the same values except bytes C_0.Therefore, the data complexity does not exceed 2^8.

2) Key Recovery.

- The amount of state recomputation in both directions is evaluated.

  Backward Direction: Because of the high diffusion of the AES-IND-128 key schedule, the subkeys 9 of $K^{(m)}[i,j]$ and $K^{(m)}[0,j]$ differ in all 16 bytes. As shown in Figure 5(b), 85 S-boxes should be recomputed.

  Forward Direction: The Whitening Subkey of $K^{(m)}[i,j]$ and $K^{(m)}[i,0]$ differ in all 16 bytes. As demonstrated in Figure 5(c), 20 S-boxes should be recomputed.

- The amount of subkey recomputation is evaluated.

  The diffusion of key schedule is enhanced: Any subkey byte in any round affects all the subkey bytes in the other rounds in terms of the difference. In the following, it is assumed that the nonlinear transform of the subkey generation does not change, $W_i \equiv W_{i-4} \oplus SB(RotByte(W_{i-1})) \oplus Rcon(i/4)(i \equiv 0 \bmod 4)$, S-boxes in subkey recomputation occurs in the first column of each round.

  Recomputations of the subkey in each round is located within TABLE 1.It is evident from the data within Table 1 that the following is true:

  Backward Direction: 4 S-boxes should be recomputed in round 3,4,5,6,7,8, respectively.

  Forward Direction: There are 4 S-boxes, 1S-box should be recomputed in round -1,0 respectively.

  So, for the subkeys $K_r^{(m)}[i,j](r \in \{-1,0,\cdots,R-1\})$, when Δ_i^K, ∇_j^K are computed,there are 29 S-boxes to be recomputed at most. Complexities: For each $K^{(m)}$,105+29=134 S-box should be recomputed, so

$$C_{recomp}^{(m)} = 2^{16} \times 134 \times 16^{-1} \times 12.5^{-1} \approx 2^{15.42}$$

  The time complexity of attacking10-round AES-IND-128 is

$$C_{full} = 2^{112}(2^7 + 2^7 + 2^{15.07} + 2^8) = 2^{127.42}$$

□

Theorem 3 shows: It can be demonstrated that although the emphasis was focused on the improvement of the AES-128 key schedule, the full time complexity of biclique attack on 10-round AES-128 cannot exceed $2^{127.42}$.

Table 1: Recomputation of subkey in biclique attacks on 10-round AES-128-IND

Round	Recomputation in subkey
-1	4
0	1
1	0
2	0
3	4
4	4
5	4
6	4
7	4
8	4

6 Conclusions

In this paper, the application of chosen ciphertext biclique attacks to AES-128 have been performed. Attacks on 10-Round AES-IND-128 and R-Round AES-128 ($R > 10$) are considered. 10-Round AES-IND-128 and R-Round AES-128 ($R > 10$) are more secure than 10-Round AES-IND-128 and 10-Round AES-128 in terms of the biclique attacks. Yet, it is evident that when the diffusion of key schedule is enhanced or the number of encryption rounds is increased, the biclique attacks remain effective to AES-128. So in order to make the biclique attack approximate exhaustive key attack in theory, we need not only enhance the diffusion of key schedule, but also increase the number of encryption rounds.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (No.61772418), Natural Science Basic Research Plan in Shaanxi Province of China (No.2017JQ6010) and National Cryptography Development Fund(No.MMJJ20180219)

References

[1] A. Bar-On, O. Dunkelman, N. Keller, E. Ronen, A. Shamir, "Improved key recovery attacks on reduced-round AES with practical data and memory complexities," in *Advances in Cryptology*, pp 187-212, 2018.

[2] A. Bogdanov, D. Chang, M. Ghosh, S. K. Sanadhya, "Bicliques with minimal data and time complexity for AES," in *International Conference on Information Security and Cryptology*, pp. 160-174, 2014.

[3] A. Bogdanov, D. Khovratovich, C. Rechberger, "Biclique cryptanalysis of the full AES," in *Advances in Cryptology*, pp. 344-371, 2011.

[4] J. Daemen, V. Rijmen, "The design of Rijndael," *Information Security and Cryptography*, 2002. (`https://autonome-antifa.org/IMG/pdf/Rijndael.pdf`)

[5] J. Daemen, L. Knudsen, V. Rijmen, "The block cipher SQUARE," in *International Workshop on Fast Software Encryption*, pp. 149-165, 1997.

[6] H. Demiric, A. Selcuk, "A meet in the middle attack on 8-round AES," in *International Workshop on Fast Software Encryption*, pp.116-126, 2008.

[7] P. Derbez, P. A. Fouque, J. Jean, "Improved key recovery attacks on reduced-round AES in the single-key setting," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 371-387, 2013.

[8] P. Derbez, P. A. Fouque, "Exhausting demirci-seluk meet-in-the-middle attacks against reduced-round AES," in *Computer Science*, pp. 541-560, 2013.

[9] O. Dunkelman, N. Keller, A. Shamir, "Improved single-key attacks on 8-round AES," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 158-176, 2010.

[10] N. Ferguson, J. Kelsey, S. Lucks, B. Schneier, M. Stay, D. Wagner, D. Whiting, "Improved cryptanalysis of Rijndael," in *International Workshop on Fast Software Encryption*, pp. 213-230, 2002.

[11] L. Grassi, C. Rechberger, S. Ronjom, "Subspace trail cryptanalysis and its applications to AES," in *IACR Transactions on Symmetric Cryptology*, vol. 2, pp. 192-225, 2017.

[12] H. Gilbert, M. Minier, "A collision attack on 7 rounds of Rijndael," in *AES Candidate Conference*, 2000. (`https://pdfs.semanticscholar.org/7405/c463a0d8477396c3a60408fb3ead0917bfb4.pdf`)

[13] L. Grassi, C. Rechberger, S. Ronjom,"A new structural-differential property of 5-round AES," in *Advances in Cryptology*, pp.289-317, 2017.

[14] J. Lu, O. Dunkelman, N. Keller, J. Kim, "New impossible differential attacks on AES," in *International Conference on Cryptology in India*, pp. 279-293, 2008.

[15] H. Mala, M. Dakhilalian, V. Rijmen, M. Modarres-Hashemi, "Improved impossible differential cryptanalysis of 7-round AES-128," in *International Conference on Cryptology in India*, pp. 282-291, 2010.

[16] A. Mirsaid, T. Gulom, "The encryption algorithm AES-RFWKIDEA32-1 based on network RFWKIDEA32-1," *International Journal of Electronics and Information Engineering*, vol. 4, no. 1, pp. 1-11, 2016.

[17] S. Ronjom, N. Bardeh, T. Helleseth, "Yoyo tricks with AES," in *Advances in Cryptology*, pp. 217-243, 2017.

[18] B. Sun, M. Liu, J. Guo, L. Qu, V. Rijmen, "New insights on AES-like SPN ciphers," in *Advances in Cryptology*, pp.605-624, 2016.

[19] T. Tiessen, "Polytopic cryptanalysis," in *Advancesin Cryptology*, pp. 214-239, 2016.

Biography

Dong Xiaoli is now a lecturer at Xian university of posts and telecommunications,China. She received the B.S. degree and the M.S. degree in mathematics major from Xi'an University of Technology in 2005 and 2008, and the Ph.D. degrees in cryptography science from Xidian University in 2011, respectively. Her main research fields include block cipher in cryptography, applied mathematics, and optimization algorithm in digital processing.

Chen Jie is now an associate professor at School of Telecommunication Engineering, Xidian University,China. She received the M.S. and Ph.D.degrees in cryptology from Xidian University in2005 and 2007, respectively. Her research fields include cryptography and network security.

Privacy-Preserving and Verifiable Electronic Voting Scheme Based on Smart Contract of Blockchain

Ting Liu, Zhe Cui, Hongjiang Du, and Zhihan Wu
(Corresponding author: Ting Liu)

Chengdu Institute of Computer Applications, Chinese Academy of Sciences
No. 9, Renmin South Road Section 4, Chengdu 610041, China
School of Computer and Control Engineering, University of Chinese Academy of Sciences
No. 19, Yuquan Road, Shijingshan District, Beijing 100049, China
(Email: liuting315@ mails.ucas.ac.cn)
(Received Sept. 2, 2019; Revised and Accepted Dec. 28, 2019; First Online Jan. 21, 2020)

Abstract

In this study, a privacy-preserving and verifiable electronic voting scheme is proposed based on a smart contract that is cost-effective and practical. The scheme uses electronic ballot as token for voting, and the smart contract verifies accuracy of the ballot. First, an agent generates electronic ballot via ElGamal encryption scheme, which is verified by the smart contract. The agent then generates decryption parameters based on the electronic ballot. Second, the agent assigns the electronic ballot to a voter and shares the decryption parameters to all voters with Shamir secret sharing scheme. Third, a voter generates and submits a vote that is the electronic ballot and a public parameter to the smart contract. Finally, the voter computes decryption data with the sum of decryption parameters restored by smart contract using shares summary submitted from voters. The voter then computes voting result via the homomorphic method with the decryption data. Experiment illustrates correctness and practicality of the proposed scheme.

Keywords: Block-Chain; Electronic Voting; Homomorphic Encryption; Smart Contract

1 Introduction

With the development and application of electronic technology, electronic voting (e-voting) has become an important method in various elections across the world [14,28]. E-voting uses computer and communication network technology to conduct voting activities with electronic ballot and digital vote instead of traditional paper printed ballot. It makes the voting more convenient and increases the efficiency of tallying votes with accuracy.

Various cryptography methods are implemented to preserve the privacy and verifiability of e-voting [17,18]. The first proposed method is termed as Mix-Nets that requires complex algorithms to protect voting privacy and realize public verification [7]. This was followed by blind signature based scheme, which depends on trusted signature institutions in the voting process. This type of scheme is not popular due to defects such as the complex of voting operation [1, 9, 13, 20]. Shamir secret sharing scheme (Shamir SSS) is another common cryptography method in e-voting. Shamir SSS splits secret digital information (an integer, for instance) into multiple shares, only some of which can restore the original secret information (the integer) [31]. In another study [21], the vote is encrypted via the ElGamal scheme, and the private key is shared to multiple authority centers by SSS to decipher the vote without restoring the key. Homomorphic cryptography is a common technology to protect voting privacy. By applying homomorphic cryptography, ciphertext of the votes tally are obtained via computing the ciphertext of votes, and the votes tally results are then decrypted. Hirt *et al.* used homomorphic cryptography to encrypt the vote, and verified the encryption by voters to ensure that the vote could not be tampered with [15]. In the scheme proposed by Ihsan Jabbar *et al.* [16], different servers encrypt the same ballot via homomorphic cryptography, then directly compute the encrypted ballots and decrypt the result to obtain the voting result. Literature [2] implements full homomorphic encryption based e-voting on cloud infrastructure. Liu *et al.* [24] took the votes of different candidates as Shamir SSS Lagrange polynomial coefficients and combined the homomorphic operation to verify tally result. Although the traditional cryptography schemes exhibit defects, such as high algorithm complexity, they are used to construct the e-voting schemes based on emerging technology to advance the progress in the field [3, 30].

Block-chain technology aims at that participants agree

on a series of consecutive blocks of transactions, invoke smart contract functions, and exchange assets [22, 36, 39, 42]. A few e-voting schemes apply block-chain to increase voting security and reduce the complexity combining with traditional cryptography methods. The schemes based on bitcoin protocol are studied as one of the main technical routes. This type of scheme necessitates the bitcoin for voting. Lee *et al.* [19] proposed a scheme conducting e-voting by means of bitcoin transaction with a third-party qualification audit mechanism. In 2017, Cruz *et al.* [10] proposed a block-chain e-voting scheme that adds ballot information to content of the transacted bitcoin. Zhao *et al.* [41] developed a voting system based on bitcoin with a mechanism to incentivize voting and zero-knowledge-proof on the vote commitment. Another type of e-voting scheme is based on Ethereum, which requires economy cost of Gas for the vote transaction [23,32,38]. The smart contract constitutes a main technology of block-chain voting system. In the protocol developed by McCorry *et al.* [26], the privacy of vote is protected via homomorphic encryption and all votes are tallied by a smart contract. Each voter broadcasts an encrypted vote , the legality of which is verified by a non-interactive zero-knowledge-proof. Literature [40] implements smart contract to verify the validity of encrypted votes during most voting stages. Currently, several voting systems based on block-chain have been developed such as BroncoVotes [11] and SecEVS [33].

The proposed scheme combines smart contract, Shamir SSS and homomorphic encryption to make e-voting privacy-preserving and verifiable. The bitcoin transaction has low time efficiency and necessitates cost economy. Thus the scheme using bitcoin as a token for voting is difficult to implement. E-voting based on Ethereum also necessitates cost economy. In this study, an agent generates electronic ballot as token for voting to a candidate that is similar to the scheme based on bitcoin. A voter transfers vote to candidate as bitcoin transaction that achieves the same credibility and lower cost. The voting scheme always uses a complex algorithm, such as zero-knowledge-proof, to provide proof of the validity of the encrypted vote during the voting process. Thus, the operating of voting becomes tedious and practically difficult. We avoided this problem by producing electronic ballot and verifying its correctness prior to voting. Additionally, by applying the designed electronic ballot and tally method, the implementation of the smart contract in the proposed scheme ensures reliability and efficiency of the block-chain operation. The proposed scheme needs further improvement for high efficiency when the voting has numerous voters.

The rest of this article is organized as follows. The next section gives the preliminaries used in the construction of our e-voting scheme. Section 3 describes technical route of the proposed e-voting scheme. Section 4 details the proposed scheme and the security analysis of it. Section 5 provides experiments on the scheme. Finally, conclusions are given in Section 6.

2 Preliminaries

In this section, we briefly introduce Shamir SSS, ElGamal encryption scheme, block-chain and bitcoin, smart contract, and cast-or-audit method.

2.1 Shamir SSS

Shamir SSS implements (k, n) threshold scheme that allows any k in all n secret shares to collaborate to retrieve the secret [31]. Shamir SSS shares secret polynomials

$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{k-1} x^{k-1}$$

to n shares $(x_i, f(x_i))$ such that $1 \le i \le n$, and restores the polynomials of $f(x)$ via Lagrange interpolation polynomial with k shares. In practical implementation, a_0 in $f(x)$ is shared as a secret, and it is restored as follows:

$$a_0 = (-1)^{k-1} \sum_{i=1}^{k} f(x_i) \prod_{j=1, j\neq i}^{k} \frac{x_j}{x_j - x_i}$$

The additive homomorphism of Shamir SSS is that the shares of multiple shared values are added together to restore the sum of all the values. The restoration of the sum of shared values are expressed by the formula as follows:

If $s = F_I(t_{i_1}, t_{i_2}, \ldots, t_{i_k})$ and $s' = F'_I(t'_{i_1}, t'_{i_2}, \ldots, t'_{i_k})$, then $s + s' = F_I(t_{i_1} + t'_{i_1}, t_{i_2} + t'_{i_2}, \ldots, t_{i_k} + t'_{i_k})$. F_I denotes Shamir SSS restoration algorithm, s and s' denote secret sharing values, $\{t_{i_1}, t_{i_2}, \ldots, t_{i_k}\}$ and $\{t'_{i_1}, t'_{i_2}, \ldots, t'_{i_k}\}$ denote the shares of s and s' [6].

2.2 ElGamal Encryption Scheme

ElGamal encryption scheme [12] is public-key cryptography and operated as follows.

1) Select a large prime q. Select numbers r and g that are both less than q. Compute $h = g^r \ mod \ q$;

2) Public key corresponds to h, private key corresponds to r, and public parameters corresponds to g and q;

3) Plain text M should be encrypted. Select a random integer k less than q, encrypt M to $\left(g^k, h^k M\right) \ mod \ q$;

4) Decrypt the ciphertext of M as $M = \left[h^k M (g^r)^{-k}\right] \ mod \ q$.

ElGamal encryption exhibits the homomorphic property as follows [8]. Character E symbolizes the encryption process, M_1 and M_2 are plain texts. The encryption of M_1 and M_2 corresponds to $E(M_1) = \left(g^{k_1}, h^{k_1} M_1\right)$ and $E(M_2) = \left(g^{k_2}, h^{k_2} M_2\right)$. It is defined that $k = k_1 + k_2$. Because $E(M_1) \times E(M_2) = \left(g^{k_1+k_2}, h^{k_1+k_2} M_1 M_2\right)$ and $E(M_1 M_2) = \left(g^k, h^k M_1 M_2\right)$, it is proven that $E(M_1 M_2) = E(M_1) \times E(M_2)$.

2.3 Block-Chain and Bitcoin

Block-chain was proposed by an anonymous scholar named "Satoshi" on the digital currency paper on bitcoin in 2007 [29]. Block-chain is an open ledger stored and maintained by different nodes. Block-chain technology realizes bitcoin trading without the participation of a trust center. The transaction is shared and stored by each node in the entire network. The bitcoin is the first application of block-chain, which safely and anonymously transacts electronic currency called bitcoin. Each bitcoin owner transfers the coin to the next by signing a hash of the previous transaction of the coin and the public key of the next owner. The public in the network can verify the signature to acquire bitcoin ownership. Sufficient computing power is necessary to transact the bitcoin, and the speed of transaction is low.

2.4 Smart Contract

The concept of intelligent contracts was first proposed in 1994 by Nick Szabo, and defined as a set of digitally specified commitments, including agreements on which the contracting parties can enforce the commitments [34]. Block-chain provides a trusted computing environment, and thus a smart contract is widely studied and implemented. The essence of a smart contract corresponds to code with specific transaction logic that runs on a block-chain. The status and content of a smart contract are public, and users of the chain can review the code to confirm the function of the contract. If the contract is confirmed, then it is not possible to tamper with the content of the contract [25, 27]. The smart contract runs on all verification nodes in the block-chain. When compiled and deployed, it can accurately respond to any parameter input. The process of execution is irreversible and cannot be forced to stop or interrupted midway [35, 37].

2.5 Cast-or-Audit Method

Benaloh [4, 5] elaborated this auditing approach to a system where the verifier marks her choice, the prover prepares the ballot. The verifier then chooses to either decrypt the prepared ballot, or to cast the prepared ballot. Since the prover is irrevocably committed to a particular encryption, and as the prover cannot predict whether the verifier will choose to cast or audit, any cheating by the prover has 50% chance of being audited (and, thus, detected). By repeating the audit as often as desired, the verifier can test the prover as often as desired and increase its confidence in the correctness of the prover's operation. The cast-or-audit method processes as follows.

1) The verifier sends the ballot to the prover. The prover encrypts the ballot with parameter and shows the encrypted ballot to the verifier. Then the verifier can choose two options: Cast or audit.

2) If the verifier chooses to audit the encryption, the prover shows the encryption parameter to it, and the verifier checks the encryption correctness. Then the prover encrypts the ballot with another parameter and the verifier chooses the options again.

3) If the verifier chooses to cast the ballot, it finishes verifying the encrypted ballot. Then the encrypted ballot is ready to be cast.

3 Technical Route of the Proposed E-voting Scheme

In the study, the voting agent and the smart contract are abbreviated as AGT and SC. We use a vote v of value 1, -1 or 0 to present yes vote, no vote or abstention vote for a candidate, respectively. The value of v also presents the difference between yes and no vote for the candidate. When a voter votes yes to a candidate for which $v = 1$, the candidate get 1 yes vote and 0 no vote. The difference between yes and no votes of the candidate gotten from this vote is 1, which equals the value of v. When a voter votes no to a candidate for which $v = -1$, the candidate get 0 yes vote and 1 no vote. The difference between yes and no votes gotten from this vote is -1, which equals the value of v. When a voter votes abstention to a candidate for which $v = 0$, the candidate get 0 yes vote and 0 no vote. The difference between yes and no votes of the candidate gotten from this vote is 0, which equals the value of v. The sum of v of all votes for a candidate equals the sum of all difference between yes and no votes for the candidate, from which the final voting result can be computed correctly. The technical route is given as follows:

1) Agent AGT generates the parameters of vote. A vote v can be divided into two parameters, namely the secret parameter e and public parameter u, product of which equals to v. AGT randomly selects a number corresponding to 1 or -1 as secret parameters e of the vote. The public parameter u is generated by AGT based on e. The parameters e and u of vote number v is shown as Table 1.

2) Smart contract SC verifies the validation of the electronic ballot via the cast-or-audit method detailed in Section 2.5. SC acts as a verifier and AGT acts as a prover. AGT encrypts e with a parameter t to $EB(e, t)$ as an electronic ballot. AGT shows t and e to prove the accuracy of each encryption. At the end of verification, t and e in the encrypted ballot that is ready to cast are maintained secret from SC.

3) AGT signs ID of the voter with $EB(e, t)$, then sends them with the signature to the voter.

4) AGT generates different public parameters u of the vote based on e. AGT creates decryption parameters $DP(u, t)$ with t and all different u. AGT sends all u and shares the corresponding $DP(u, t)$ via Shamir SSS to all voters.

Table 1: Parameters and presentations of a vote

Vote	Yes vote	No vote	Difference between yes and no vote	$v = e \times u$	e	u
Yes	1	0	1	1	1	1
					-1	-1
No	0	1	-1	-1	1	-1
					-1	1
Abstention	0	0	0	0	1	0
					-1	

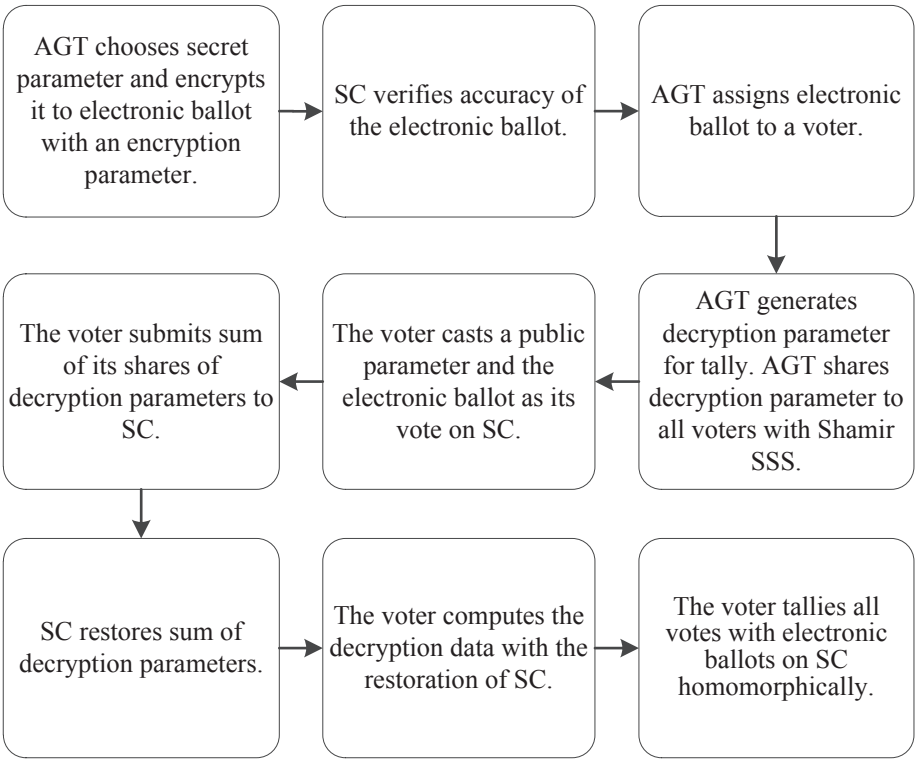


Figure 1: Technical route of the scheme

5) Each voter chooses and submits one $\{u, EB(e,t)\}$ as vote, and the signature of its ID and $EB(e,t)$ by AGT to SC.

6) Each voter submits its sum of shares to SC, and SC restores the sum of all shared $DP(u,t)$. Then the voter computes decryption data from restored sum of all $DP(u,t)$ on SC.

7) Voter tallies all votes saved on SC and decrypts the result with decryption data via homomorphic computation method. The tally result corresponds to the difference between the number of yes and no votes, which leads to the number of yes and no vote with the number of voters. The number of voters excluding the abandoning ones plus the sum of vote of this scheme, and the result summary divides 2 is the number of yes votes.

The technical route of the scheme is shown in Figure 1.

4 Implementation of the Proposed E-voting Scheme

The proposed scheme allows voters to vote from distance with the device of their own. In the proposed scheme, a voter can vote yes, no, and abstention to a candidate, and is allowed to give up voting. It is assumed that m voters are taking part in the voting. Each voter has its own public and private keys for encryption and signature. The signing operations by AGT and voter V_i are denoted as SIG_{AGT} and SIG_i respectively. The hash operation is denoted as $HASH$. This implementation shows the voting approach of V_i for C_j, and voting of V_i for others is the same.

4.1 Initialization

AGT generates its keys and parameters of the ElGamal scheme. AGT generates the secret parameters of votes

and encrypts them to electronic ballots, and the validity of which is verified by SC. Additionally, AGT prepares public parameters of different voting options for voters.

Step 1: As detailed in Section 2.2, AGT generates parameters and keys g, q, K_s, and K_p for ElGamal scheme.

Step 2: With respect to V_i, AGT selects a secret parameter $e_i \in \{-1, 1\}$ and generates random encryption parameter t_i. After encrypting e_i with t_i to electronic ballot that $EB(e_i, t_i) = g^{e_i}(K_p)^{t_i} \ mod\ q$, AGT invokes SC to verify the accuracy of the encryption with the method of cast-or-audit as shown in Section 2.5.

Step 3: The decryption parameter $DP(u_{i,j}, t_i)$ is defined as $DP(u_{i,j}, t_i) = k_s t_i u_{i,j}$. AGT computes 3 decryption parameters $\{DP(1, t_i), DP(-1, t_i), DP(0, t_i)\}$ with public parameters $\{1, -1, 0\}$.

4.2 Registration

The voter registers with AGT to prove its eligibility. AGT provides signed electronic ballot to the voter, and shares decryption parameters. V_i verifies the correctness of $EB(e_i, t_i)$ and AGT proves it via the cast-or-audit method detailed in Section 2.5.

Step 1: V_i sends its identification to AGT. If V_i is eligible, AGT continues the process, or else it rejects voting of V_i.

Step 2: V_i chooses an encryption $EB(e_i, t_i)$ recorded in SC. AGT provides t_i and e_i of the $EB(e_i, t_i)$ to V_i. V_i verifies the accuracy of $EB(e_i, t_i)$ and records e_i. AGT makes signature $SIG_{AGT}(HASH(V_i, EB(e_i, t_i)))$ and provides it to V_i for further operation. Then AGT saves $SIG_{AGT}(HASH(V_i, EB(e_i, t_i)))$ and $\{V_i, EB(e_i, t_i)\}$ on SC for public checking.

Step 3: AGT shares all three decryption parameters $\{DP(1, t_i), DP(-1, t_i), DP(0, t_i)\}$ with Shamir SSS to all voters via network. The shares for voters are denoted with corresponding public parameters 1, -1, and 0.

Step 4: AGT permanently deletes shares of decryption parameters, $DP(u_{i,j}, t_i)$, and $\{t_i, e_i\}$.

4.3 Voting

After registration, V_i submits its vote to the smart contract on its own device.

Step 1: V_i considers $\{V_i, EB(e_i, t_i), u_{i,j}\}$ as its vote for C_j. V_i generates the transaction of vote $Trans_{i,j} = \{V_i, EB(e_i, t_i), u_{i,j}, C_j\}$. V_i signs the hash of $Trans_{i,j}$ to $SIG_i(HASH(Trans_{i,j}))$, then submits $Trans_{i,j}$ and the signature to SC.

Step 2: After checking validity of $\{V_i, EB(e_i, t_i)\}$ saved by AGT in step 2 of the registration phase, SC verifies the signature, hash, and whether $u_{i,j} \in \{-1, 1, 0\}$. If the verification is successful, SC records $Trans_{i,j}$ and $SIG_i(HASH(Trans_{i,j}))$.

4.4 Tally Preparation

According the additive homomorphism of Shamir SSS, SC restores the sum of decryption parameters with the sum of shares submitted by voters. V_i computes decryption data with the restoration of SC.

Step 1: Based on the vote transaction saved on SC, each voter sums all the shares of $DP(u_{i,j}, t_i)$ of voters who cast votes, and submits the summary to SC.

Step 2: SC restores the sum of decryption parameter DP_j for C_j with the shares summary submitted by different voters so that

$$DP_j = \sum_{i=1}^{m} DP(u_{i,j}, t_i) = K_s \sum_{i=1}^{m} (t_i u_{i,j}).$$

Step 3: V_i checks correctness of the restoration with the shares on SC, and computes decryption data DD_j with DP_j.

$$DD_j = g^{DP_j} \ mod\ q = g^{K_s \sum_{i=1}^{m} (t_i u_{i,j})} \ mod\ q.$$

4.5 Tally

Voter V_i tallies the votes on SC with the decryption data.

Step 1: V_i tallies saved votes of C_j on SC by computing the following:

$$\begin{aligned} EB_j &= \prod_{i=1}^{m} (EB(e_i, t_i)^{u_{i,j}}) \\ &= g^{\sum_{i=1}^{m} e_i u_{i,j}} g^{K_s \sum_{i=1}^{m} (t_i u_{i,j})} \bmod q \\ EB_j / DD_j &= g^{\sum_{i=1}^{m} e_i u_{i,j}} \ mod\ q. \end{aligned} \quad (1)$$

Because the absolute value of $\sum_{i=1}^{m} e_i u_{i,j}$ is not exceeding the number of all voters, it is easy to compute

$$X_j = g^{\sum_{i=1}^{m} e_i u_{i,j}}$$

from the result of EB_j / DD_j computed from Equation (1).

Step 2: The vote which is the difference between yes and no votes of V_i voting for C_j is denoted as $v_{i,j}$. The sum of votes which is the sum of difference between yes and no votes to C_j from all voters is denoted as v_j. SC computes v_j as follows:

$$v_j = \sum_{i=1}^{m} v_{i,j} = \sum_{i=1}^{m} e_i u_{i,j} = \log_g X_j.$$

Step 3: V_i counts the number a_j of voters who voted abstention or abstain from voting for C_j on the SC. It is assumed that C_j gets the number of y_j yes votes and n_j no votes from all voters. As detailed in Section 3, because $m - a_j = y_j + n_j$, V_i obtains $y_j = \frac{m-a_j+v_j}{2}$ and $n_j = \frac{m-a_j-v_j}{2}$.

4.6 Scheme Analysis

In this section, based on all aforementioned methods, the primary information security of the proposed scheme is summarized as follows.

1) Eligibility: The voter registers its IDs with AGT, so that only eligible voter can get the electronic ballot signed by AGT to vote.

2) Privacy: At the end of the registration stage, AGT permanently deletes the parameters and shares which can reveal the vote or decrypt the information in the transaction. The voting privacy is provided by Shamir SSS and ElGamal encryption.

3) Verifiability: All encryptions of secret parameters are verified by SC without revealing the encryption parameters. Voter can check the verification programme deployed on SC. The vote with its voter and candidate are saved on the SC with signature of AGT, and it is not possible to tamper with any of the records without being discovered. According to the smart contract records, each voter can verify whether the voting result is correct.

4) Reliability: The scheme realizes a decentralized voting that is safe from attacks via internet. In the tally preparation stage, SC can compute the decryption data with submissions from only part of all voters, number of which is equal or exceeding the sharing threshold. Voters less than the threshold are unable to effect the tally result without being discovered.

5) Efficiency: In the electronic ballot generation, the encryption parameter, the secret parameter, and the public parameter are chosen to be positive or negative. So the data size in the tally stage is limited as the offset of positive or negative number via the additive homomorphic computation. The tally efficiency of the proposed scheme ensures its practical implementation.

5 Experiments

We considered a voting that 7 voters vote for a candidate as example of the proposed scheme. Because that the large size of data is difficult to be published in the article, the experiment was performed with short parameters. We implemented the proposed scheme based on Hyperledger Fabric 1.4 and the smart contract via the chaincode mechanism. All the symbols in this section have the same meaning with those in Section 4. The experiment of all stages in the proposed scheme is detailed as follows.

5.1 Initialization

Specifically, AGT chooses $q = 10007$, $g = 1009$, and $K_s = 1317$ for ElGamal encryption scheme. Seven voters, namely V_1, V_2, ..., V_6, and V_7 vote for candidate C_1. After verification of SC, the secret parameter e_i, encryption parameter t_i and electronic ballot $EB(e_i, t_i)$ of each voter are listed in Table 2.

Table 2: Parameters and encryptions in the initialization stage

i	V_i	e_i	t_i	$EB(e_i,t_i)$
1	V_1	1	76176	6107
2	V_2	-1	-73426	6799
3	V_3	1	-45241	562
4	V_4	-1	-88765	1013
5	V_5	1	84314	7430
6	V_6	-1	47838	2268
7	V_7	-1	81653	6522

5.2 Registration

AGT generates decryption parameter $DP(u_{i,1}, t_i)$ with public parameter $u_{i,1}$ such that $u_{i,1} \in \{-1, 1, 0\}$ for 3 types of options. The decryption parameter $DP(u_{i,1}, t_i)$ that V_i and public parameter $u_{i,1}$ of each voter voting for C_1 are listed in Table 3.

Table 3: Public parameters of votes and decryption parameters of votes

i	V_i	$u_{i,1}$	$DP(u_{i,1},t_i)$
1	V_1	1	100323792
2	V_2	-1	96702042
3	V_3	1	-59582397
4	V_4	0	0
5	V_5	1	111041538
6	V_6	1	63002646
7	V_7	1	107537001

The votes and their parameters are listed in Table 4.

Table 4: The votes and their parameters

i	V_i	$u_{i,1}$	e_i	$v_{i,1}$	Vote
1	V_1	1	1	1	Yes
2	V_2	-1	-1	1	Yes
3	V_3	1	1	1	Yes
4	V_4	0	-1	0	Abstention
5	V_5	1	1	1	Yes
6	V_6	1	-1	-1	No
7	V_7	1	-1	-1	No

5.3 Voting

Each vote $\{V_i, EB(e_i, t_i), u_{i,j}\}$ cast for C_1 is listed in Table 5.

Table 5: Votes cast for C_1

i	V_i	$EB(e_i, t_i)$	$u_{i,1}$
1	V_1	6107	1
2	V_2	6799	-1
3	V_3	562	1
4	V_4	1013	0
5	V_5	7430	1
6	V_6	2268	1
7	V_7	6522	1

5.4 Tally Preparation

The SC restores the sum of decryption parameters DP_1 for C_1 as $DP_1 = 89766720$. V_i computes the decryption data $DD_1 = 3353$.

5.5 Tally

For C_1, V_i computes the sum of difference between yes and no votes v_1, the abstention vote a_1, the yes vote y_1, and the no vote n_1 with decryption data DD_1 and the ballots multiplicative value EB_1. These data are listed in Table 6.

Table 6: Data and result of the tally

DD_1	EB_1	v_1	a_1	y_1	n_1
3353	8508	2	1	4	2

Because $y_1 = 4$, $n_1 = 2$, and $a_1 = 1$, the voting result is obtained as 4 yes votes, 2 no votes and 1 abandon vote, which are in consistent with all votes from voters shown in Table 4. So the correctness of the proposed scheme is experimented.

6 Conclusion

In the study, we proposed a novel e-voting scheme based on smart contract designed to ensure voting privacy and verifiability. In this scheme, the electronic ballot is generated as the token for voting to a candidate. Hence, the scheme is more cost-effective than the bitcoin-based and Ethereum-based scheme. Using the electronic ballot generated from encrypted secret parameter of vote with verification by a smart contract, this scheme does not necessitate complex zero-knowledge-proof during the voting period and is more practical. The proposed scheme realizes decentralization during vote casting and tallying, therefore attacking via network become very difficult. It achieves primary key security requirements of e-voting.

On-site registration should be discussed if there is a need for increasing information security such as receipt-freeness. On a physical site, it is possible to achieve more protection for voters against leaking voting information. The proposed scheme needs further improvement on the secret sharing scheme to be applied to large scale e-voting of numerous voters with high efficiency.

Acknowledgments

This study was financially supported by the National Natural Science Foundation of China under grant No. 61501064, Sichuan Technology Support Program under grant No. 2015GZ0088, Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis Open Fund under grant No. HCIC201502 and No. HCIC201701. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] K. M. R. Alam, A. Maruf, Md. R. R. Rakib, G. G. Md. N. Ali, P. H. J. Chong, and Y. Morimoto, "An untraceable voting scheme based on pairs of signature," *International Journal of Network Security*, vol. 20, no. 4, pp. 774–787, 2018.

[2] A. A. A. Aziz, H. N. Qunoo, and A. A. A. Samra, "Using homomorphic cryptographic solutions on e-voting systems," *International Journal of Computer Network and Information Security*, vol. 10, no. 1, pp. 44–59, 2018.

[3] S. Bartolucci, P. Bernat, and D. Joseph, "SHARVOT: Secret SHARe-based VOTing on the blockchain," in *IEEE/ACM 1st International Workshop on Emerging Trends in Software Engineering for Blockchain*, pp. 30–34, May 2018.

[4] J. Benaloh, "Simple verifiable elections," in *Proceedings of the USENIX Workshop on Accurate Electronic Voting Technology*, pp. 5–14, June 2006.

[5] J. Benaloh, "Ballot casting assurance via voter-initiated poll station auditing," in *Proceedings of the USENIX Workshop on Accurate Electronic Voting Technology*, pp. 14–20, June 2007.

[6] J. C. Benaloh, "Secret sharing homomorphisms: Keeping shares of a secret (extended abstract)," in *Conference on the Theory and Application of Cryptographic Techniques (CRYPTO'86)*, pp. 251–260, Aug. 1986.

[7] D. L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, vol. 24, no. 2, pp. 84–90, 1981.

[8] J. C. Corena and J. A. Posada, "Multiplexing schemes for homomorphic cryptosystems," *Elementos*, vol. 1, no. 1, pp. 21–32, 2013.

[9] L. F. Cranor and R. K. Cytron, "Sensus: A security-conscious electronic polling system for theinternet,"

in *The 30th Hawaii International Conference on System Sciences (HICSS'97)*, vol. 3, pp. 561–570, Jan. 1997.

[10] J. P. Cruz and Y. Kaji, "E-voting system based on the bitcoin protocol and blind signatures," *Transactions on Mathematical Modeling and Its Applications*, vol. 10, no. 1, pp. 14–22, 2017.

[11] G. G. Dagher, P. B. Marella, M. Milojkovic, and J. Mohler, "BroncoVotes: Secure voting system using ethereum's blockchain," in *The 4th International Conference on Information Systems Security and Privacy (ICISSP'18)*, pp. 96–107, Jan. 2018.

[12] T. Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469–472, 1985.

[13] A. Fujioka, T. Okamoto, and K. Ohta, "A practical secret voting scheme for large scale elections," in *The Workshop on the Theory & Application of Cryptographic Techniques (AUSCRYPT'92)*, pp. 244–251, Dec. 1992.

[14] J. P. Gibson, R. Krimmer, V. Teague, and J. Pomares, "A review of e-voting: The past, present and future," *Annals of Telecommunications*, vol. 71, no. 7-8, pp. 279–286, 2016.

[15] M. Hirt and K. Sako, "Efficient receipt-free voting based on homomorphic encryption," in *International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT'00)*, pp. 539–556, May 2000.

[16] I. Jabbar and N. A. Saad, "Design and implementation of secure remote e-voting system using homomorphic encryption," *International Journal of Network Security*, vol. 19, no. 5, pp. 694–703, 2017.

[17] R. Jardí-Cedó, J. Pujol-Ahulló, J. Castellí-Roca, and A. Viejo, "Study on poll-site voting and verification systems," *Computers & Security*, vol. 31, no. 8, pp. 989–1010, 2012.

[18] H. Jonker, S. Mauw, and J. Pang, "Privacy and verifiability in voting systems: Methods, developments and trends," *Computer Science Review*, vol. 10, pp. 1–30, 2013.

[19] K. Lee, J. I. James, T. G. Ejeta, and H. J. Kim, "Electronic voting service using block-chain," *The Journal of Digital Forensics, Security and Law*, vol. 11, no. 2, pp. 123–136, 2016.

[20] C. T. Li and M. S. Hwang, "Security enhancement of chang-lee anonymous e-voting scheme," *International Journal of Smart Home*, vol. 6, no. 2, pp. 45–52, 2012.

[21] Y. J. Li, C. G. Ma, and L. S. Huang, "An electronic voting scheme(in chinese)," *Journal of Software*, vol. 16, no. 10, pp. 1805–1810, 2005.

[22] I. C. Lin and T. C. Liao, "A survey of blockchain security issues and challenge," *International Journal of Network Security*, vol. 19, no. 5, pp. 653–659, 2017.

[23] V. C. T. Linh, C. M. Khoi, D. L. B. Chuong, and A. N. Tuan, "Votereum: An Ethereum-based e-voting system," in *IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF'19)*, pp. 1–6, Mar. 2019.

[24] Y. N. Liu and Q. Y. Zhao, "E-voting scheme using secret sharing and k-anonymity," *World Wide Web*, vol. 2, pp. 1657–1667, 2018.

[25] D. Macrinici, C. Cartofeanu, and S. Gao, "Smart contract applications within blockchain technology: A systematic mapping study," *Telematics and Informatics*, vol. 35, pp. 2337–2354, 2018.

[26] P. Mccorry, S. F. Shahandashti, and F. Hao, "A smart contract for boardroom voting with maximum voter privacy," in *International Conference on Financial Cryptography and Data Security (FC'17)*, pp. 357–375, Apr. 2017.

[27] W. Z. Meng, J. F. Wang, X. M. Wang, J. Liu, Z. X. Yu, J. Li, Y. J. Zhao, and S. S. M. Chow, "Position paper on blockchain technology: Smart contract and applications," in *The 12th International Conference on Network and System Security (NSS'18)*, pp. 474–483, Aug. 2018.

[28] M. F. M. Mursi, G. M. R. Assassa, A. Abdelhafez, and K. M. A. Samra, "On the development of electronic voting: A survey," *International Journal of Computer Applications*, vol. 61, no. 16, pp. 1–11, 2013.

[29] S. Nakamoto, *Bitcoin: A Peer-to-Peer Electronic Cash System*, 2019. (`https://bitcoin.org/bitcoin.pdf`)

[30] K. Nir and V. Jeffrey, "Blockchain-enabled e-voting," *IEEE Software*, vol. 35, no. 4, pp. 95–99, 2018.

[31] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[32] S. Shukla, A. N. Thasmiya, D. O. Shashank, and H. R. Mamatha, "Online voting application using ethereum blockchain," in *International Conference on Advances in Computing, Communications and Informatics (ICACCI'18)*, pp. 873–880, Sep. 2018.

[33] A. Singh and K. Chatterjee, "SecEVS: Secure electronic voting system using blockchain technology," in *International Conference on Computing, Power and Communication Technologies (GUCON'18)*, pp. 863–867, Sep. 2018.

[34] N. Szabo, "Formalizing and securing relationships on public networks," *First Monday*, vol. 2, no. 9, pp. 1–21, 1997.

[35] F. Tariq and R. Colomo-Palacios, "Use of blockchain smart contracts in software engineering: A systematic mapping," in *The 19th International Conference on Computational Science and Its Applications (ICCSA'19)*, pp. 327–337, Oct. 2019.

[36] S. Underwood, "Blockchain beyond bitcoin," *Communications of the ACM*, vol. 59, no. 11, pp. 15–17, 2016.

[37] S. Wang, L. Ouyang, Y. Yuan, X. C. Ni, X. Han, and F. Y. Wang, "Blockchain-enabled smart contracts: Architecture, applications, and future trends," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, pp. 1–12, 2019.

[38] X. Yang, X. Yi, S. Nepal, and F. L. Han, "Decentralized voting: A self-tallying voting system using a smart contract on the ethereum blockchain," in *The 19th Web Information Systems Engineering (WISE'18)*, pp. 18–35, Nov. 2018.

[39] Y. Yong and F. Y. Wang, "Blockchain: The state of the art and future trends," *Acta Automatica Sinica (in chinese)*, vol. 42, no. 4, pp. 81–94, 2016.

[40] W. Zhang, S. Huang, Y. Yuan, Y. Y. Hu, S. H. Huang, S. J. Cao, and A. Chopra, "A privacy-preserving voting protocol on blockchain," in *IEEE 11th International Conference on Cloud Computing (CLOUD'18)*, pp. 401–408, June 2018.

[41] Z. Zhao and T. H. Chan, "How to vote privately using bitcoin," in *The 17th International Conference on Information and Communications Security*, pp. 82–96, Dec. 2015.

[42] Z. Zheng, S. Xie, H. Dai, X. P. Chen, and H. M. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *IEEE International Congress on Big Data (BigData Congress)*, pp. 557–564, June 2017.

Biography

Ting Liu received the M.S degree in Computer Software and Theory from Xi'an Technological University in 2011. He is currently a Ph.D candiadate in University of Chinese Academy of Sciences. His research interests include Electronic Voting, Block-chain and Secret Sharing.

Zhe Cui received the Ph.D degree in Computer Software and Theory from University of Chinese Academy of Sciences in 2011. He is Ph.D supervisor and researcher of Chinese Academy of Sciences. His research interests include Pattern Recognition and Information Security.

Hongjiang Du received the M.S degree of Engineering in Computer Science and Technology from Sichuan University in 2006. He is currently a Ph.D candidate in University of Chinese Academy of Sciences. His research interests include Electronic Voting, Coding Theory, and Information Security.

Zhihan Wu received the Bachelor degree of Engineering in Information Scecurity from Sichuan University in 2017. She is currently a M.S candiadate in University of Chinese Academy of Sciences. Her research interests include Electronic voting, Block-chain, Cryptography.

Extension of PCL Theory and Its Application in Improved CCITT X.509 Analysis

Lei Yu[1], Zhi-Yao Yang[2], and Ze-Peng Zhuo[2]
(Corresponding author: Lei Yu)
School of Computer Science and Technology, Huaibei Normal Universtiy[1]
School of Mathematical Sciences, Huaibei Normal University [2]
Huaibei, Anhui 235000, China
(Email: yulei@chnu.edu.cn)
(Received Sept. 5, 2019; Revised and Accepted Dec. 6, 2020; First Online Apr. 3, 2020)

Abstract

In the original PCL theory, due to the lack of a strict definition and inference rules of the relations among message subterms, the protocol analysis process was described as having neither rigor nor formalization, which seriously affected the accuracy of the analysis results. Secondly, The temporal ordering between the actions of the principals is the key basis for judging whether the principals correctly perform the roles of the protocol or not. The analysis on it based on timestamp mechanism which can directly reflect the temporal ordering of the actions of the principals, will greatly reduce the complexity of protocol analysis. However, there are no verification or inference rules based on the timestamp mechanism for the temporal relationship between the acts of protocol principals in the existing PCL theory. Accordingly, this paper is aimed to extend the PCL theory from two aspects: Message subterms relationship and timestamp mechanism. First, the inference rules of message subterms are given on the basis of a strict definition of the relations between message subterms. Secondly, based on the defined timestamp relations and the original PCL inference system, the rules for judging the temporal ordering of the receiving and sending behavior of protocol principals are given. To verify the validity of PCL extension theory, the conciseness of PCL in protocol security analysis and the correctness of improved CCITT X.509 protocol, a formal description of the improved CCITT X.509 protocol is given by using cue calculus language, and a formal description of the security properties of the protocol is given by using PCL logic. And then, the security analysis of the protocol is given by using the extended PCL theory in three areas: Authentication, confidentiality and data integrity. The process and results of protocol analysis show that the extended PCL theory can effectively reduce the complexity of protocol analysis, and the improved CCITT X.509 protocol can meet the goal of protocol security attribute design.

Keywords: CCITT X.509; Formal Analysis Method; Protocol Composition Logic; Security Protocol

1 Introduction

Security protocol has become the basis of Cyberspace Security [20]. It is very complex to design a correct security protocol. Protocol defect analysis has become the main method and means of security protocol design [4, 16, 23]. At present, formal method has been proved to be the most scientific, rigorous and effective method of security protocol defect analysis [1–3]. Protocol Composition Logic (PCL) [7,9], proposed by Datta, Derek and Michell in 2003, is a formal design and analysis method of security protocols based on Floyd-Hoare logic. The formal proof system of PCL consists of protocol modeling system, protocol logic and proof system. In the protocol modeling system, PCL uses cryptographic primitives to describe the basic elements of the protocol, such as sending and receiving messages. Cords calculus links the security properties of the protocol with the execution semantics of the protocol. It can not only formally describe the protocol itself, but also accurately describe the security properties of the protocol. In the protocol logic system, PCL uses standard logic concepts such as predicate logic and model operators to eliminate the influence of informal factors such as "belief" and "jurisdiction" on the correctness of protocol analysis results. In the proof system, PCL adopts honest rules and does not need explicit inference about intruder's behaviors, greatly reducing the complexity of protocol analysis process. In addition, the logical inference system of PCL can ensure the security analysis of parallel and sequential combination of protocols. Therefore, PCL has scientific and rigorous inference system, as well as flexible and efficient analysis methods, compared with other formal analysis methods. PCL has been broadly extended and improved by researchers [8,21,22,25] in recent years. So far, PCL has been widely used in formal design and analysis of protocols [10, 14, 15, 18]. However, there are still many flaws in the PCL theory, such as inadequate

formal theory of message algebraic space, limitations of honesty theory, *etc.* [6].

When the PCL theory is adopted to analyze security protocols, some issues are found, e.g. the message space theory of PCL is less systematic; the definition of message structure, message types and the relations among message subterms are not rigorous; and the inference rules of the relation among message subterms are missing. In the process of protocol analysis, $contains(a, b)$ only be used to assert the subterms relations between different messages. The inference of the subterm relations of messages is latent, subjective and informal. In order to prevent message replay attacks, timestamp mechanism is often used to ensure the freshness of messages in the process of security protocol design. For example, timestamp mechanism is used in CITT X.509, Denning-Sacco, Wide-Mouth Frog, and Kerberos. Timestamp not only prevents message replay attacks, but also potentially establishes the temporal relationship between the actions of principals. In the PCL logical inference system, the temporal relationship between the behaviors of protocol principals is the key basis to judge whether the protocol is executed correctly. The logical inference and establishment of the temporal relationship between the actions of protocol principals are not only the basis of correctness analysis of protocol security properties, but also the key of PCL analysis method that determines the complexity of protocol analysis. Timestamp can directly reflect the timing relationship of the action of the principals, so the complexity of protocol analysis will be greatly reduced, if the timestamp is bound to the action of the principals to analyze the temporal relationship of the action of the principals [5]. In the existing PCL theory, there are no verification or inference rules based on timestamp mechanism for the temporal relationship of the actions of protocol principals. Therefore, this paper is focused on expanding the original PCL theory from two aspects - message subterms relationship and timestamp mechanism, in order to further improve the theoretical basis of PCL, expand the application scope of PCL theory, improve the formalization of protocol analysis, and enhance the efficiency and accuracy of protocol analysis.

CITT X.509 [12] is a security protocol based on public key cryptosystem. In its design, not only random values but also timestamp mechanism are used. The security properties of CITT X.509 consist of authentication, confidentiality and data integrity. The actions of principals contain many basic message operation types, such as encryption, decryption, signature, verification and so on. The diversity of CITT X.509 in security mechanism, security objectives and message operation types requires higher theoretical basis and inference system of formal methods. Since the publication of CITT X.509, some security defects in authentication and confidentiality have been detected by various security protocol analysis methods [11, 13, 17–19, 24]. Researchers have addressed a variety of improvement schemes by reconstructing message structure. Based on the improved scheme of CITT X.509 given by literature [29], the security proof of improved CITT X.509 protocol is delivered by using extended PCL logic to verify the efficiency of extended PCL logic and the correctness of improved CITT X.509 protocol.

2 Protocol Composition Logic

2.1 Symbols and Terminology

The basic symbols and terms used in this paper are as follows.

1) ρ : The role in the protocol;
2) $\hat{X}$: Principal that performs protocol role;
3) t: term;
4) T_{stamp}: The set of timestamps;
5) $m(X, Y, t)$: Formatted message terms. X is the sender of the message, Y is the receiver of the message, and t is the content of the message;
6) K_X: Key set of principal $\hat{X}$;
7) k_X, k_X^{-1}: The public and private keys of principal $\hat{X}$;
8) K_{XY}: Shared key of principal $\hat{X}$ and $\hat{Y}$;
9) $\{t\}_k$: Encryption of term t with key k;
10) $|t|_k$: Signature of term t with key k;
11) gh: Connection of term g and h;
12) α, β: Actions of the principal;
13) **a,b**: Action formula;
14) S: Strands;
15) P: Threads;
16) n: Random value;
17) $\top$: True.

2.2 Protocol Programming Language

PCL uses a protocol programming language based on Cords calculus to describe protocol message interaction. The formal definitions of message operation and message sequence are given below.

1) *new* t: Generate a new term t;
2) *send* u: Send a term u;
3) *receive* u: Receive a term u;
4) *match* u, u: Match a term to a patter;
5) $x := sign\ u, k$: sign the term u with k;
6) $verify\ u, u, k$: Verify the signature;

7) $x := enc\ u, k$: Encrypt the term u with k;

8) $x := dec\ u, k$: Decrypt the term u with k;

9) $x := gh$: Tuple the term g and h;

10) $[\alpha; \cdots ; \alpha]_P$: Actions sequence of $\hat{P}$.

2.3 Protocol Logic

1) Action formulas.

$$\mathbf{a} ::= Send(X,t)|Receive(X,t)|New(X,t)|$$
$$Encrypt(X,t)|Decrypt(X,t)|Sign(X,t)|$$
$$Verify(X,t)|Match(X,t)|Tuple(X,t).$$

2) Logic formulas.

$$\phi ::= \mathbf{a}|Has(X,t)|Fresh(X,t)|Gen(X,t)|$$
$$FirstSend(X,t,t')Honest(X)|t = t|$$
$$Contains(t,t')|\phi \wedge \phi|\neg\phi|Start(X)|\mathbf{a} < \mathbf{b}.$$

3) Modal formulas.

$$\theta ::= \phi S \phi.$$

2.4 Inference System

According to the function of inference formula, the inference formula of PCL is divided into seven types. The proof of inference formula is detailed in reference [8].

1) Protocol actions.

AA1 $\top[\alpha]_X \mathbf{a}$

AA2 $Start(X)[\ ]_X \neg \mathbf{a}(X)$

AA3 $\neg Send(X,t)[\alpha]_X \neg\ Send(X,t)$

AA4 $\top[\alpha; \cdots ; \beta]_X \mathbf{a} < \mathbf{b}$

AN1 $New(X,t) \wedge New(Y,t) \supset X = Y$

AN2 $\top[new\ t]_X Has(Y,t) \supset Y = X$

AN3 $\top[new\ t]_X Fresh(X,t)$

AN4 $Fresh(X,t) \supset Gen(X,t)$

2) Possession axioms.

AM1 $\top[\ ]_X Has(X, K_X)$

AM2 $(x, \cdots)[\ ]_X Has(X,x)$

ORIG $New(X,t) \supset Has(X,t)$

REC $Receive(X,t) \supset Has(X,t)$

TUP $Has(X,a) \wedge Has(X,b) \supset Has(X,ab)$

ENC $Has(X,t) \wedge Has(X,k) \supset Has(X,\{t\}_k)$

DEC $Has(X,\{t\}_k) \wedge Has(X,k^{-1}) \supset Has(X,t)$

PROJ $Has(X,ab) \supset Has(X,a) \wedge Has(X,b)$

GEN1 $Has(X,t) \wedge Has(X,k) \wedge Send(X,\{t\}_k)$
$\supset Gen(X,\{t\}_k)$

GEN2 $Gen(X,\{t\}_k) \supset Has(X,t)$

GEN3 $Gen(X,\{t\}_k) \supset Has(X,k)$

3) Encryption and signature.

SEC $Honest(\hat{X}) \wedge Decrypt(Y,\{x\}_{k_{\hat{X}}}) \supset \hat{Y} = \hat{X}$

VER $Honest(\hat{X}) \wedge Verify(Y,|x|^{-1}_{k_{\hat{X}}}) \wedge \hat{X} \neq \hat{Y} \supset$
$\exists X.Sign(X,|x|^{-1}_{k_{\hat{X}}}) \wedge Send(X,m)$
$\wedge Contains(m,x)$

4) Preservation axioms.

P1 $Persist(X,t)[\alpha]_X Persist(X,t)$
where $Persist \in \{Has, FirstSend, \mathbf{a}, Gen\}$

P2 $Fresh(X,t)[\alpha]_X Fresh(X,t)$
where $\neg Contains(t,a)$

5) Temporal ordering.

FS1 $Fresh(X,t)[send\ t']_X FirstSend(X,t,t')$
where $Contains(t',t)$

FS2 $FirstSend(X,t',t) \vee \mathbf{a}(Y,t'') \supset$
$Send(X,t') < \mathbf{a}(Y,t'')$
where $X \neq Y \wedge Contains(t'',t)$

6) Generic rules.

G1 $\dfrac{\theta[P]_X\phi \qquad \theta[P]_X\psi}{\theta[P]_X\phi \wedge \psi}$

G2 $\dfrac{\theta[P]_X\psi \qquad \phi[P]_X\psi}{\theta \wedge \phi[P]_X\psi}$

G3 $\dfrac{\theta[P]_X\phi}{\theta'[P]_X\phi'}$
where $Contains(\theta',\theta) \wedge Contains(\phi',\phi)$

G4 $\dfrac{\phi_1[P]_X\phi_2 \qquad \phi_2[P']_X\phi_3}{\phi_1[PP']_X\phi_3}$

7) Honesty rule.

HON$_Q$ $\forall \rho \in Q \cdot \forall P \in BS(\rho)$
$$\frac{Start(X)[\ ]_X\phi \quad \phi[P]_X\phi}{Honest(\hat{X}) \supset \phi}$$

2.5 Initial Configuration of Protocol

Definition 1. *Let C be initial configuration of protocol Q,C is determined by:*

1) A group of principals, some of which are designated as honest.

2) A cord space constructed by assigning roles of Q to threads of honest principals.

3) One or more intruder cords, which may use keys of dishonest principals.

4) A finite number of buffer cords, enough to accommodate every send action by honest threads and the intruder threads.

3 Extension of PCL

3.1 Subterm Relations

In the existing PCL theory, only one attribute assertion $Contains(a, b)$ is given to indicate that message a is a subterm of b, but it does not give a strict definition of message subterm relationship, nor does it give the relevant inference rules of message subterm relationship. When in using PCL for security protocol analysis, the inference of message subterm relationship is latent and subjective, and this makes the protocol analysis process lack of rigorous theoretical basis and formal methods, and directly affects the correctness of protocol analysis results.

Definition 2. *Let t,g,h be message terms and k be the key of the protocol principle, the message subterm relationship can be defined recursively as follows:*

1) $Contains(t, t)$,*Message term is its own subterm;*
2) $Contains(t, \{h\}_k)$,*If and only if* $Contains(t, h) \vee t = \{h\}_k$;
3) $Contains(t, |h|_k)$,*If and only if* $Contains(t, h) \vee t = |h|_k$;
4) $contains(t, gh)$,*If and only if* $Contains(t, g) \vee Contains(t, h)$.

The inference rules of message subterm relations can be given from Definition 2:

STR1 $gh \supset Contains(g, gh) \wedge Contains(h, gh) \wedge Contains(gh, gh)$

STR2 $\{t\}_k \supset Contains(t, \{t\}_k) \wedge Contains(\{t\}_k, \{t\}_k)$

STR3 $|t|_k \supset Contains(t, |t|_k) \wedge Contains(|t|_k, |t|_k)$

STR4 $Contains(t, g) \wedge Contains(g, h) \supset Contains(t, h)$

3.2 Timestamp Mechanism

Timestamp is the main mechanism to guarantee the freshness of message terms in security protocols. The design of CCITT X.509 protocol adopts timestamp mechanism. In the existing PCL inference system, there is no rule of verification and inference based on timestamp mechanism to judge the temporal relationship of the behaviors of the principals, and it is impossible to formally express and inference the timestamp mechanism in the protocol correctly. In order to reduce the complexity of protocol analysis and improve the efficiency of protocol analysis, it is necessary to extend the logic inference system of PCL from the aspect of timestamp mechanism.

Timestamp exists in the form of message subterms, which are bound to the actions of the principals. How to formalize the relationship between message terms and the actions of protocol principals, and the temporal relationship between different actions based on timestamps are not addressed in the existing PCL theory.

Definition 3. *Let m be the message term of action $\boldsymbol{a}$, and then m is defined as $term(\boldsymbol{a})$, i.e. $m = term(\boldsymbol{a})$.*

Definition 4. *Let t_1 and t_2 be timestamp constants created at protocol runtime:*

1) *if t_1 is created before t_2, the relationship between t_1 and t_2 is defined as $t_1 < t_2$;*
2) *if t_1 and t_2 are created at the same time, the relationship between t_1 and t_2 is defined as $t_1 = t_2$.*

Property 1. *Let t be the timestamp constant created by the protocol runtime and t_{sys} the current time, then $t <= t_{sys}$.*

Theorem 1. *Let $\hat{X}, \hat{Y}$ be the principal role of the protocol; t_1 and t_2 be timestamp constants, and $t_1 < t_2$, m_1 and m_2 be terms and $FirstSend(X, t_1, m_1) \wedge FirstSend(X, t_2, m_2)$. Then $Send(X, m_1) < Send(X, m_2)$.*

Theorem 2. *Let $\hat{X}, \hat{Y}$ be the principal role of the protocol; t be timestamp constants, $\boldsymbol{a}$ be the action assertion, m be term and $Contains(m, t) \wedge term(\boldsymbol{a}) = m$, $FirstSend(X, t, m)$. Then $Send(X, m) <= \boldsymbol{a}$.*

Theorems 1 and 2 can be proved by Definition 2, **AA1** and **AA4**, which are not discussed here.

Corollary 1 can be derived from Theorem 2.

Corollary 1. *Let $\hat{X}, \hat{Y}$ be the principal role of the protocol, t be the timestamp, t_{sys} be the current time, m be the term and $Contains(m, t)$. Then $Send(X, m) < Receive(Y, m)$.*

From Theorem 1, Theorem 2 and Corollary 1, the following inference rules can be given.

TT1 $$\frac{FirstSend(X, t_1, m_1) \wedge FirstSend(\hat{Y}, t_2, m_2)}{(t_1 < t_2) \supset Send(X, m_1) < Send(Y, m_2)}$$ where $t_1, t_2 \in T_{stamp}$

TT2 $$\frac{FirstSend(X, t, m)}{Contain(term(\mathbf{a}), t) \supset Send(X, m) < \mathbf{a}}$$

TT3 $$\frac{FirstSend(X, t, m)}{Send(X, m) < Receive(Y, m)}$$ where $t \in T_{stamp}$

4 Improved CCITT X.509

4.1 Improved CCITTX.509 Modeling

The improved CCITTX.509 protocol execution process is shown in Figure 1.

CCITTX.509 protocol is based on public key cryptosystem. It has two roles: A initiator and B responder. T_a and T_b are the timestamps produced by A and B, N_a and N_b are the random numbers generated by A and B, X_a and Y_a are the data generated by A, X_a and X_b are the data generated by B, k_A and k_A^{-1} are the public and private keys of A, and k_B and k_B^{-1} are the public and private keys of B.

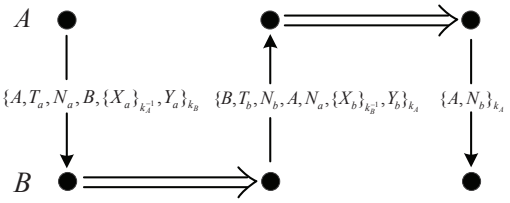


Figure 1: Graph of improved CCITT X.509 protocol

Definition 5. *Let $Init_{X.509}$ and $Resp_{X.509}$ be the initiator and responder roles of the improved CCITTX.509 respectively. $\hat{A}$ and $\hat{B}$ are the principals of the protocol roles, then $Init_{X.509}$ and $Resp_{X.509}$ are defined as below:*

$$
\begin{aligned}
Init_{X.509} \equiv (\hat{A},\hat{B})[\\
&new\ T_a;\\
&new\ N_a;\\
&new\ X_a;\\
&t_1 := enc\ X_a, k_A^{-1};\\
&new\ Y_a;\\
&t_2 := \{A,T_a,N_a,B,t_1,Y_a\};\\
&m_1 := enc\ t_2, k_B;\\
&send\ m_1;\\
&receive\ m_2;\\
&t_3 := dec\ m_2, k_A^{-1};\\
&match\ t_3, \{B,T_b,N_b,A,N_a,t_4,Y_b\};\\
&verify\ t_4, X_b, k_B;\\
&t_5 := \{A,N_b\};\\
&m_3 := enc\ t_5, k_A;\\
&send\ m_3;\\
&]_A <> .
\end{aligned}
$$

$$
\begin{aligned}
Resp_{X.509} \equiv (\ )[\\
&receive\ n_1;\\
&r_1 := dec\ n_1, k_B^{-1};\\
&match\ r_1, \{A,T_a,N_a,B,r_2,Y_a\};\\
&verify\ r_2, X_a, k_A;\\
&new\ T_b;\\
&new\ N_b;\\
&new\ X_b;\\
&r_3 := sign\ X_b, k_B^{-1};\\
&new\ Y_b;\\
&n_2 := enc\ \{B,T_b,N_b,A,N_a,r_3,Y_b\}, k_A;\\
&send\ n_2;\\
&receive\ n_3;\\
&r_4 := dec\ n_3, k_B^{-1};\\
&match\ r_4, \{A,N_b\};\\
&]_B <> .
\end{aligned}
$$

4.2 Protocol Attribute Modeling

CCITTX.509 protocol is designed to share Y_a and Y_b on the basis of mutual authentication of principals. The protocol uses timestamps T_a and T_b, as well as random values N_a and N_b to ensure the freshness of message terms. Encryption with private key signature ensures the integrity of X_a and X_b, and encryption with public key ensures the confidentiality of Y_a and Y_b. Therefore, the main security properties of the protocol include authentication, secrecy and data integrity.

1) Authentication.

Definition 6. *Let $\hat{A}$ be the principal of $Init_{X.509}$ and $\hat{B}$ be the principal of $Resp_{X.509}$. If the protocol satisfies the mutual authentication between $\hat{A}$ and $\hat{B}$, $\phi_{AUTH}(\hat{A})$ is the authentication of $\hat{A}$ to $\hat{B}$, and $\phi_{AUTH}(\hat{B})$ is the authentication of $\hat{B}$ to $\hat{A}$, then:*

$$
\begin{aligned}
\phi_{AUTH}(\hat{A}) \equiv \exists B.(((&Send(A,m_1) < Receive(B,m_1))\\
&\wedge (Receive(B,m_1) < Send(B,m_2))\\
&\wedge (Send(B,m_2) < Receive(A,m_2))\\
&\wedge (Receive(A,m_2 < Send(A,m_3))).
\end{aligned}
$$

$$
\begin{aligned}
\phi_{AUTH}(\hat{B}) \equiv \exists A.(((&Send(A,n_1) < Receive(B,n_1))\\
&\wedge (Receive(B,n_1) < Send(B,n_2))\\
&\wedge (Send(B,n_2) < Receive(A,n_2))\\
&\wedge (Receive(A,n_2 < Send(A,n_3))).
\end{aligned}
$$

2) Data secrecy.

Definition 7. *Let $\hat{A}$ be the principal of $Init_{X.509}$ and $\hat{B}$ be the principal of $Resp_{X.509}$. $\phi_{SEC}(\hat{A})$ denotes that $Init_{X.509}$ satisfies the confidentiality of Y_a and Y_b, and $\phi_{SEC}(\hat{B})$ denotes that $Resp_{X.509}$ satisfies the confidentiality of Y_a and Y_b. Then:*

$$\phi_{SEC}(\hat{A}) \equiv \exists Z.Has(Z,(Y_a,Y_b)) \supset (Z = A \vee Z = B)$$

$$\phi_{SEC}(\hat{B}) \equiv \exists Z.Has(Z,(Y_a,Y_b)) \supset (Z = A \vee Z = B)$$

3) Data integrity.

Definition 8. *Let $\hat{A}$ be the principal of $Init_{X.509}$ and $\hat{B}$ be the principal of $Resp_{X.509}$; $\phi_{INTE}(\hat{A})$ denotes the integrity of X_b to $\hat{A}$, $\phi_{INTE}(\hat{B})$ denotes the integrity of X_a to $\hat{B}$, then:*

$$
\begin{aligned}
\phi_{INTE}(\hat{A}) \equiv \exists Z.Sign(Z,\{X_b\}_{k_Z^{-1}}) \wedge Send(Z,m)\\
\wedge Contains(m,\{X_b\}_{k_Z^{-1}}) \supset Z = B
\end{aligned}
$$

$$
\begin{aligned}
\phi_{INTE}(\hat{B}) \equiv \exists Z.Sign(Z,\{X_a\}_{k_Z^{-1}}) \wedge Send(Z,m)\\
\wedge Contains(m,\{X_a\}_{k_Z^{-1}}) \supset Z = A
\end{aligned}
$$

5 Analysis of Improved CCITT X.509

5.1 Authentication Analysis

Proposition 1. *Let C be initial configuration of improved CCITTX.509, $\hat{A}$ and $\hat{B}$ be the principal of the*

initiator and responder roles respectively. If principal $\hat{A}, \hat{B} \in Honest(C)$*, then* $\phi_{AUTH}(\hat{A})$ *is true.*

Proof 1.

$$\textbf{AM1}, \textbf{AM2} \quad (\hat{A}, \hat{B})[\,]_A Has(A, A) \wedge Has(A, B) \wedge Has(A, k_A^{-1}) \wedge Has(A, k_B) \quad (1)$$

$$\textbf{AN2}, \textbf{AN2} \quad \top[new\ n_a]_A Has(A, N_a) \wedge Fresh(A, N_a) \quad (2)$$

$$\textbf{AN2}, \textbf{AN3} \quad \top[new\ T_a]_A Has(A, T_a) \wedge Fresh(A, T_a) \quad (3)$$

$$\textbf{TUP}, \textbf{STR1} \quad \top[t_2 := \{A, T_a, N_a, B, t_1, Y_a\}]_A\ Tuple(A, t_2) \supset Contains(N_a, t_2) \wedge Contains(T_a, t_2) \quad (4)$$

$$\textbf{STR2}, \textbf{STR4} \quad \top[m_1 := enc\ t_2, k_B]_A Encrypt(A, t_2) \supset Contains(N_a, m_1) \wedge Contains(T_a, m_1) \quad (5)$$

$$\textbf{2}, \textbf{5}, \textbf{AA1}, \textbf{FS1} \quad Fresh(A, N_a) \wedge Contains(N_a, m_1)\ [send\ m_1]_A FirstSend(A, N_a, m_1) \quad (6)$$

$$\textbf{2}, \textbf{5}, \textbf{AA1}, \textbf{FS1} \quad Fresh(A, T_a) \wedge Contains(T_a, m_1)\ [send\ m_1]_A FirstSend(A, T_a, m_1) \quad (7)$$

$$\textbf{AA1}, \textbf{REC} \quad \top[receive\ m_2]_A Receive(A, m_2) \wedge Has(A, m_2) \quad (8)$$

$$\textbf{AA1}, \textbf{AA4} \quad \top[dec\ m_2, k_A^{-1}/send\ m_3]_A\ Decrypt(A, m_2) < Send(A, m_3) \quad (9)$$

$$\textbf{1}, \textbf{8}, \textbf{9}, \textbf{DEC}\ Has(a, m_2) \wedge Has(a, k_A^{-1}) \wedge Honest(\hat{A}) \supset Contains(N_a, m_2) \wedge Contains(T_b, m_2) \quad (10)$$

$$\textbf{7}, \textbf{10}, \textbf{FS2} \quad FirstSend(A, N_a, m_1) \wedge Contains(N_a, m_2) \wedge Honest(\hat{B}) \supset (Receive(B, m_1) < Send(B, m_2)) \wedge FirstSend(B, T_b, m_2) \quad (11)$$

$$\textbf{5}, \textbf{6}, \textbf{10}, \textbf{TT3} \quad FirstSend(A, T_a, m_1) \wedge Receive(B, m_1) \wedge Contains(T_a, m_1) \supset (Send(A, m_1) < Receive(B, m_1)) \quad (12)$$

$$\textbf{7}, \textbf{10}, \textbf{11}, \textbf{TT3} \quad FirstSend(B, T_b, m_2) \wedge Receive(A, m_2) \wedge Contains(T_b, m_2) \supset (Send(B, m_2) < Receive(A, m_2)) \quad (13)$$

$$\textbf{AA4}, \textbf{P1}, \textbf{G4}, \textbf{HON}_Q \quad (\hat{A}, \hat{B})[Init_{X.509}]_A \supset (Receive(A, m_2) < Send(A, m_3)) \quad (14)$$

$$\textbf{11}, \textbf{12}, \textbf{13}, \textbf{14}, \textbf{HON}_Q \quad (\hat{A}, \hat{B})[Init_{X.509}]_A Honest(\hat{B}) \supset \phi_{AUTH}(\hat{A}) \quad (15)$$

Proposition 2. *Let* C *be initial configuration of improved CCITTX.509,* $\hat{A}$ *and* $\hat{B}$ *be the principal of the initiator and responder roles respectively. If principal* $\hat{A}, \hat{B} \in Honest(C)$*, then* $\phi_{AUTH}(\hat{B})$ *is true.*

The conclusion of Proposition 2 is true, and the proof process is similar to Proposition 1, which is no longer repeated.

According to the proof of proposition 1 and 2, the improved CCITTX.509 protocol satisfies authentication.

5.2 Secrecy Analysis

Proposition 3. *Let* C *be initial configuration of improved CCITTX.509,* $\hat{A}$ *and* $\hat{B}$ *be the principal of the initiator and responder roles respectively. If principal* $\hat{A}, \hat{B} \in Honest(C)$*, then* $\phi_{SEC}(\hat{A})$ *is true.*

Proof 2.

$$\textbf{AM1}, \textbf{AM2} \quad (\hat{A}, \hat{B})[\,]_A Has(A, A) \wedge Has(A, B) \wedge Has(A, k_A^{-1}) \wedge Has(A, k_B) \quad (16)$$

$$\textbf{AN2}, \textbf{AN2} \quad \top[new\ n_a]_A Has(A, N_a) \wedge Fresh(A, N_a) \quad (17)$$

$$\textbf{AN2}, \textbf{AN3} \quad \top[new\ Y_a]_A Has(A, Y_a) \wedge Fresh(A, Y_a) \quad (18)$$

$$\textbf{AA1}, \textbf{STR1} \quad \top[t_2 := \{A, T_a, N_a, B, t_1, Y_a\}]_A\ Tuple(A, t_1) \wedge Contains(Y_a, t_2) \quad (19)$$

$$\textbf{STR2}, \textbf{STR4} \quad \top[m_1 := enc\ t_2, k_B]_A Encrypt(A, t_2) \wedge Contains(t_2, m_1) \supset Contains(N_a, m_1) \wedge Contains(Y_a, m_1) \quad (20)$$

$$\textbf{20}, \textbf{FS1}, \textbf{AA3} \quad Fresh(A, N_a) \wedge Contains(N_a, m_1)\ [send\ m_1]_A FirstSend(A, N_a, m_1) \quad (21)$$

$$\textbf{20}, \textbf{FS1}, \textbf{AA3} \quad Fresh(A, Y_a) \wedge Contains(Y_a, m_1)\ [send\ m_1]_A FirstSend(A, Y_a, m_1) \quad (22)$$

$$\textbf{AA1}, \textbf{REC} \quad \top[receive\ m_2]_A Receive(A, m_2) \wedge Has(A, m_2) \quad (23)$$

$$\textbf{AA1}, \textbf{DEC} \quad Has(A, k_A^{-1})[t_3 := dec\ m_2, k_A^{-1}]_A\ Decrypt(A, m_2) \supset Contains(N_a, m_2) \wedge Contains(Y_b, m_2) \wedge Has(A, Y_b) \wedge Has(A, N_a) \quad (24)$$

$$\textbf{21}, \textbf{22}, \textbf{FS2} \quad FirstSend(A, N_a, m_1) \wedge Contains(N_a, m_2) \wedge Honest(\hat{B}) \supset (Receive(B, m_1) < Send(B, m_2)) \wedge FirstSend(B, Y_b, m_2) \quad (25)$$

$$\textbf{HON}_Q \quad Honest(\hat{B})[\,]_A Has(B, A) \wedge Has(B, B) \wedge Has(B, k_B^{-1}) \wedge Has(B, k_A) \quad (26)$$

$$\textbf{24}, \textbf{25}, \textbf{DEC} \quad Receive(B, m_1) \wedge Has(B, k_B^{-1}) \wedge Contains(Y_a, m_1) \supset Has(B, Y_a) \quad (27)$$

$$\textbf{23}, \textbf{24}, \textbf{ENC} \quad FirstSend(B, Y_b, m_2) \supset Has(B, Y_b) \quad (28)$$

$$\textbf{18}, \textbf{27}, \textbf{P1} \quad (\hat{A}, \hat{B})[Init_{X.509}]_A(\exists Z.Has(Z, Y_a) \supset (Z = A \vee Z = B)) \quad (29)$$

$$\textbf{18}, \textbf{28}, \textbf{P1} \quad (\hat{A}, \hat{B})[Init_{X.509}]_A(\exists Z.Has(Z, Y_b) \supset (Z = A \vee Z = B)) \quad (30)$$

$$\textbf{29, 30, HON}_Q \quad (\hat{A},\hat{B})[Init_{X.509}]_A\phi_{SEC}(\hat{A}) \tag{31}$$

Proposition 4. *Let C be initial configuration of improved CCITTX.509, $\hat{A}$ and $\hat{B}$ be the principal of the initiator and responder roles respectively. If principal $\hat{A},\hat{B} \in Honest(C)$, then $\phi_{SEC}(\hat{B})$ is true.*

The same principle can prove the correctness of the conclusion of proposition 4, which is omitted here.

Propositions 3 and 4 verify that the improved CCITTX.509 protocol can satisfy the confidentiality.

5.3 Data Integrity Analysis

Proposition 5. *Let C be initial configuration of improved CCITTX.509, $\hat{A}$ and $\hat{B}$ be the principal of the initiator and responder roles respectively. If principal $\hat{B} \in Honest(C)$, then $\phi_{INTE}(\hat{A})$ is true.*

Proof 3.

$$\textbf{AM1, AM2} \quad (\hat{A},\hat{B})[\,]_A Has(A,A) \wedge Has(A,B) \wedge Has(A,k_A^{-1}) \wedge Has(A,k_B) \tag{32}$$

$$\textbf{AM1, AM2} \quad Honesty[\,]_B Has(B,A) \wedge Has(B,B) \wedge Has(B,k_B^{-1}) \wedge Has(B,k_A) \tag{33}$$

$$\textbf{AN1, AN3} \quad \top[new\ N_a]_A Has(A,N_a) \wedge Fresh(A,N_a) \tag{34}$$

$$\textbf{AA1STR2, STR4} \quad \top[m_1 := enc\ t_2, k_B]_A\ Encrypt(A,t_2) \wedge Contains(t_2,m_1) \supset Contains(N_a,m_1) \tag{35}$$

$$\textbf{36, FS1, AA3} \quad Fresh(A,N_a) \wedge Contains(N_a,m_1)\ [send\ m_1]_A FirstSend(A,N_a,m_1) \tag{36}$$

$$\textbf{AA1, REC} \quad \top[receive\ m_2]_A Receive(A,m_2) \wedge Has(A,m_2) \tag{37}$$

$$\textbf{AA1, DEC} \quad Has(A,k_A^{-1})[t_3 := dec\ m_2, k_A^{-1}]_A\ Decrypt(A,m_2) \supset Contains(N_a,m_2) \tag{38}$$

$$\textbf{37, 39, FS2, HON}_Q \quad FirstSend(A,N_a,m_1) \wedge Contains(N_a,m_2) \wedge Honest(\hat{B}) \supset (Receive(B,m_1) < Send(B,m_2)) \wedge FirstSend(B,Y_b,m_2) \tag{39}$$

$$\textbf{AA1, AA4} \quad \top[receive\ m_2/t_3 := dec\ m_2, k_A^{-1}]_A\ Receive(A,m_2) < Decrypt(A,m_2) \supset Has(A,t_3) \wedge Contains(m_2,t_3) \tag{40}$$

$$\textbf{AA1, STR1} \quad \top[match\ t_3, \{B,T_b,N_b,A,N_a,t_4,Y_b\}]_A \wedge Match(A,t_3) \supset Has(A,t_4) \wedge Contains(t_4,t_3) \tag{41}$$

$$\textbf{AA1, HON}_Q \quad Honest(\hat{B})[verify\ t_4, X_b, k_b]_A\ Verify(A,t_4) \supset Sign(B,t_4 := |X_b|_{k_B^{-1}}) \tag{42}$$

$$\textbf{41, 42, STR4} \quad Contains(m_2,t_3) \wedge Contains(t_3,t_4) \supset Contains(m_2,t_4) \tag{43}$$

$$\textbf{40, 43, 44, HON}_Q \quad (\hat{A},\hat{B})[Init_{X.509}]_A Honesty(\hat{B}) \supset \exists Z.(Sign(Z,|X_b|_{k_Z^{-1}}) \wedge Send(Z,m_2) \wedge Contains(|X_b|_{k_Z^{-1}},m_2)) \supset (Z = B) \tag{44}$$

Proposition 6. *Let C be initial configuration of improved CCITTX.509, $\hat{A}$ and $\hat{B}$ be the principal of the initiator and responder roles respectively. If principal $\hat{A} \in Honest(C)$, then $\phi_{INTE}(\hat{B})$ is true.*

The same principle can prove the correctness of the conclusion of proposition 6, which is omitted here.

Propositions 5 and 6 verify that the improved CCITTX.509 protocol can guarantee the integrity of sum.

The proof of propositions 1 to 6 shows that the improved CCITTX.509 protocol can meet the security attribute design goals of authentication, secrecy and data integrity.

5.4 Comparison with the Traditional Method

In traditional methods used in the analysis of CR [8], Otway-Rees [14] and NSL [22] based on PCL, the analysis of action sequence of protocol principals is mainly based on the judgment of freshness of random value N_a and N_b. The main inference rules used in protocol analysis are **FS1**, **FS2**, **P1** and **P2**. In order to illustrate the validity of the principal action sequence judgment rules based on the timestamp mechanism, table 1 gives a detailed comparison with the traditional methods in judging parameters and inference rules used in a challenge response round of the protocol, as well as the value range of proving steps. Because of the rigor and intuitiveness of the proof process, the value range given in Table 1 are only steps to reasonably prove the action sequence of the protocol, and the value range is not very accurate, which is only a reference for the comparison of the complexity of two analysis methods.

From the comparison results of the two methods in Table 1, in a challenge response round of the protocol, new method only needs about 5 to 10 steps to determine the action sequence of the protocol principals. Compared with the traditional method, it simplifies the steps of protocol analysis and effectively reduces the complexity of protocol analysis. The improvement and application of **STR1, STR2, STR3** and **STR4** in message subitem relationship make protocol analysis more scientific and rigorous.

6 Conclusions

Formal analysis process and results of the improved CCITTX.509 protocol using the extended PCL show that the inference rules of the relations among message subterms make the protocol analysis process more rigorous and formalized. Compared with the methods used in literature 5, the logical inference rules based on timestamp

Table 1: Comparison of two methods in a challenge response protocol round

Method	Protocol	Parameter	Inference Rules	Steps(n)
Traditional Method	$Otway-Rees$ CR, NSL	N_a, N_b	**FS1**, **FS2** **P1**, **P2**	$8 \leq n \leq 15$
Methods in this paper	$CCITTX.509$	T_a, T_b	**FS1**, **FS2** **TT1**, **TT2**, **TT3** **STR1**, **STR2** **STR3**, **STR4**	$5 \leq n \leq 10$

mechanism can greatly simplify the steps of protocol authentication analysis and effectively reduce the complexity of protocol analysis. Logical inference rules based on timestamp mechanism further improve the theoretical system of PCL. This method can be used to effectively analyze the authentication objectives of security protocols designed based on timestamp mechanism. In addition, according to the formal analysis process of CCITTX.509 protocol, it is obvious that propositional hypothesis analysis method further standardizes the PCL formal analysis method, making the protocol analysis process more intuitive and clear. The description method of security protocol based on protocol thread programming language and logical inference system based on the behavior assertion of protocol principal make PCL more formal and logical than other formal analysis methods [13, 17, 24].

Although PCL is highly formal in protocol description and logical inference system, the process of protocol analysis is relatively simple and intuitive. However, the assumption of honest rules makes it impossible for PCL to analyze attack types from within the protocol [6, 8]. Meanwhile, the language description system of PCL is not perfect and the standard definition of message algebraic space is not in place yet. These defects limit the modeling and analysis ability of PCL. Improving the message algebraic space theory of PCL and enhancing the ability of protocol description and analysis will be the main research goal in the next stage.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (61902140 and 61300048), Science Foundation for The Excellent Youth Scholars of Anhui University (gxyq2017154), Anhui Provincial Natural Science Foundation (1608085MF159 and 1908085QF288) and in part by Natural Science Foundation of Anhui University (KJ2014A231, KJ2015A315,KJ2018A0678, KJ2018A396). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] T. Y. Chang, M. S. Hwang, and C. C. Yang, "Password authenticated key exchange and protected password change protocols," *Symmetry*, vol. 9, no. 8, pp. 134, 2017.

[2] X. Chen and H. Deng, "Analysis of cryptographic protocol by dynamic epistemic logic," *IEEE Access*, vol. 7, pp. 29981–29988, 2019.

[3] V. Cheval, V. Cortier, and B. Warinschi, "Secure composition of pkis with public key protocols," in *IEEE 30th Computer Security Foundations Symposium (CSF'17)*, pp. 144–158, Aug. 2017.

[4] S. F. Chiou, H. T. Pan, E. F. Cahyadi, and M. S. Hwang, "Cryptanalysis of the mutual authentication and key agreement protocol with smart cards for wireless communications," *International Journal Network Security*, vol. 21, no. 1, pp. 100–104, 2019.

[5] N. Chumakova, V. Olenev, and I. Lavrovskaya, "Conformance testing of the STP-ISS protocol implementation by means of temporal logic," in *The 21st Conference of Open Innovations Association*, pp. 71–78, 2017.

[6] C. J. F. Cremers, "On the protocol composition logic PCL," in *Proceedings of ACM Symposium on Information, Computer and Communications Security*, pp. 66–76, 2008.

[7] A. Datta, A. Derek, J. C. Mitchell, and D. Pavlovic, "A derivation system for security protocols and its logical formalization," in *The 16th IEEE Computer Security Foundations Workshop*, pp. 109–125, 2003.

[8] A. Datta, A. Derek, J. C. Mitchell, and A. Roy, "Protocol composition logic (PCL)," *Electronic Notes in Theoretical Computer Science*, vol. 172, pp. 311–358, 2007.

[9] N. A. Durgin, J. C. Mitchell, and D. Pavlovic, "A compositional logic for proving security properties of protocols," *Journal of Computer Security*, vol. 11, no. 4, pp. 677–722, 2003.

[10] T. Feng, Y. Yi, and J. Ma, "Secure authenticated key agreement protocol for wmen based on protocol composition logic," in *IEEE 3rd International Conference on Communication Software and Networks*, pp. 284–288, May 2011.

[11] K. Gaarder and E. Snekkenes, "Applying a formal analysis technique to the CCITT X.509 strong two-way authentication protocol," *Journal of Cryptology*, vol. 3, no. 2, pp. 81–98, 1991.

[12] C. I'Anson and C. Mitchell, "Security defects in ccitt recommendation x.509: The directory authentication framework," *Computer Communication Review (CCR'90)*, vol. 20, pp. 30–34, Apr. 1990.

[13] H. Jiang, G. Zhang, and J. Fan, "Structure analysis and generation of X.509 digital certificate based on national secret," in *Journal of Physics: Conference Series*, vol. 1187, pp. 042067, Apr 2019.

[14] J. F. Ma, L. F. Lu, X. D. Duan, "Improvement and formal proof on protocol Otway-Rees," *Journal on Commuications*, vol. 33, pp. 250–254, Sep. 2012.

[15] L. Lu and J. Ma, "Formal analysis model of security protocol based on PCL," in *International Conference on Computer Application and System Modeling (ICCASM'10)*, Oct. 2010. DOI: 10.1109/ICCASM.2010.5620624.

[16] C. H. Ling, S. M. Chen, and M. S. Hwang, "Cryptanalysis of Tseng-Wu group key exchange protocol," *International Journal Network Security*, vol. 18, no. 3, pp. 590–593, 2016.

[17] J. F. Liu and M. T. Zhou, "Analysis of X.509 authentication protocol via authentication test," *Computer Engineering and Applications*, vol. 08, pp. 23–25, Aug. 2006.

[18] P. Liu and P. Zhou, "Formal analysis of improved EAP-AKA based on protocol composition logic," in *The 2nd International Conference on Future Computer and Communication*, May 2010. DOI: 10.1109/ICFCC.2010.5497694.

[19] S. Mendes and C. Huitema, "A new approach to the X.509 framework: Allowing a global authentication infrastructure without a global trust model," in *Proceedings of the Symposium on Network and Distributed System Security*, pp. 172–189, Feb. 1995.

[20] B. Meng, J. T. Lu, D. J. Wang, and X. D. He, "Survey of security analysis of security protocol implementations," *Journal of Shandong University (Natural Science)*, vol. 53, no. 1, pp. 1–18, 2018.

[21] A. Roy, A. Datta, A. Derek, J. C. Mitchell, and J. P. Seifert, "Secrecy analysis in protocol composition logic," in *Advances in Computer Science*, pp. 197–213, 2006.

[22] J. Song, M. Xiao, K. Yang, X. Wang, and X. Zhong, "LoET-E: A refined theory for proving security properties of cryptographic protocols," *IEEE Access*, vol. 7, pp. 59871–59883, 2019.

[23] C. H. Wei, M. S. Hwang, and A. Yeh-Hao Chin, "A secure privacy and authentication protocol for passive RFID tags," *International Journal of Mobile Communications* , vol. 15, no. 3, pp. 266–277, 2017.

[24] L. Yu, Y. Y. Guo, and M. M. Jiang, "Improvement of strand space theory for application of minimal element method," *Quarterly Journal of Indian Pulp and Paper Technical Association*, vol. 30, pp. 94–105, Mar. 2018.

[25] Z. You, J. T. Li, and X. Xie, "Extension and application of protocol composition logic," in *The 2nd International Conference on Computer Engineering and Technology*, Apr. 2010. DOI: 10.1109/ICCET.2010.5485720.

Biography

Lei Yu was born in 1978. He received the MS an BS degree in computer science and technology from Huaibei Normal University of China. Currently, he is an assistant professor and MS supervisor in the school of computer science and technology, Huaibei Normal University, China. His major research interests include cryptography and information security. He has published many papers in related journals.

Zhi-Yao Yang was born in 1995. He received the B.S.degree from Information College of Huaibei Normal University in 2017. He has been with the School of Mathematical Science, Huaibei Normal Universtiy, where he is a postgraduate student. His research interests include cryptography and information theory.

Ze-Peng Zhuo was born in 1978. He received the M.S.degree from Huaibei Normal University in 2007, and the Ph.D. degree from Xidian University in 2012. Since 2002, he has been with the School of Mathematical Science, Huaibei Normal Universtiy, where he is currently a professor. His research interests include cryptography and information theory.

Personalized K-In&Out-Degree Anonymity Method for Large-scale Social Networks Based on Hierarchical Community Structure

XiaoLin Zhang[1], Jiao Liu[1], HongJing Bi[2], Jian Li[1], and YongPing Wang[1]
(Corresponding author: XiaoLin Zhang)
School of Information Engineering, Inner Mongolia University of Science and Technology [1]
Baotou, Inner Mongolia 014010, China
Department of Computer Science, Tangshan Normal University [2]
Tangshan, Hebei 063000, China
(Email: 2784899426@qq.com)
(Received Oct. 11, 2019; Revised and Accepted Jan. 15, 2020; First Online Feb. 5, 2020)

Abstract

The existing social network privacy protection technologies have the problems of neglecting the protection of the community structure and failing to meet users' different privacy protection requirements when processing the large-scale social network directed graphs. A personalized K-In&Out-Degree anonymity (PKIODA) algorithm based on hierarchical community structure is proposed. The algorithm divides the community based on the hierarchical community structure. According to the different user privacy protection levels *Lv0*∼*Lv3*, the grouping and the anonymity sequence are distributed, and the nodes are added in parallel to realize the anonymity. Based on GraphX to transfer information between nodes, the virtual node pairs are merged and deleted according to the hierarchical community entropy change to reduce the information loss. The experimental results show that the PKIODA algorithm improves the efficiency of processing large-scale social network directed graph data, ensures the high availability of community structure analysis during data release, and satisfies the privacy protection requirements of different users.

Keywords: GraphX; Hierarchical Community Structure; K-in&out-degree Anonymity; Personalized; Social Network Directed Graph

1 Introduction

With the development of social networks, the number of network users of various social APPs has been increasing, and the links between users have become closer. Large-scale users generate a large amount of social network data in the process of using social networks, and there is still a certain network structure in the actual social network directed graph. Some users with the same hobbies, similar attributes or frequent contacts will form a specific group, that is, the community structure of the social network. Therefore, it has important research significance for the analysis of community structure when large-scale social network directed graphs are released. The information in the social network often involves personal privacy. If the network data is directly distributed to a third party, it is easy to cause the leakage of sensitive information of the user. Therefore, researchers propose different privacy protection models for different privacy information, such as modifying models [6, 11, 13], clustering generalization models [21], data perturbation models [16], differential privacy model [20] and so on. However, in actual social networks, different users have different needs for privacy protection, and some users do not want to hide their identity [15]. At present, most privacy protection technologies only deal with social network undirected graphs. Although the information loss of graph modification is reduced, the protection of the original graph's community structure is neglected, and the user's personalized privacy protection requirements cannot be met. Personalized privacy protection for large-scale social network directed graphs based on GraphX distribution parallelism, and the availability of community structure analysis during data release during anonymity.

The main work and contributions are as follows:

1) A new attack model is proposed for large-scale social network directed graphs. On the basis of protecting the community structure, a personalized K-in&out-degree anonymity model based on hierarchical community structure is proposed, and the user's needs are set to four privacy protection levels of *Lv0*∼*Lv3*;

2) Propose a large-scale social network personalized K-in&out-degree anonymity (PKIODA) algo-

rithm based on hierarchical community structure, which satisfies the personalized K-in&out-degree anonymity and effectively protects the community structure of large-scale social network directed graphs;

3) Experiments on real datasets prove that PKIODA algorithm improves the processing efficiency of large-scale social network directed graph privacy protection, and ensures the high availability of community structure analysis when data is released;

The organization of the rest of this paper is as follows. In Section 2, we describe the related work to social network privacy protection and community structure protection. Section 3 introduces relevant preliminary knowledge and gives definitions of research questions. Section 4 introduces the PKIODA algorithm that protects the community structure. Section 5 is based on real social network datasets and experimentally tested in terms of algorithm performance, information loss, and data availability. Section 6 concludes the full text.

2 Related Work

At present, social network privacy protection information mainly includes node privacy, edge privacy and graph structure privacy [17]. To deal with different social network privacy protection information, researchers have proposed a variety of social network privacy protection technologies.

Yuan and Zhou*et al.* [1, 18] proposed a K-degree-L-diversity anonymity model for social network undirected graphs with attribute labels, and implements attribute anonymity by adding and deleting edges and adding noise nodes. Casas *et al.* [5] proposed the UMGA algorithm, the greedy algorithm and the exhaustive method are used to generate the anonymity sequence. The nondirectional graph is modified by the random edge selection and the neighbor central edge selection method to realize the K-degree anonymity. Kiabod *et al.* [7] introduced a time-saving k-degree anonymization method in social network (TSRAM) that anonymizes the social network graph without having to rescan the datasets for different levels of anonymity. Li *et al.* [8] proposed a novel Graph clustering framework based on potential game optimization (GLEAM) for parallel graph clustering. Based on (α, k)-anonymity, a personalized (α, k)-anonymity model [9] is proposed to achieve K-degree anonymity according to different privacy protection requirements of users. Campan *et al.* [2] used the community partitioning algorithm based on graph segmentation theory. By calculating the Laplacian matrix, the spectral constraint conditions are set and the constraints of adding and deleting edges are calculated. Kumar *et al.* [14] used the upper approximation concept of rough sets, divided the community and performed anonymity, and maintained the community structure nature of the graph before and after anonymity. Zhang [19] proposed the application of artificial intelligence in computer safety protection by combining artificial intelligence with computer network security.

Aiming at the current social network privacy protection methods, there are problems in dealing with large-scale social network directed graph data with low efficiency, neglecting community structure protection, and failing to satisfy users' personalized privacy protection requirements. A large-scale social network personalized K-in&out-degree anonymity (PKIODA) algorithm based on hierarchical community structure is proposed. The method improves the efficiency of processing large-scale dynamic social networks directed graphs, and at the same time personalizes anonymous social network directed graph to ensure the availability of community structure analysis when data is published.

3 Preliminary Knowledge and Problem Definition

The social network is represented as a personalized directed graph G=(V, E), where V(G) and E(G) respectively represent the node set and edge set of graph G. The edge ⟨u,v⟩ indicates that one edge from the node u points to node v. The edge ⟨u,v⟩ is called the out-edge of u and the in-edge of v. The number of the in-edge of node u is the in_deg of u, which is denoted as d_{in}(u); The number of out-edge of node u is the out_deg of u, which is denoted as d_{out}(u); The in&out-degree of node u is represented by (d_{in}(u), d_{out}(u)). The privacy protection level of node is *Lv0*∼*Lv3*.

3.1 Community Detection Based on Hierarchical Community Structure

The community structure is a group of nodes that are closely connected within the community and have close connections between communities. Clauset *et al.* [3] used hierarchical community trees to reflect the hierarchical community structure of social networks. We aim to improve the community detection algorithm based on hierarchical community structure [3].

Definition 1. *(Directed graph hierarchy community tree H_G) Given a social network directed graph G, the hierarchical structure graph (HRG) is used to represent the community structure of the directed graph G, which is recorded as a directed graph hierarchical community tree(H_G). The leaf nodes of the H_G represent the nodes in the directed graph G, and each internal node r in the H_G represents the connection probability P_r. T_r is a subtree of an internal node r of the H_G. The connection probability of the leaf node in the left subtree T_r^L and the leaf node of the right subtree T_r^R in the directed graph G is represented by P_r, reflecting the strength of the connection between the left subtree and the right subtree leaf node. The larger the P_r is, the closer the connection is. The P_r is calculated*

as follows:

$$P_r = \frac{|E_r|}{\left|T_r^R\right| \cdot \left|T_r^L\right| \cdot 2} \quad (1)$$

where $|E_r|$ *is the number of edges* $\langle v_i, v_j \rangle \in E$ *with* $v_i \in T_r^L$ *and* $v_j \in T_r^R$*, and* T_r^L (T_r^R) *is the number of vertices in internal node r's left (right) subtree.*

The HRG of the social network directed graph G is not unique, so it is crucial to select the optimal HRG and then obtain the community structure of the social network directed graph according to H_G. The likelihood function L can be used for the different HRG to evaluate their suitability for the social network directed graph G.

Definition 2. *(Likelihood function L) The likelihood function L is a posterior function of the* H_G *generated by the social network directed graph G. The HRG tree having the largest L value is selected as the* H_G *of the directed graph G. The L is calculated as follows:*

$$L(H_G) = \prod\nolimits_{r \in H_G} \left[{P_r}^{P_r} \left(1 - P_r\right)^{1 - P_r} \right]^{\left|T_r^R\right| \cdot \left|T_r^L\right| \cdot 2} \quad (2)$$

Figure 1(a) shows the social network directed graph G_0, and Figure 2 shows the two possible HRGs of G_0. It is calculated that L=4.639^{e-7} of Figure 1(a), L=6.465^{e-5} of Figure 1(b), and the L value of Figure 1(b) is the largest. Therefore, Figure 1(b) is selected as H_{G0} of the directed graph G_0, and the community result is shown in Figure 1(b), in which nodes A, B, and C belong to community 1, nodes D, E, and F belong to community 2.

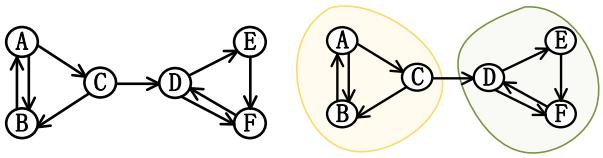


Figure 1: Community detection of directed graph G_0

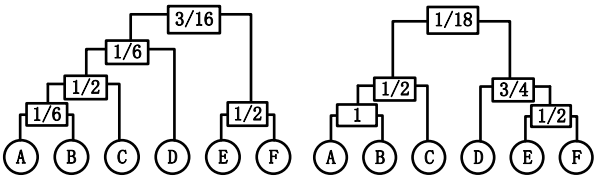


Figure 2: Hierarchical community structure

3.2 Anonymity Model of Personalized K-in&out-degree for Large-scale Social Networks based on Hierarchical Community Structure

An attacker can identify the target node based on the degree information of the node in the social network. Aiming at the social network directed graph, an in&out-degree attack model is proposed.

Definition 3. *(In&out-degree attack model) Suppose the attacker knows the in_deg(out_deg) of the target node, or knows both the in_deg and the out_deg. With the background knowledge, the attacker can uniquely identify the target node, which is called in&out-degree attack.*

As shown in Figure 3, it is assumed that the attacker knows that the degree of the target node is 2, and Alice, Kayla, and Bob can be identified. However, the attacker not only knows the degree of the target node but also knows that the out_deg of the target node is 2, so that the attacker an uniquely identify the target node is Kayla, the privacy information of the Kayla node is leaked.

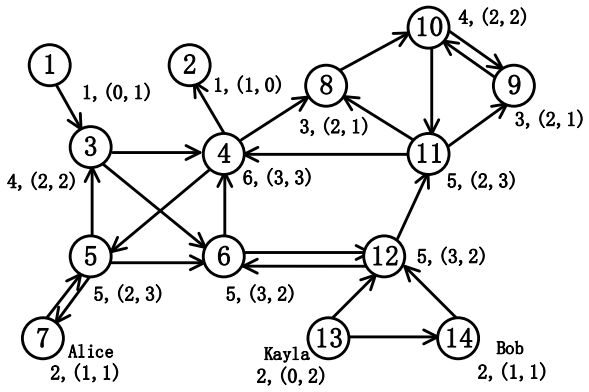


Figure 3: Social network directed graph G

Definition 4. *(K-in&out-degree anonymity) Given a social network directed graph G=(V, E) and the anonymous parameter K. For any node* $v \in V(G)$ *in the directed graph, there are m* $(m \geq k\text{-}1)$ *other nodes which are the same number of in_deg and out_deg of the node v, i.e.* $d_{in}(v) = d_{in}(v_i)$*,* $d_{out}(v) = d_{out}(v_i)$ $(1 \leq i \leq m)$*. Then the directed graph G is the K-in&out-degree anonymity graph.*

Definition 5. *(Personalized K-in&out-degree anonymity) Personalized K-in&out-degree anonymity based on user privacy protection needs, thereby the user's requirements are set to four privacy protection levels:* Lv0∼Lv3.

1) Lv0: *The user does not require privacy protection for the node;*

2) Lv1: *The user only requires privacy protection for the out_deg information of the node, so that the probability that the attacker identifies the target node by the degree of the node is not more than 1/K;*

3) Lv2: *The user only requires privacy protection for the in_deg information of the node, so that the probability of the attacker identifies the target node by the in_deg of the node is not more than 1/K;*

4) Lv3: *The user requires privacy protection for the in_deg and out_deg information of the node, so that the probability of the attacker identifies the target*

node through the in_deg and the out_deg of the node is not more than 1/K.

Definition 6. *(Large-scale social network personalized K-in&out-degree anonymity model based on hierarchical community structure) Given the social network directed graph G=(V,E) and the anonymous parameter K.*

1) *Dividing the community based on the hierarchical community structure, and determining the community to which the nodes belong in the original social network directed graph G;*

2) *Personalized K-in&out-degree anonymity according to user privacy protection level* Lv0∼Lv3. *Sorting, grouping and anonymizing the degree sequence of the nodes in the social network directed graph G to obtain the anonymity sequence;*

3) *Anonymous graphs are constructed in parallel according to the anonymity sequence distribution. The information between nodes is transmitted based on the GraphX, and the virtual nodes are merged and deleted multiple times to improve the availability of community structure analysis during data release.*

Anonymous social network directed graph G*=(V*, E*) satisfying these three conditions conforms to the large-scale social network personalized K-in&out-degree anonymity model based on hierarchical community structure.

4 PKIODA

Personalized K-in&out-degree anonymity(PKIODA) algorithm for large-scale social networks based on hierarchical community structure combines Spark, a fast and versatile computing engine designed for large-scale data processing. The algorithm is executed in a distributed parallel environment to implement a transparent privacy enforcement strategy for large-scale social network data.

4.1 Community Detection Algorithm based on Hierarchical Community Structure

The community detection algorithm based on hierarchical community structure first obtains the HRG of the social network directed graph, and then uses the Markov Monte Carlo sampling method [12] to converge to select the better hierarchical random graph. Then obtain the social network graph community according to H_G. The construst HRG algorithm is detailed as Algorithm 1.

The Construst_HRG(G,ε_1)) algorithm randomly selects a HRG to initialize the Markov chain, and perform multiple iterations until the Markov chain converges (lines 3-12). Let the current state of the Markov chain be T_{t-1}, in which an internal node n is randomly selected. The next state T' is generated by replacing the adjacent HRG of the node n. Due to randomness, T' is not unique, and a better T' is selected by calculating the value of the likelihood function L. By comparing the error values of the likelihood function L before and after the exchange, the state exchange acceptance rate is set to $min\left(\frac{exp\left(\frac{\varepsilon_1}{2\triangle u}logL(T')\right)}{exp\left(\frac{\varepsilon_1}{2\triangle u}logL(T_{t-1})\right)},1\right)$ by means of the Markov Monte Carlo sampling method (Lines 6-10). At the end of the iteration, the Markov chain converges to obtain a better H_G. The better H_G generated by the directed graph G shown in Figure 3 is as shown in Figure 4(a), and the community detection result is as shown in Figure 4(b).

Algorithm 1 Construst_HRG(G,ε_1)

Input: Original graph G, differential privacy parameter ε_1

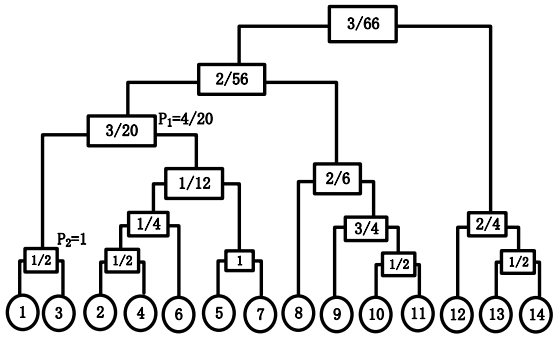
Output: H_G

1: Randomly generate an HRG of the original graph G according to the Markov chain, denoted as T_0;
2: t ← 1
3: **while** Markov chain is not converged **do**
4: Randomly select an internal node n in T_{t-1};
5: Generate HRG T' by randomly transforming adjacent subtrees of node n;
6: **if** the probability of transformation satisfies $min\left(\frac{exp\left(\frac{\varepsilon_1}{2\triangle u}logL(T')\right)}{exp\left(\frac{\varepsilon_1}{2\triangle u}logL(T_{t-1})\right)},1\right)$ **then**
7: T_t=T';
8: **else**
9: T_t=T_{t-1};
10: **end if**
11: t=t+1;
12: **end while**
13: Return while

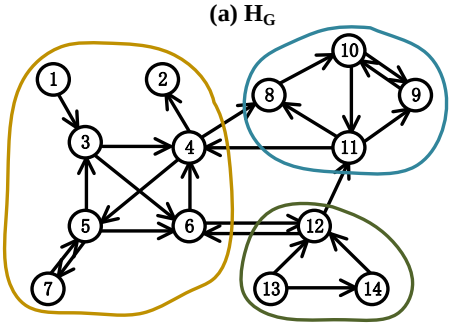
4.2 Community Detection Algorithm Based on Hierarchical Community Structure

Aiming at the social network directed graph, the Sequence Partition(G, k) algorithm is proposed, and the target degree goal of different levels is obtained according to the user privacy protection level. The partition is performed by judging the values of the SPC1 and SPC2. SPC1 represents the anonymity cost of the current element merging into the previous group, and SPC2 indicates the cost of forming a new group with the following K-1 elements. If SPC2 < SPC1, then the element forms a new group, otherwise it merges into the previous group.

For the nodes with privacy protection levels *Lv1* and *Lv2*, the target goal is the maximum value of all in_deg(out_deg) in the group, i.e., goal(in_deg/out_deg) =(max{in_deg/out_deg of all elements in the group}). The personalized sequence partition algorithm for nodes with privacy protection levels *Lv1* and *Lv2* is in Algorithm 2.



(a) H_G



(b) Community dection of G

Figure 4: Community detection based on hierarchical community structure

Algorithm 2 Sequence Partition_1(G,k,Lv)

Input: Original directed graph G, anonymous parameter k, privacy protection level Lv
Output: Anonymity sequence $\tilde{d}$

```
MF_Seq=[];
for each node do
    Insert in MF_Seq;
    count=MF_Seq_Value[node];
end for
if Lv=1 then
    Sort(MF_Seq) by out_deg;
else
    Sort(MF_Seq) by in_deg;
end if
last_partition_index=0;
for i=k to i+k do
    SPC1=MF_Seq[last_partition_index]-MF_Seq[i];
    SPC2=0;
    for j=i+1 to i+k do
        SPC1=SPC1+MF_Seq[i+1]-MF_Seq[j];
        SPC2=SPC2+MF_Seq[i]-MF_Seq[j-1];
    end for
    if SPC2 < SPC1 then
        last_partition_index=i;
        i=i+k;
    else
        i++;
    end if
end for
```

For the nodes with privacy protection level *Lv3*, the target goal is the maximum values of in_deg and out_deg in the group, goal(in_deg,out_deg)=(max{in_deg of all elements in the group}, max{out_deg of all elements in the group}). The personalized sequence partition algorithm for nodes with privacy protection level *Lv3* is in Algoruthm 3.

Algorithm 3 Sequence Partition_2(G,k)

Input: Original directed graph G, anonymous parameter k
Output: Anonymity sequence $\tilde{d}$

```
for v_i ∈ MF_Seq do
    for l=last to i-1 do
        goal_1(in_deg,out_deg)=max(in_deg[i],out_deg[i]);
    end for
    for m=i to i+k do
        goal_2(in_deg,out_deg)=max(in_deg[m+1],
        out_deg[m+1]);
        goal_3(in_deg,out_deg)=max(in_deg[m],
        out_deg[l]);
    end for
    SPC1=goal_1-MF_Seq[i], SPC2=0;
    for j=i+1 to i+k do
        SPC1=SPC1+goal_3-deg[j-1];
        SPC2=SPC2+goal_2-deg[j];
    end for
    if SPC2 < SPC1 then
        last_partition_index=i;
        i=i+k;
    else
        i++;
    end if
end for
```

For the original graph G shown in Figure 3, assumed that the privacy protection level of node 1-5 is *Lv3*(K=2). The partition anonymity is shown in Figure 5. Figure 5(a) shows the initial in&out-degree sequence d. The first two elements are first placed in a group, as shown in Figure 5(b). Judging the partition of the third element as shown in Figure 5(c). SPC1<SPC2, so the third element is merged with the first two elements. The result of partition is shown in Figure 5(d), and Figure 5(e) shows the anonymous sequence $\tilde{d}$ of Nodes 1-5.

4.3 Selecting the Virtual Node Pair Merge-Delete Algorithm

Assume that nodes 1-5 privacy protection level is *Lv3*, nodes 6-10 are *Lv2*, nodes 11-13 are *Lv1*, and node 14 is *Lv0*. The virtual nodes are added in parallel according to the anonymity sequence to obtain the anonymous directed graph G' shown in Figure 6. In order to reduce the information loss, the label of node is transmitted in parallel based on the GraphX, and the virtual node pairs are merged and deleted to improve the availability of data.

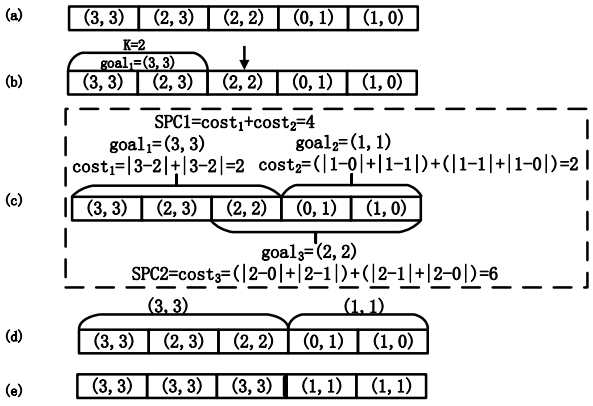


Figure 5: Partition anonymous process, (a) initial in&out-degree sequence d; (b) initial partition; (c) calculate the value of SPC1, SPC2; (d) partition results; (e) K-in&out-degree anonymous sequence $\tilde{d}$

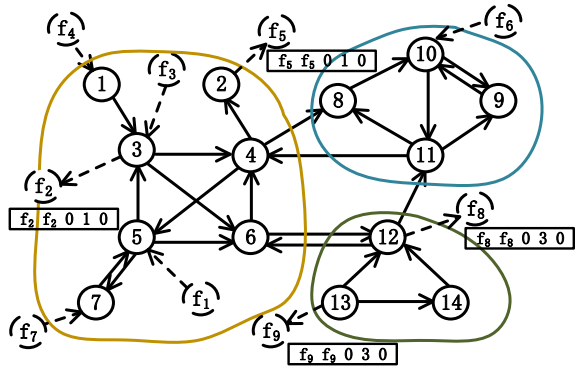


Figure 6: Anonymous social network directed graph G'

The data structure of the information transfer is represented by a five-tuple (dstid, srcid, hops, community, tags), which is called n-hops neighborhood table(HNT). dstid and srcid indicate the ID of the destination node and the source node, hops indicates the Superstep, community indicates the community of the source node, and tags=1 indicates that both the source node and the destination node are both virtual nodes. Each line of the HNT is an HNTE(n-hops neighborhood table entry).

The result of initializing the anonymous graph G' is shown in Figure 6(only shows partial virtual node initial HNTE). Initially, the node's dstid and srcid are the node ID, hops=0, tags=0. For example, HNTE of node f_2={f_2, f_2, 0, 1, 0}.

Definition 7. *(Directed graph hierarchy community entropy DGHCE) Use the directed graph hierarchy community entropy to quantify the effect of adding edge operations on the community hierarchy, denoted DGHCE(G, H_G). The DGHCE(G, H_G) is calculated as follows:*

$$DGHCE(G, H_G) = -\sum_{t=1}^{|V|-1} \frac{\left|T_r^R\right| \cdot \left|T_r^L\right| \cdot 2 \cdot P_r}{|E|} lb \frac{\left|T_r^R\right| \cdot \left|T_r^L\right| \cdot 2 \cdot P_r}{|E|} \quad (3)$$

Definition 8. *(Change value of DGHCE) Use the change value UL of the DGHCE to measure the information loss caused by the virtual node's edge operation added after the merge delete. The UL(G, G') is calculated as follows:*

$$UL(G, G') = \left| DGHCE(G, G') - DGHCE(G\ H_G^{'}) \right| \quad (4)$$

Definition 9. *(Virtual node pair merge delete condition VNMDC) Exists edge $\langle u, f_w \rangle$ and edge $\langle f_x, v \rangle$, virtual node pair (f_w, f_x) can be merged and deleted, if and only if $\forall$(f_w, f_x)$\in$ VirtualSet meets the following three conditions:*

1) f_w, $f_x \notin$ VirtualRDD;

2) $\langle u, v \rangle \notin$ EdgeRDD;

3) $u \neq v$.

Theorem 1. *For the virtual node pair (f_w, f_x), satisfying the VNMDC is a sufficient condition that (f_w, f_x) can merge and delete.*

Proof. As shown in Figure 7(a), in order to merge and delete virtual node pairs (f_w, f_x), need to delete edges $\langle u, f_w \rangle$, $\langle f_x, v \rangle$ and add edges $\langle u, v \rangle$. However, the edge $\langle u, v \rangle$ already exists in the directed graph G, so (f_w, f_x) cannot be merged and deleted. Similarly, as shown in Figure 7(b), the virtual nodes f_w and f_x are both connected to the node u, so (f_w, f_x) cannot be merged and deleted. Therefore, if and only if the virtual node pair (f_w, f_x) satisfies the VNMDC, (f_w, f_x) can be merge and delete as shown in Figure 7(c). □

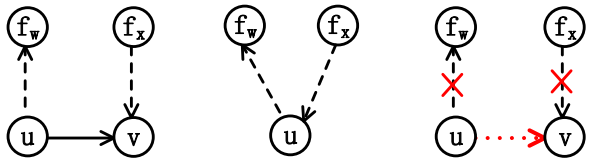


Figure 7: Virtual node pair merge delete condition

The virtual node pair set (VirtualSet) is obtained by the transfer of the information between the nodes, and judge whether $\forall$ (f_w,f_x)$\in$ VirtualSet satisfies the VNMDC. The virtual node pair satisfying the condition is put into the candidate virtual node set (CandidateSet). The virtual node pair select algorithm is in Algorithm 4.

If the number of virtual node pairs in the CandidateSet is more than one, for the virtual node pair of the same community (or are the different communities), the DGHCE value is calculated, and the virtual node pair with the small UL value is selected to be merged and deleted (Lines 4-7). If the number of virtual node pairs is 1, directly select the virtual node pair (f_w, f_x) (Lines 8-12).

Algorithm 4 Select Merge_Delete(CandidateSet)

Input: CandidateSet
Output:(f_w, f_x)

M = the number of same community in CandidateSet;
N = CandidateSet.size;
if N > 1 **then**
  if M > 1 || M==0 **then**
    (f_w, f_x)=the min(UL) from CandidateSet;
    return (f_w, f_x);
  end if
  if M==1 **then**
    return (f_w, f_x);
  end if
else
  return (f_w, f_x).
end if

Algorithm 5 PKIODA

Input: Original directed graph G, anonymous parameter k, privacy protection level Lv
Output: Anonymous directed graph G*

Construst_HRG algorithm generates H_G;
if $Lv \neq 0$ **then**
  if $Lv = 3$ **then**
    Sequence Partition_2(G,k) algorithm;
  else
    Sequence Partition_1(G,k,LV) algorithm;
  end if
end if
Add the virtual node based on the anonymous sequence to obtain an anonymous graph G';
Initialize G', CandidateSet=∅, VirtualRDD=∅;
for SuperStep=1 to 6 **do**
  Dst.Message ← Src.Message;
  for Message from Dst.HNTE **do**
    if Message.Tags==1 **then**
      Dst.VirtualSet ← Message;
    end if
  end for
  for Message from Dst.VirtualSet **do**
    f_w=Message.srcid, f_x=Message.disid;
    if (f_w, f_x) satify VNMDC **then**
      CandidateSet ← f_w, f_x;
    end if
  end for
  if CandidateSet.size > 0 **then**
    (f_w, f_x)=Select Merge_Delete(CandidateSet);
    G'.EdgeRDD.Remove ⟨u, f_w⟩;
    G'.EdgeRDD.Remove ⟨f_x, v⟩;
    G'.EdgeRDD.Add ⟨u, v⟩;
    VirtualRDD.Add (f_w, f_x);
    VoteToHalt (f_w, f_x);
  end if
end for
return G'.

4.4 PKIODA Algorithm

The large-scale social network personalized K-in&out-degree anonymity (PKIODA) algorithm based on hierarchical community structure is in Algorithm 5.

The specific steps of the PKIODA algorithm are as follows:

1) Superstep=0, the node initialization gets the initial EdgeRDD.

2) Superstep=1, the node receives its own 1-hop neighborhood information and generates a 1-hop neighbor-hood table. The first iteration flag is 0, VirtualSet=∅, CandidateSet=∅.

3) Superstep=2, the 2-hop neighborhood table is shown in Table 1 (only shows the HNTE of some virtual nodes), and check if there exist tags=1. Iteratively obtains VirtualSet={(f_2, f_1)}. Because virtual nodes f_2 and f_3 are connected to node 3 and do not satisfy the VNMDC, they cannot be merged and deleted, and CandidateSet=∅ is obtained.

4) Superstep=3, the 3-hop neighborhood table is shown in Table 2, and iteratively obtains VirtualSet={(f_2, f_1),(f_2, f_4)}.

Table 1: 2-hop neighborhood table

node	dstid	srcid	hops	community	tags
f_2	**f_2**	**f_3**	**2**	**1**	**1**
	f_2	1	2	1	0
	f_2	5	2	1	0
f_5	f_5	4	2	1	0
f_8	f_8	6	2	1	0
	f_8	13	2	3	0
	f_8	14	2	3	0

Table 2: 3-hop neighborhood table

node	dstid	srcid	hops	community	tags
f_2	**f_2**	**f_1**	**3**	**1**	**1**
f_2	**f_2**	**f_4**	**3**	**1**	**1**
	f_2	4	3	1	0
	f_2	7	3	1	0
f_5	f_5	3	3	1	0
	f_5	6	3	1	0
	f_5	11	3	2	0
f_8	f_8	3	3	1	0
	f_8	5	3	1	0

The virtual node pair satisfies the VNMDC, so CandidateSet={(f_2, f_1),(f_2, f_4)}. The virtual node pair (f_2, f_1),(f_2, f_4) performs Algorithm 4 to calculate the change of DGHCE, and selects the virtual node pair with small UL value to merge and delete. If merge delete (f_2, f_1), the connection probability P_1 of the H_G changes from

3/20 to 4/20; when merge deletes (f_2, f_4), P_2 changes from 1/2 to 1, as shown in the Figure 4(a).

The original graph DGHCE=3.57415, $DGHCE_{f2,f1}$ =3.44011, $UL_{f2,f1}$ =0.13404, $DGHCE_{f2,f4}$ =3.56307, $UL_{f2,f4}$ =0.01108. Choose the virtual node pairs (f_2, f_4) to selete and merge, and VirtualRDD={f_2, f_4}. The third iteration stops and the result is shown in Figure 8.

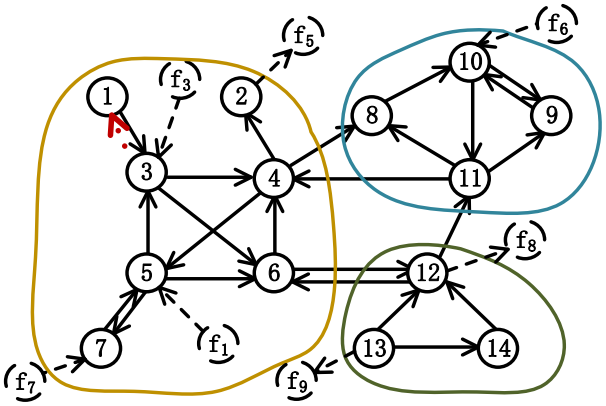


Figure 8: Results of the third iteration

The PKIODA algorithm iterates six times and the virtual node merge delete stops, resulting in an anonymous social network directed graph G* as shown in Figure 9.

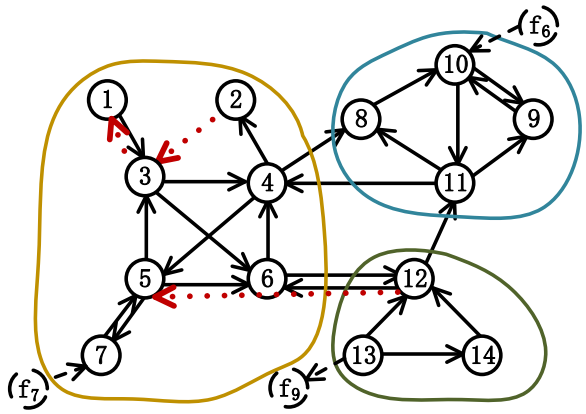


Figure 9: Anonymous social network directed graph G*

5 Experimental Analysis

The PKIODA algorithm is compared with the personalized K-degree-L-diversity anonymity (PKDLD) algorithm [4] and the personalized PPDP algorithm [10].The experiment used two real social network directed graph datasets Eu-Email and Epinions published by Stanford University.

5.1 Experimental Setup

Eu-Email network is generated using email data from large European research institutions. During the period from October 2003 to May 2005 (18 months), the dataset provided anonymous information about all incoming and outgoing e-mails from research institutions. Given a set of email messages, each node corresponds to an email address. If node i sends a message to j, the directed edge ⟨i, j⟩ is created between Nodes i and j.

Epinions network is a consumer online social network for commenting. Members of the site can decide whether to "trust each other". All trust relationships interact and form a trust network. The directed edge ⟨u, v⟩ indicates that the user u trusts the user v. Table 3 shows the statistics related to the dataset.

Table 3: The statistics related to the dataset

	Epinions	**Eu-Email**
Number of node	75879	265214
Number of edge	508837	420045
Max in_deg	3035	7631
Max out_deg	1801	930
ACC	0.1378	0.0671
Avg in&out-degree	6.71	1.58
Number of node(0,1)	18328	170768
Number of node(1,0)	11774	36922
Diameter	14	14

Distributed environment: GraphX, 15 computing nodes, CPU 1.8GHz, 16GB RAM, Hadoop 2.7.2, Spark 2.2.0, Scala 2.11.12.

5.2 Algorithm Performance Analysis

Figure 10 shows the running times of the algorithm with different anonymous parameter k. It can be seen from Figure 10 that as the value of k increases, the running times also increases. The running time of the PPDP algorithm is the smallest, the PKIODA algorithm is the second, and the PKDLD algorithm runs the longest. This is because the PKIODA algorithm firstly divides the original graph based on the hierarchical community structure, then groups the degree sequence according to the different privacy protection requirements of the user, and finally merges and deletes the virtual node pairs. As the value of k increases, more virtual node pairs need to be merged and deleted. Therefore, the execution time of the PKIODA algorithm is slightly larger than the PPDP algorithm, but the PKIODA algorithm does not run very long.

5.3 Information Loss Analysis

In order to measure the information loss during the anonymity of the algorithm, evaluate the change of the

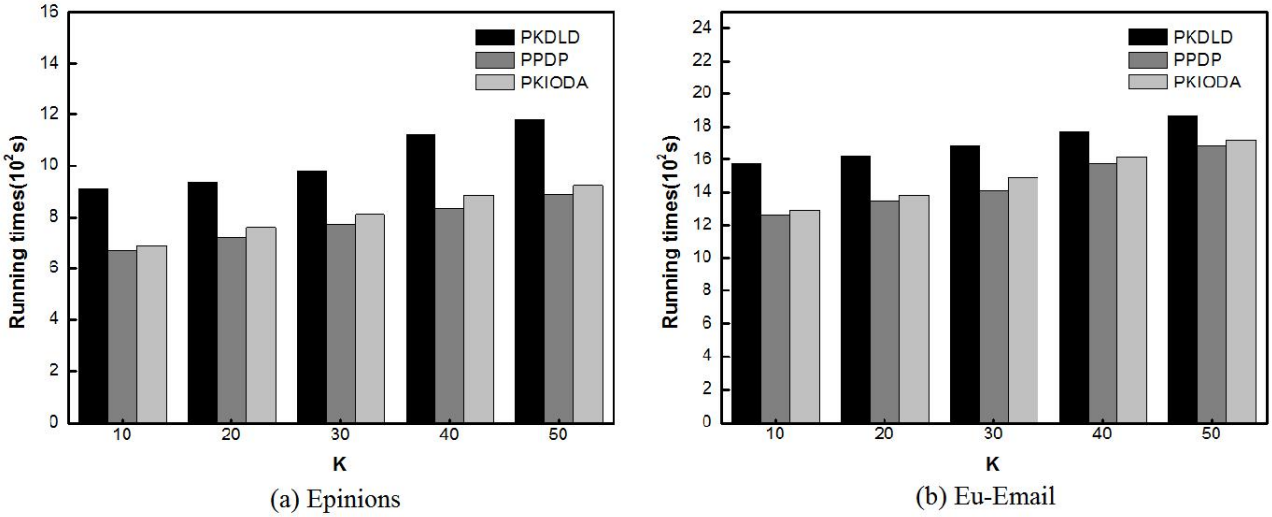


Figure 10: Running times

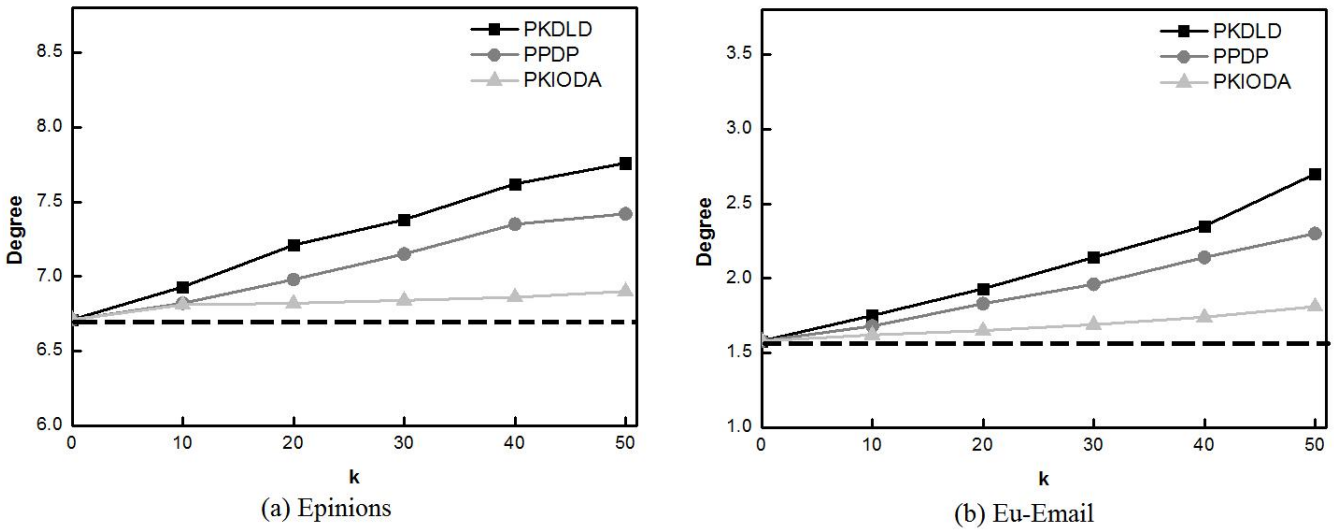


Figure 11: Change of the average in&out-degree

average in&out-degree and the average clustering coefficient (ACC). Figure 11 shows the results of the average in&out-degree of different algorithms on different datasets as the value of k increases. The dotted line in Figure 11 is the average in&out-degree the original graph. It can be seen from Figure 11 that as the value of k increases, the average degrees of the nodes of the PKDLD and PPDP algorithms change greatly after anonymity, while the PKIODA algorithm is closer to the original average degrees of the nodes. This is because the PKDLD algorithm not only considers the degree of the node but also the label attributes of the node. The PKIODA algorithm minimizes the modification of the graph, so the average in&out-degree changes less after anonymity, and the graph information loss is smaller.

Figure 12 shows the change of ACC in the anonymous process.

$$ChangeRatio = |ACC^* - ACC|/ACC. \quad (5)$$

Where ACC* indicates the average clustering coefficient after anonymity. It can be seen from Figure 12 that the change of ACC of the PKIODA algorithm less than that of other algorithms. With the increase of the k value, the PKIODA algorithm has a small rate of change in the structure of the graph, so the PKIODA algorithm better protects the structural properties of the graph.

5.4 Data Availability Analysis

The second small eigenvalue (μ_2) of the Laplacian matrix (L) is an important eigenvalue of the Laplacian matrix and is used to indicate how the community is separated. Where μ_i ($0=\mu_1 \leq \mu_2 \leq ... \leq \mu_m \leq m$) is a characteristic

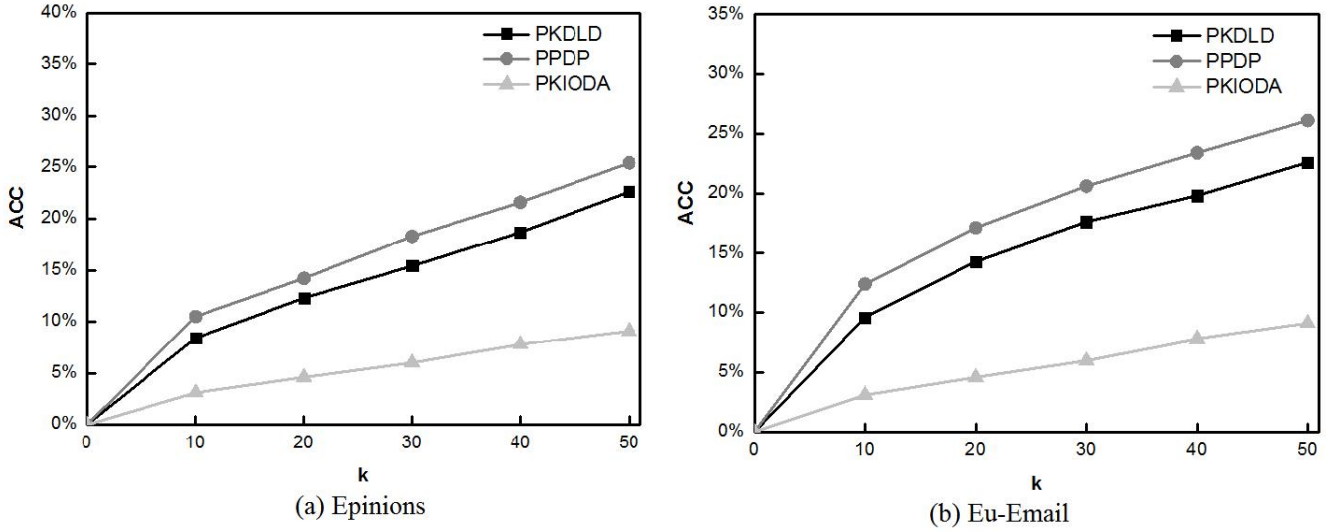


Figure 12: Change of ACC

value of L. Figure 13 shows the similarity of the μ_2 of different algorithms in different datasets as the k value increases. As can be seen from Figure 13 that the μ_2 value of the PKIODA algorithm is more similar to the original graph. This is because the PKIODA algorithm considers the community structure of the original graph when merge and delete the virtual node pairs. However, the PKDLD and PPDP algorithms ignore the protection of the community structure when anonymous, and thus cause great loss to the community structure.

Figure 14 shows the change of the community to which the node belongs in the original graph after anonymity. As can be seen from Figure 14, the PKDLD and PPDP algorithms do not consider the community of the nodes in the anonymity process. Therefore, the change of the community to which the nodes belong after anonymity is large. The PKIODA algorithm ensures that the community of the node is unchanged when the virtual node pair is deleted and deleted. Therefore, the community change rate after anonymity is less than 10%, which better maintains the data availability of the anonymous graph in the community detection.

6 Conclusion

Aiming at the large-scale social network directed graph, a large-scale social network personalized K-in&out-degree anonymity (PKIODA) algorithm based on hierarchical community structure is proposed. The algorithm divides the community based on the hierarchical community structure. According to different privacy protection requirements of users, partition the in&out-degree sequence. Finally, the virtual node pairs are merged and deleted in parallel to reduce information loss. In the process of merging and deleting virtual nodes, consider the community to which the nodes in the original graph belong. Experiments based on real social network show that the PKIODA algorithm is implemented in a distributed parallel environment, and implements a transparent privacy enforcement strategy for large-scale social network data. Compared with the traditional K-degree anonymity algorithm, the PKIODA algorithm improves the processing efficiency of large-scale social network directed graph data, and better ensures the availability of community structure analysis when data is released. PKIODA algorithm has a good effect in the average in&out-degree, ACC, μ_2, community structure protection and so on.

Acknowledgments

This work is partially supported by Natural Science Foundation of China (No.61562065) and Inner Mongolia Natural Science Foundation (No.2019MS06001).

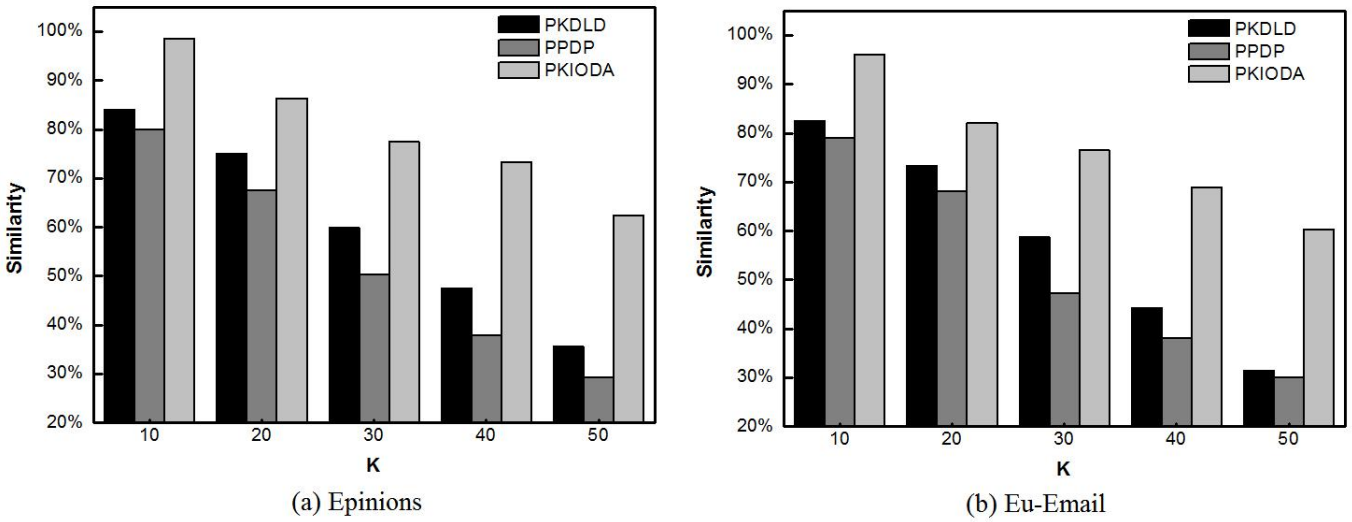
References

[1] Z. Bin and P. Jian, "The k-anonymity and l-diversity approaches for privacy preservation in social networks against neighborhood attacks," *Knowledge & Information Systems*, vol. 28, no. 1, pp. 47–77, 2011.

[2] A. Campan, Y. Alufaisan, and T. M. Truta, "Preserving communities in anonymized social networks," *Transactions on Data Privacy*, vol. 8, no. 1, pp. 55–87, 2015.

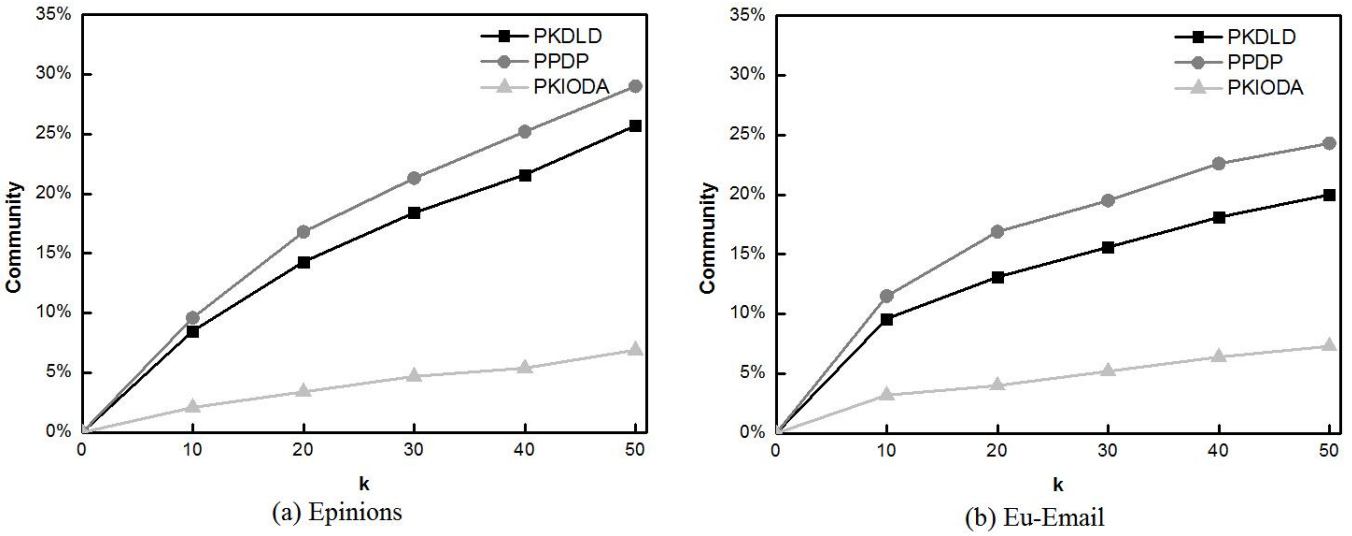
[3] A. Clauset, C. Moore, and M. E. J. Newman, "Hierarchical structure and the prediction of missing links in networks," *Nature*, vol. 453, no. 98–101, 2008.

[4] J. Jiao, P. Liu, and L. Xianxian, "A personalized privacy preserving method for publishing social network data," in *International Conference on Theory & Applications of Models of Computation*, pp. 141–157, 2014.



(a) Epinions (b) Eu-Email

Figure 13: Similarity of μ_2



(a) Epinions (b) Eu-Email

Figure 14: Change of community

[5] C. R. Jordi, H. J. Jordi, and T. Vicen ç, "k-degree anonymity and edge selection: improving data utility in large networks," *Knowledge & Information Systems*, vol. 50, no. 2, pp. 1–28, 2017.

[6] S. Juli and T. Vicen, "Graphic sequences, distances and k-degree anonymity," *Discrete Applied Mathematics*, vol. 188, no. 1, pp. 25–31, 2015.

[7] M. Kiabod, M. N. Dehkordi, and B. Barekatain, "TSRAM: A time-saving k-degree anonymization method in social network," *Expert Systems With Applications*, vol. 125, pp. 378–396, 2019.

[8] H. J. Li, Z. Bu, Y. Li, Z. Y. Zhang, Y. Chu, and G. Li, "Evolving the attribute flow for dynamical clustering in signed networks," *Chaos Solitons & Fractals*, vol. 110, pp. 20–27, 2018.

[9] X. W. Liu, Q. Q. Xie, and L. M. Wang, "Personalized extended (α, k)-anonymity model for privacy-preserving data publishing," *Concurrency and Computation Practice and Experience*, vol. 29, no. 6, 2017.

[10] Q. Lu, C. Wang, Y. Xiong, H. Xia, W. Huang, X. Gong, "Personalized privacy-preserving trajectory data publishing," *Chinese Journal of Electronics*, vol. 26, no. 2, pp. 285–291, 2017.

[11] K. R. Macwan, and S. J. Patel, "k-NMF anonymization in social network data publishing," *Computer Journal*, vol. 61, no. 4, pp. 601–613, 2018.

[12] X. Qian, R. Che, and T. K. Lee, "Differentially private network data release via structural inference," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 911–920, 2014.

[13] F. Rousseau, J. Casas-Roma, M. Vazirgiannis, "Community-preserving anonymization of graphs," *Knowledge and Information Systems*, vol. 54, pp. 315–343, 2018.

[14] K. Saurabh and K. Pradeep, "Upper approximation based privacy preserving in online social networks," *Expert Systems with Applications*, vol. 88, pp. 276–289, 2017.

[15] Y. Wang, B. Zheng, "Preserving privacy in social networks against connection fingerprint attacks," in *IEEE International Conference on Data Engineering*, 2015. DOI: 10.1109/ICDE.2015.7113272.

[16] S. Yongjiao, Y. Ye, W. Guoren, and C. Yurong, "Splitting anonymization: A novel privacy-preserving approach of social network," *Knowledge & Information Systems*, vol. 47, no. 3, pp. 595–623, 2016.

[17] L. X. Yu, W. Bin, and Y. X. Chun, "Survey on privacy preserving techniques for publishing social network data," *Journal of Software*, vol. 25, no. 3, pp. 576–590, 2014.

[18] M. Yuan, L. Chen, P. S. Yu, T. Yu, "Protecting sensitive labels in social network data anonymization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 3, pp. 633–647, 2013.

[19] J. Zhang, "Application of artificial intelligence technology in computer network security," *International Journal of Network Security*, vol. 20, no. 6, pp. 1016–1021, 2018.

[20] X. Zheng, Z. Cai, and G. Luo, "Privacy-preserved community discovery in online social networks," *Future Generation Computer Systems*, vol. 93, pp. 1002–1009, 2019.

[21] X. L. Zhang, W. C. Zhang, C. Zhang, L. X. Liu, and X. Y. He, "D-GSPerturb: A distributed social privacy protection algorithm based on graph structure perturbation," *Journal of Computers*, vol. 28, no. 5, pp. 51–61, 2017.

Biography

Xiaolin Zhang was born in Baotou, Inner Mongolia, December 1966. She received her bachelor's degree in computer science and technology from Northeastern University in 1988, her autochemistry degree from Beijing University of Science and Technology in 1995, and her Ph.D. in computer science and technology from North-eastern University in 2006. Since 1988, she has worked in Inner Mongolia University of Science and Technology. She is currently the deputy director of the Professor Committee of the Information Technology College of Inner Mongolia University of Science and Technology, the head of the computer science department of Inner Mongolia University of Science and Technology, the director of the Department of Computer Science of Inner Mongolia University of Science and Technology, the member of the Chinese Computer Society, the member of the Information System Professional Committee of the China Computer Society, and the director of the Inner Mongolia Autonomous Region Computer Society. More than 60 master's graduates have been trained and 15 masters are studying. Her current research areas: big data processing technology, social network privacy protection technology, XML database, XML data stream, wireless sensor network, uncertain database, flame image database. Prof. Zhang is responsible for many projects such as the National Natural Science Foundation of China, the National Social Science Fund Project, the Chunhui Project of the Ministry of Education, the Natural Science Foundation of Inner Mongolia, and the Inner Mongolia Education Department Fund. She has published more than 60 academic papers, including more than 20 articles in EI and 2 articles in SCI.

Jiao Liu was born in Tangshan, Hebei Province, 1995. She received the BS degree in medical information engineering from Taishan Medical College, Shandong, China, in 2017. Currently, she is pursuing a master's degree in computer science and technology at Inner Mongolia University of Science and Technology. Her research interests include big data processing technology, social network privacy protection technology.

Hongjing Bi was born in Tangshan, Hebei Province, 1983. She is a lecturer at department of Computer Science, Tangshan Normal University. She received the MS degree in computer science and technology at Inner Mongolia University of Science and Technology, Inner Mongolia, China, in 2010. Her research interests include information security, big data application, social network privacy protection technology.

Jian Li was born in Hulunbeier, Inner Mongolia, 1995. He received the BS degree in software engineer-ing from Inner Mongolia University of Science and Technology, Inner Mongolia, China, in 2017. At present, he is pursuing a master's degree in computer science and technology at Inner Mongolia University of Science and Technology, China. His research areas include big data processing technology, social network privacy protection technology.

Yongping Wang was born in Baotou, Inner Mongolia, 1982. She is a lecturer at Inner Mongolia University of Science and Technology. Her research interests include large-scale social network privacy protection, large-scale social network community structure privacy protection. She has presided over one school-level project and published three academic papers.

Research on Crawling Network Information Data with Scrapy Framework

Dashan Wang, Qingbin Zhang, and Shaoxian Hong
(Corresponding author: Shaoxian Hong)

Hainan College of Vocation and Technique, China
No. 95, Nanhai Avenue, Haikou, Hainan 570216, China
(Email: xianyi060760@163.com)

(Received Jan. 31, 2019; Revised and Accepted Feb. 18, 2020; First Online Feb. 16, 2021)

Abstract

In the Internet era of big data, the emergence of crawlers significantly improves information retrieval efficiency. This paper briefly introduced the basic structure of crawler software, the scrapy framework, and the clustering algorithm used to improve the performance of information crawling and classification. Then, the crawler software and clustering algorithm were programmed by the python software. Experiments were carried out using the MATLAB software in the LAN in a laboratory to test the Weibo data between October 1 and October 31. Moreover, a crawler software that adopted the scrapy framework but did not add the clustering algorithm was taken as a control. The results showed that the scrapy framework based crawler software could not achieve the same Weibo information classification as the actual classification whether the clustering algorithm was added or not; the crawler software that was added with the clustering algorithm was closer to the exact proportion in classification and obtained classification results with higher accuracy and lower false alarm rate in a shorter time.

Keywords: Clustering Algorithm; Crawler Software; Network Data; Scrapy Framework

1 Introduction

With the development of computer technology and the birth of the Internet, the speed of information generation has gained an explosive improvement [9]. Especially in recent years, with the popularization of 4G communication technology, the mobile Internet has been fully developed. After combining the mobile Internet and the traditional Internet, the generation and transmission speed of information data further increases.

The advent of the big data era makes people's life more convenient, which is mainly reflected in the fact that users can use more data to assist their different choices and service providers can optimize their services according to big data. However, the emergence of big data not only brings convenience but also brings difficulties. The excellence of big data is reflected in a large number of laws hidden in a large number of data, which can assist the decision-making of individuals or enterprises better. However, due to a large amount of data, data fragments that support different laws are scattered, and the method of human retrieval alone cannot meet the retrieval needs [2]. Crawler technology can replace manual search to retrieve big data and also can carry out preliminary classification of the retrieved data, which is convenient for mining the rules.

In order to crawl deep web pages, Feng *et al.* [13] designed an intelligent crawler with a two-stage framework. In the first stage, with the help of a search engine, the central page search based on the site is performed to avoid visiting a large number of pages. In the second stage, the most relevant links are mined to realize fast site search. The simulation results showed that the crawler could effectively retrieve the deep web interface in large-scale websites and obtained a higher harvest rate than other crawlers. Seyfi [10] proposed a focus crawler that uses specific HTML elements of a page to predict the topic focus of all pages in the current page that have unvisited links and verified the effectiveness of the method through simulations.

Huang *et al.* [5] put forward an extensible GeoWeb crawler framework that could search various GeoWeb resources and verified through simulations that the framework had good extendibility. This paper briefly introduced the basic structure of crawler software, the scrapy framework, and the clustering algorithm that was used for improving the performance of information crawling and classification. Then, the crawler software and clustering algorithm were programmed by the python software. Experiments were carried out using the MATLAB software in the LAN in a laboratory to test the Weibo data between October 1 and October 31. Moreover, a crawler software that adopted the scrapy framework but did not add the clustering algorithm was taken as a control.

2 Crawler Software Based on the Scrapy Framework

2.1 The Basic Structure of Crawler Software

The basic framework of the crawler software for network data crawling [6] is shown in Figure 1. In the overall structure, crawler software is divided into an interaction layer, logical business layer, and database layer. The interaction layer is the top layer of the software, responsible for the human-computer interaction with users.

The main content of the interaction layer is the design of the application form, which includes the main page module, task view module, server view module, and client view module. The main page module is responsible for querying the task information list and carry out various operations on the task. The task view module is the module for editing the task information, which can directly edit the task by visualizing the task data. The module is generally nested in the main page module. The server module is responsible for monitoring the user's use of the client. The client module is used by the user to view the software's connection to the server and to receive or release tasks.

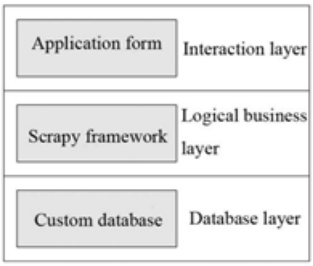


Figure 1: The basic framework of crawler software

Next is the logical business layer, which is mainly composed of a scrapy framework [1]. It is the functional core of the whole crawler software. Its main function in the software is to realize the task description submitted by the interaction layer, generate the corresponding crawler, download the network data in the given URL address, and summarize and count the string.

The last one is the database layer, whose main structure is a custom database. Its main function is to store or delete the network data searched in the URL address in the logical business layer. The user-defined database will be created according to the user's needs. When creating the database, the user only needs to input the necessary information such as database type, name, and account password into the configuration file of the database. The user-defined database will also automatically layer different crawling tasks for the easy query.

2.2 Scrapy Framework

After the Internet has entered the era of big data, the amount of information data has expanded rapidly, which greatly increases the difficulty of information retrieval. The huge amount of data not only increases the difficulty of information retrieval but also brings more high-value hidden rules. In the face of the increasing amount of network information, the emergence of search engines makes information retrieval more convenient. The working principle of a search engine is to crawl information data in the Internet using crawler software and classify the information according to the needed keywords.

As Python is easy to learn and has a large standard library of modules, crawlers are usually written in Python. Scrapy is a crawler framework completely written by Python language [4], and its operation diagram is shown in Figure 2. The crawler framework consists of a scrapy engine, scheduler, crawler, crawler middleware, downloader, download middleware, project pipeline, and the Internet. The scrapy engine is the core of the whole operating framework, which is used for processing the data flow in the operation process. The scheduler is connected with the engine to store the crawling requests from the engine and provide the stored crawling requests to the engine, *i.e.*, a crawling task list.

The downloader is connected with the Internet, downloads the web page information on the Internet according to the task target order given by the scheduler, and transmits it to the crawler. The crawler module that contains the Internet crawling logic and parsing rules for downloaded content is responsible for generating the parsing results and subsequent search requests. The project pipeline will receive the result data from the crawler, clean and verify the data to reduce the interference of bad information, and store them in the database. The crawler middleware and download middleware are the intermediate processing modules between the crawler and engine and between the downloader and engine, respectively [8].

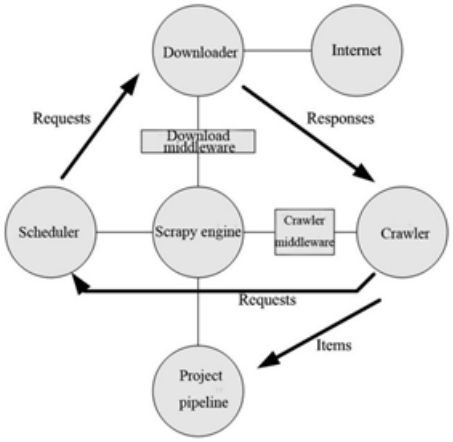


Figure 2: Scrapy operation framework

The basic operation flow of scrapy is as follows:

1) The crawler generates the request according to the crawling logic and then transmits it to the scheduler to get the crawler request list without being processed by the engine.

2) The engine obtains the crawler request from the scheduler and starts to crawl the information, and the request is passed to the downloader through the download middleware.

3) The downloader downloads the web page information from the Internet according to the address given by the crawler request and transmits it to the engine through the download middleware.

4) The engine feeds back the web page information to the crawler through the crawler middleware, and the crawler parses the information according to the set parsing rules.

5) The data obtained after parsing is transferred to the project pipeline by the engine to clean and verify the data.

The above steps cycle until there are no more crawler requests in the scheduler.

2.3 Clustering Algorithm for Data Arrangement

Crawler software usually crawls the network information to obtain the needed information and improve the retrieval efficiency. Facing the huge amount of network information, in order to facilitate storage and subsequent retrieval, crawler software will classify the crawled information.

The traditional crawler software usually classifies the string of the crawled information by word segmentation and divides the information containing the same keywords into one category. This method is relatively simple, and information containing the same keywords is generally relevant on the surface. However, in the era of big data, it is more important to deeply mine the hidden rules in the network information. Classifying by relying on keywords only is likely to divide the information with the same or similar content but different keywords into different topics, which will ultimately affect the effectiveness and comprehensiveness of the retrieval results. Clustering algorithm [14] is an algorithm that divides based on the difference between data, which not only depends on the difference of keywords but also depends on the deep connection of information.

In the crawler software, the crawler of the scrapy framework crawls the information data according to the URL. Before storing the data in the database, the data are classified using the clustering algorithm to facilitate the subsequent accurate storage and retrieval. The data classification flow of the clustering algorithm is shown in Figure 3.

1) Firstly, the crawling data are preprocessed to remove the information noise and segment words. The removal of information noise includes deleting the meaningless characters and the text that cannot express the meaning because of few words. The segmentation of words is to obtain individual words from the text to form the vector features of the information data.

2) Then, Gibbs sampling is performed on the preprocessed information data [7] to reduce the vector feature dimension of the information data and the amount of calculation. The sampling formula is:

$$\begin{aligned} & P(Z_j = k|\overrightarrow{Z_{\neg i}}, \overrightarrow{w}, \alpha\beta) \propto \theta_{mk} \cdot \varphi_{kt} \\ = & \frac{n^k_{m,\neg i} + \alpha_k}{\sum_{k=1}^{K}(n^t_{m,\neg i} + \alpha_K)} \cdot \frac{n^t_{m,\neg i} + \beta_k}{\sum_{t=1}^{K}(n^t_{k,\neg i} + \beta_K)} \end{aligned}$$

where Z_j is the j^{th} word in all information data sets, i is a two-dimensional subscript, which is composed of m and n, representing n words in m information data, K represents for the number of hidden themes, $\overrightarrow{w}$ stands for a word, and α and β are the prior superparameter of information data theme and the prior superparameter of words under the theme respectively, both of which obey the Dirichlet distribution [11]. After repeated sampling to convergence, the theme distribution of every information data (θ) can be obtained, which is taken as the vector feature of information data.

3) According to the input K value, K cluster centers are randomly generated, and then the distance between information data and different cluster centers is calculated according to the distance calculation formula:

$$d(x, Z_j) = \sqrt{\sum_{i=1}^{n}(x_i - Z_{ji})^2},$$

where $d(x, Z_j)$ stands for the distance between data x and cluster center Z_j, x_i stands for the i^{th} dimensional data of x, and Z_{ji} stands for the i^{th} dimensional data of Z_j. Then, according to the distance, the information data are allocated to different cluster centers.

4) After clustering, the cluster center of every kind of data set is recalculated, and then Step 3 is repeated to reclassify.

5) Steps 3 and 4 are repeated until the clustering criterion function reaches the predetermined standard, and its equation expression [12] is:

$$J(I) = \sum_{j=1}^{k}\sum_{i=1}^{n_j} ||x_i^j - Z_j(I)||^2$$
$$|J(I) - J(I-1)| < \xi,$$

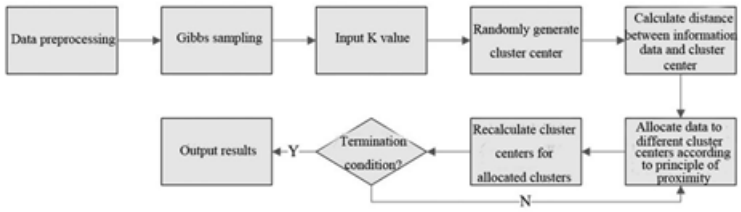


Figure 3: The calculation flow of the clustering algorithm

where $J(I)$ stands for the square error of the I^{th} clustering results, x_i^j stands for the i^{th} data in j category, $Z_j(I)$ stands for the cluster center of the j category at the I^{th} clustering, and ξ is a threshold for determining whether the iteration terminates or not.

3 Simulation Analysis

3.1 Experimental Environment

In this study, the crawler software and its clustering algorithm were programmed using the python software. The simulation experiment was carried out in a laboratory server using MATLAB [3]. The configuration of the server was Windows 7 operating system, 16G memory, and Core i7 processor.

3.2 Experimental Data

In order to verify the effectiveness of the crawler software in crawling information data after adding the clustering algorithm, this study compared it with the crawler software without the clustering algorithm. In order to ensure the accuracy of the comparison results, it is necessary to know the actual information data of the subject crawled by the crawler software. However, in the real Internet, new information will be generated constantly, and it is nearly impossible to collect complete actual information data. Therefore, the experiment in this study built a LAN in the laboratory.

In the LAN, a server provided website services, and the rest of PC used the crawler software to crawl the website information. The web page data in the server providing website service came from the collectible information of Weibo. The Weibo data between October 1 and October 31 were collected through the application programming interface (API) of Weibo. There were a total of 3000 text messages, including five themes: 5G (520 messages), mobile payment (390 messages), anti-corruption work (650 messages), animation (750 messages), and environmental protection (690 messages). Through the manual review, the theme of Weibo data came from the central idea reflected by each message. The messages might not include the same keywords as the theme name. There was also a connection between different themes, and there was a small amount of Weibo information containing keywords of other theme names.

The above situation of keyword mixing in different themes could be used as the interference of crawler software on crawling information classification storage. In the experiment, crawler software with clustering algorithm and software without clustering algorithm were used to crawl the micro blog information in the laboratory LAN, and then the information after crawling classification was analyzed.

3.3 Experimental Results

Two kinds of crawler software crawled the Weibo information in the LAN of the laboratory and then classified and stored the crawled information. The final results are shown in Figure 4, showing the actual proportion of Weibo information classification. It was seen from Figure 4 that "5G" messages accounted for 17.33%, "mobile payment" messages accounted for 13.00%, "anti-corruption work" messages accounted for 21.67%, "animation" messages accounted for 25.00%, "environmental protection" messages accounted for 23.00%, and the rest of messages accounted for 0% among the actual Weibo information; in the classification of the crawler software without the clustering algorithm, "5G" messages accounted for 16.00%, "mobile payment" messages accounted for 12.00%, "anti-corruption work" messages accounted for 21.07%, "animation" messages accounted for 24.37%, "environmental protection" messages accounted for 22.5%, and the rest of messages accounted for 4.07%.

In the classification of the crawler software that was added with the clustering algorithm, "5G" messages accounted for 17.30%, "mobile payment" messages accounted for 12.93%, "anti-corruption work" messages accounted for 21.57%, "animation" messages accounted for 24.93%, "environmental protection" messages accounted for 22.97%, and the rest of messages accounted for 0.30%. It was seen from Figure 4 that both crawler software could effectively crawl effective information from Weibo and classify information. There were only five classification themes in the actual Weibo information. Although the two kinds of crawler software also classified five themes, there were some other information classification, especially the classification by the crawler software without the clustering algorithm. Only a small part of the infor-

Table 1: Accuracy and false alarm rate of two kinds of crawler software for classification of Weibo crawling information

	Crawler Software **without** the Clustering Algorithm		Crawler Software **with** the Clustering Algorithm	
	Accuracy/%	False alarm rate/%	Accuracy/%	False alarm rate/%
5G	88.7	5.2	98.2	1.7
Mobile payment	89.6	5.6	98.3	1.6
Anti-corruption work	89.4	5.4	97.5	2.3
Animation	88.8	5.2	98.6	1.4
Environmental protection	89.2	5.7	99.1	0.7
Comprehensive evaluation	89.1	5.4	98.3	1.5

mation was classified as other categories by the crawler software added with the clustering algorithm.

On the whole, the classification of the crawling information by the crawler software that was added with the clustering algorithm was very close to the actual Weibo information classification; however, the crawler software without the clustering algorithm classified more information into other categories, and the classification of crawling information was more deviated from the actual information classification.

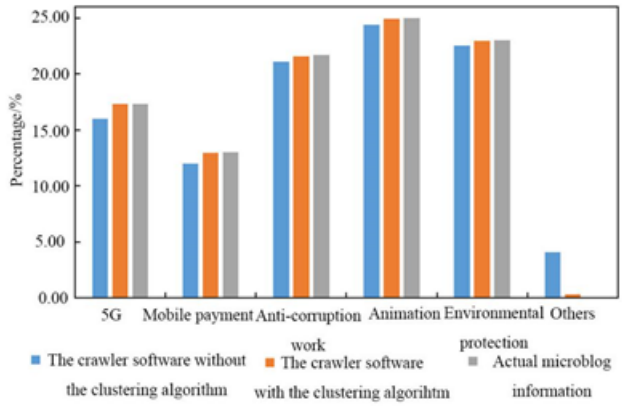


Figure 4: Classification proportion of Weibo crawling information by the two crawlers and the actual proportion

Although the classification proportion of Weibo crawling information shown above also reflected the effect of the two crawler software on information crawling and classification, the classification proportion only evaluated the classification information from the whole but could not reflect whether the different information was classified accurately. Table 1 shows the classification accuracy and false alarm rate of the two crawlers. By comparison, it was found that no matter what kind of classification information, the crawler software with the clustering algorithm had higher accuracy and lower false alarm rate.

When the two kinds of crawler software classified the Weibo information, there were other types that were not identified in the actual Weibo information classification; moreover, the crawler software with the clustering algorithm had higher classification accuracy and lower false alarm rate. The reason was that keyword mixing between different themes interfered with the classification of the two software, especially the crawler software that was not added with the clustering algorithm. The crawler software with the clustering algorithm classified the text information based on the vector features of the information; therefore, the influence caused by fixed keyword mixing was relatively small. As the keyword was also a part of the vector feature, the keyword mixing still impacted the features, leading to classification errors.

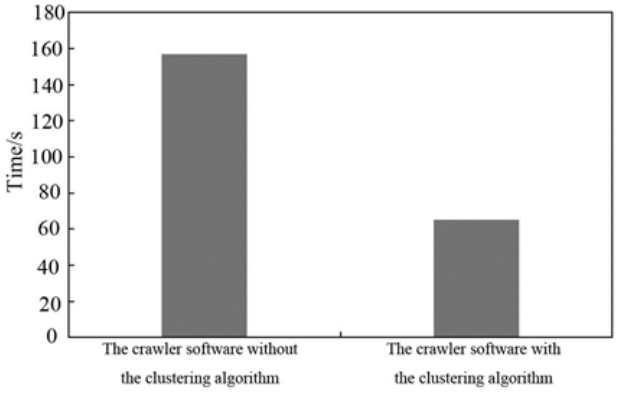


Figure 5: The time required for two kinds of crawler software to crawl and classify Weibo information

As shown in Figure 5, it took 157 s for the crawler software without clustering algorithm to crawl and classify Weibo information and 65 s for the crawler software with the clustering algorithm. The comparison in Figure 5 shows that the scrapy framework based crawler software that was added with the clustering algorithm could classify the crawling information faster in the face of big data Weibo information could classify the crawled information faster when faced with a large amount of Weibo information. Combined with the above results, it was seen that the crawler software that adopted the scrapy framework could effectively crawl the Weibo information and could classify the crawled information faster and more

accurately after using the clustering algorithm.

4 Conclusion

This paper briefly introduced the basic structure of crawler software, the scrapy framework, and the clustering algorithm that was used for improving the performance of information classification. Then, the crawler software and clustering algorithm were programmed by the python software, and an experiment was carried out on Weibo data between October 1 and October 31 using the MATLAB software in LAN. The crawler software that adopted the scrapy framework but did not add the clustering algorithm was used as the control. The final experimental results are as follows:

1) In the statistics of the classification proportion of Weibo information, the two crawlers could effectively crawl the effective information from the Weibo and classify the information, but neither of them could make the same proportion as the actual classification; the category except the five themes appeared in the classification of both software, but the classification proportion obtained by the crawler that was added with the clustering algorithm was closer to the actual proportion;

2) The scrapy framework based crawler software had higher accuracy and lower false alarm rate in crawling and classifying Weibo information after being added with the clustering algorithm;

3) The scrapy framework based crawler software spent less time crawling and classifying information after being added with the clustering algorithm.

References

[1] J. Bao, P. Liu, H. Yu, C. Xu, "Incorporating twitter-based human activity information in spatial analysis of crashes in urban areas," *Accident Analysis & Prevention*, vol. 106, pp. 358-369, 2017.

[2] T. Fang, T. Han, C. Zhang, Y. J. Yao, "Research and construction of the online pesticide information center and discovery platform based on web crawler," *Procedia Computer Science*, vol. 166, pp. 9-14, 2020.

[3] A. C. Gabardo, R. Berretta, P. Moscato, "M-Link: a link clustering memetic algorithm for overlapping community detection," *Memetic Computing*, vol. 12, no. 2, pp. 87-99, 2020.

[4] U. R. Hodeghatta, S. Sahney, "Understanding twitter as an e-WOM," *Journal of Systems & Information Technology*, vol. 18, no. 1, pp. 89-115, 2016.

[5] C. Y. Huang, H. Chang, "GeoWeb crawler: an extensible and scalable web crawling framework for discovering geospatial web resources," *International Journal of Geo-Information*, vol. 5, no. 8, pp. 136, 2016.

[6] M. A. Kausar, V. S. Dhaka, S. K. Singh, "Design of web crawler for the client - server technology," *Indian Journal of Science and Technology*, vol. 8, no. 36, 2015.

[7] F. Li, L. L. Dai, Z. Y. Jiang, S. Z. Li, "Single-Pass Clustering Algorithm Based on Storm," *Journal of Physics Conference*, vol. 806, pp. 012017, 2017.

[8] S. H. Peng, P. Y. Liu, J. Han, "A python security analysis framework in integrity verification and vulnerability detection," *Wuhan University Journal of Natural Sciences*, vol. 24, no. 2, pp. 141-148, 2019.

[9] S. Raj, R. Krishna, A. Nayak, "Distributed component-based crawler for AJAX applications," in *Second International Conference on Advances in Electronics, Computers and Communications*, pp. 1-6, 2018.

[10] A. Seyfi, "Analysis and evaluation of the link and content based focused treasure-crawler," *Computer Standards & Interfaces*, vol. 44, pp. 54-62, 2016.

[11] F. F. Wang, B. H. Zhang, S. C. Chai, "Deep auto-encoded clustering algorithm for community detection in complex networks," *Chinese Journal of Electronics*, vol. 28, no. 3, pp. 49-56, 2019.

[12] B. Yuan, T. Jiang, H. Z. Yu, "Emotional classification algorithm of micro-blog text based on the combination of emotional characteristics," *Advanced Materials Research*, vol. 1077, pp. 246-251, 2015.

[13] F. Zhao, J. Zhou, C. Nie, H. Huang, H. Jin, "SmartCrawler: A two-stage crawler for efficiently harvesting deep-web interfaces," *IEEE Transactions on Services Computing*, vol. 9, no. 4, pp. 608-620, 2016.

[14] F. Zhao, Y. Zhu, H. Jin, L. T. Yang, "A personalized hashtag recommendation approach using LDA-based topic model in microblog environment," *Future Generation Computer Systems*, vol. 65, pp. 196-206, 2016.

Biography

Dashan Wang, born in 1980, received the master's degree of engineer from Hubei Agricultural College in 2003. He is a lecturer in Hainan College of Vocation and Technique. He is interested in computer network.

Qingbin Zhang, born in 1989, received the bachelor's degree from Dongbei University of Finance and Economics in 2020. He is a network engineer in Hainan College of Vocation and Technique. He is interested in computer network.

Shaoxian Hong, born in 1977, received the master's degree of education from Henan University in 2001. She is an associate professor in Henan University. She is interested in English teaching theory and research.

Decentralized Multi-Authority Attribute-based Searchable Encryption Scheme

Juan Ren[1,2], Leyou Zhang[2], and Baocang Wang[3,4]
(Corresponding author: Juan Ren)

Science and Technology on Communication Security Laboratory, Chengdu 610041, P.R. China[1]
School of Mathematics and Statistics, Xidian University, Xi'an 710126, China[2]
School of Information Engineering, Xuchang University, Xuchang 461000, China[3]
State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China[4]
(Email: juaner_r@126.com)
(Received Oct. 14, 2019; Revised and Accepted May 8, 2020; First Online Feb. 18, 2021)

Abstract

Attribute-based searchable encryption (ABSE) scheme is an efficient mechanism to implement access control and secure keywords search based on attributes over encrypted data. However, most existing ABSE schemes rely on single trusted authority to manage the attribute private keys. In real life, it is impractical that one authority completes all verifications and certifications to all attributes. In addition, the existence of the vulnerable item makes them be vulnerable to secret-key-recovery attack in some existing multi-authority attribute-based encryption (ABE) schemes based on access tree. To solve above problems, we design a decentralized multi-authority ABSE scheme based on access tree, which can resist the keyword guessing (KG) attack and the secret-keys-recovery attack. We also give performance analysis of the proposed scheme and prove it to be selectively secure under the decisional bilinear Diffie-Hellman (DBDH) assumption, the hash Diffie-Hellman (HDH) assumption and the bilinear Diffie-Hellman (BDH) assumption.

Keywords: Access Tree; Decentralized; Searchable Encryption; Stronger Resistance-Attack

1 Introduction

In the era of big data, in order to store a large amount of data generated by users conveniently, cloud storage technology emerges at the historic moment. The users outsource their data to cloud server in cloud storage technology. However, it can be find that users lose control to the outsourced data which brings huge challenges to the privacy security of users. The most common solution to solve the above problem is to encrypt the data and upload it. Later, a new puzzle has emerged about how to search ciphertext according to the keywords when users need to find relevant files containing certain keywords. The best way to solve this problem is to search keywords in the plaintext after downloading and decrypting the files, which also results in a lot of network overhead and computational overhead due to it requires firstly downloading and decrypting useless files before searching them. Basing on its strong computing power, the cloud server is expected to perform retrieval function instead of the users, in which the cloud server can decrypt and search after obtaining the keys and keywords. In fact, the cloud server is not trusted, so there is a risk of leakage when the privacy of users is exposed to the cloud server. In order to solve these problems, attribute-based searchable encryption (ABSE) technology comes into being. ABSE scheme can achieve fine-grained access control, and perform the search operation without compromising users' privacy.

In searchable encryption schemes, the data owner encrypts the data and uploads the ciphertext to the cloud server. To search, the user generates a trapdoor (namely search capability) and sends it to the cloud server. With the trapdoor, the cloud server can search the index and retrieve the corresponding data. If the match is successful, it means that the keywords is included in the ciphertext. Finally, the cloud server return the ciphertext to user. After receiving the search results, the user only need to decrypt the returned ciphertext. A searchable encryption scheme allows the cloud server to search in encrypted data on behalf of a user without learning information about keywords or plaintext. Recently, some efficient ABSE schemes supporting some useful properties were proposed. But there are some problems as follows:

1) Single authority: At present, many ABE schemes with keywords search [6, 10, 14, 22, 24] were proposed, in which the data owner could control the search result. Both keywords privacy and data privacy are protected during the searching process in these schemes. However, we must trust the single-authority to issue the attribute private keys for authorized users in above schemes, which easily causes

congestion and reduces the system efficiency. Besides, there is more than one party to act as an authority in practice. Considering this example: the attributes of a student could be ID number, student number and driving license number, which are managed by the Public Security Bureau, the School Office and the department of Motor Vehicles respectively. So it is still a meaningful challenge to come up with a multi-authority ABSE scheme.

2) Vulnerable to secret-keys-recovery attack: The second problem is caused due to the existence of the item g^r in multi-authority ABE schemes based on access tree. ABSE schemes could be generated based on ABE scheme in general. However, there are some problems in existing multi-authority ABE schemes based access tree. In a typical ABE scheme based on access tree [2], a secret value r is selected to build the private key component $g^r H(att_i)^{a_x}$ by the trusted single-authority. Later, these scheme [7, 11, 12] are vulnerable to secret-keys-recovery attack once either AA or CA obtains the secret value r or g^r, unless AA and CA are trusted authorities.

3) Vulnerable to keyword guessing (KG) attack: The last issue is about resistance to KG attack. The existing some ABSE schemes such as [10,13,14,23] cannot resist the KG attack since the test operation can be performed by any part in system.

1.1 Related Work

1.1.1 Searchable Encryption

Song *et al.* [19] proposed the first keyword search on ciphertext with symmetric encryption method. It can only support single keyword search, and search requires linearly scan each file document word by word. However, symmetric searchable encryption schemes only support user-server-user model and unsuitable for three-party situation, which is unsuitable in the cloud environment. Boneh *et al.* [3] solved this problem and proposed the first public key encryption keyword search (PEKS). Their scheme provides a solution for the third-party user to search on the encrypted data. However, Boneh's scheme requires a secure channel and cannot achieve indistinguishability of trapdoor. Abdalla *et al.* [1] proposed a new definition consistency of keyword search in ciphertext and designed a new PEKS scheme from identity-based encryption (IBE).

Combined the ABE schemes and searchable encryption schemes, the ABSE scheme can realize fine-grained access control and keywords search. Wang *et al.* [20] designed an ABE scheme with a scalable authorized keyword search, which supported conjunctive keyword query without increasing more computation overhead. However, the decryption key must be shared through a secure communication channel from the data owner to the data user, and no theoretical security proofs are given in this scheme. Later, ABE scheme with keyword search [6] was proposed, which only supported single keyword search and the receiver's identity was disclosed in these schemes. Recently, many ABSE schemes [21, 24] were proposed respectively. Miao *et al.* [14] designed a searchable keyword-based scheme to deal with multi-keyword query. However, it cannot resist the KG attack due to the test phase can be performed by any part of the system. The scheme [9] focuses on achieving the multi-keyword search and improving the efficiency, but it can not be against the KG attack. Wang *et al.* [22] proposed an ABSE scheme with revocation for mobile cloud storage, which could resist the KG attack. However, it only supports the single keyword search. An ABSE scheme with the constant size ciphertext was proposed in [10], which addressed the suffer from linear storage and computation costs. But the scheme cannot provide multi-keyword query. Yang *et al.* [23] designed an ABSE scheme for cloud platform with supporting access control and keyword search. However, it is necessary to trust the single-authority for issuing the attribute private keys of authorized users in above schemes.

For searching operation over ABE, it is not realistic to trust a single authority to monitor all attributes keys in practical situation. Kuchta *et al.* [13] presented a multi-authority ABSE scheme, in which multiple attribute authorities managed the secret keys. Zhu *et al.* [29] also proposed a multi-authority ABSE scheme without CA, but it lacked of some practical properties, such as the resistance of KG attack.

1.1.2 Attribute-based Encryption

Sahai and Waters [18] proposed a transformation from IBE to ABE. The earliest ABE scheme can only support threshold access control. ABE schemes are divided into two types including Key-Policy ABE (KP-ABE) and Ciphertext-Policy ABE (CP-ABE). The attributes are used to annotate the ciphertexts and formulas over these attributes are ascribed to users' secret keys in the KP-ABE scheme. CP-ABE is complementary in that attributes are associated with the user's credentials and the formulas over these credentials are attached to the ciphertext by the encrypting party. Subsequently, Bethencourt *et al.* [2] constructed the first CP-ABE scheme where access structures are described by a monotonic "access tree". However, this scheme proved its security under the generic bilinear group model. Recently, many ABE schemes [8, 25, 27, 28] were proposed, but they only supported single authority. Chase and Chow [4] made the extension to Sahai and Waters scheme from another view, and proposed a multi-authority ABE scheme achieving the practical requirements. Chow [5] proposed a new privacy-preserving architecture for multi-authority ABE without the CA. Rouselakis *et al.* [17] proposed an efficient large-universe multi-authority CP-ABE system. Their construction achieves maximum versatility by allowing multiple authorities to control the key distribution for an exponential number of attributes. Following their

works, many researchers focus on multi-authority ABE schemes [15, 26] that satisfy the practical requirements.

1.2 Our Contributions

In view of these disadvantages of the above schemes, this paper focuses on designing a decentralized multi-authority ABSE scheme. The main contributions can be described as follows:

- *Decentralized Multi-authority ABSE:* The main idea of our scheme is to find a way to extend the single-authority ABKS scheme to decentralized multi-authority ABKS scheme. The proposed scheme realizes flexible access control and keyword search without a trusted CA, meanwhile, guaranteeing AAs' extensibility.

- *Improving the schemes [7, 11, 12] to hide the vulnerable items g^r and r:* To address the weakness of the scheme [7, 11, 12] in terms of parameters item g^r and r, our scheme is improved by introducing the secret key distribution protocol and is extended into a decentralizing multi-authority ABSE scheme. After performing the interaction with the data owner based on the key distribution protocol, AAs could get the secret key components $g^r H(att_i)^{r_{i,j}}$ without knowing g^r. In the process, g^r is hidden by blinding itself to resist secret-keys-recovery attack.

- *Resistance to KG attack:* In our scheme, the search trapdoor is structured by using blinded secret key components $blinded\_D_{i,j}$ instead of original secret key components $D_{i,j}$ used to decrypt, which increases the resistance to KG attack. In addition, the cloud server completes the search operation using its own secret value a to resist KG attack.

1.3 Organization

The rest paper is organized as follows. Section 2 describes some preliminaries which includes some basic definitions and assumption. The system model and security model will be presented in Section 3. The main construction and the security proof will be presented in Section 4. Section 5 gives a detailed performance analysis. At last, we end this work with a brief conclusion in Section 6.

2 Preliminaries

In this section, we introduce several necessary definitions and techniques used to design our scheme.

2.1 Bilinear Map and Complexity Assumption

Definition 1. (Bilinear map) *Let G and G_T be two multiplicative cycle groups of same prime order p, g is the generator of G. $e : G \times G \to G_T$ is a bilinear map with the following properties:*

- *Bilinear: $\forall a, b \in Z_p$ and $e(g^a, g^b) = e(g, g)^{ab}$.*
- *Non-Degenerate: $e(g, g) \neq 1$.*
- *Computable: $e(g, g)$ is polynomial-time computable.*

Definition 2. (BDH assumption) *Let $a, b, c, z \in_R Z_p$. Given the tuple $(A, B, C) = (g^a, g^b, g^c)$, the bilinear Diffie-Hellman (BDH) assumption holds when no polynomial-time algorithm $\mathcal{B}$ can compute the value g^{abc}.*

Definition 3. (DBDH assumption) *Let $a, b, c, z \in_R Z_p$. Given the tuple $(A, B, C) = (g^a, g^b, g^c)$, the decisional bilinear Diffie-Hellman (DBDH) assumption holds when no polynomial-time algorithm $\mathcal{B}$ can distinguish g^{abc} and g^z with non-negligible advantage. The advantage of algorithm $\mathcal{B}$ is*
$Adv_{\mathcal{B}}^{DBDH}$
$= |\Pr[\mathcal{B}(A, B, C, g^{abc}) = 1] - \Pr[\mathcal{B}(A, B, C, g^z) = 1]| \leq \epsilon.$

Definition 4. (HDH assumption) *Let $a, b \in_R Z_p$. Given the tuple $(A, B, C) = (g^a, g^b, H(g^z))$ and hash function $H(\cdot)$, the hash Diffie-Hellman (HDH) assumption holds when no polynomial-time algorithm $\mathcal{B}$ can distinguish $z = ab \,(mod\ \ p)$ and $z \in Z_p$ with non-negligible advantage. The advantage of algorithm $\mathcal{B}$ is*
$Adv_{\mathcal{B}}^{HDH}$
$= |\Pr[\mathcal{B}(A, B, H(g^{ab}) = 1] - \Pr[\mathcal{B}(A, B, H(g^z) = 1]| \leq \epsilon.$

2.2 Access Structure

We propose a set of $p_1, p_2, ..., p_n$ as an attribute set. For $\forall B, C$: if $B \in A \wedge B \subseteq C$, then $C \in A$, we can get the set $A \subseteq 2^{\{p_1, p_2, ..., p_n\}}$ is monotone. An access structure (respectively, monotone access structure) is a set A which is non-empty subsets of $p_1, p_2, ..., p_n$. The sets in A are named authorized sets, and the sets not belong to A are named as unauthorized sets.

Definition 5. (Access tree) *A secret s is divided into n shares in a such way that any subset of t or more shares can reconstruct the secret, but no subset of fewer than t shares can. The scheme is based on polynomial interpolation. $f(x)$ is a $t-1$ degree polynomial, which is uniquely defined by t points (x_i, y_i).*

- *Secret sharing:*
  - *Randomly choose a secret $s \in Z_p$.*
  - *Let $c_0 = s$. Choose randomly $t-1$ coefficients $c_1, c_2, ..., c_{t-1} \in Z_p$, and define $f(x) = \sum_{i=0}^{t-1} c_i x^i$.*
  - *Computer $s_i = f(i) \,mod\, P$, where s_i is the $i-th$ share of the secret s.*
- *Secret reconstruction: Let $S \subseteq \{1, ..., n\}$ denote any subset that contains t values. Using t shares s_i where*

$i \in S$, the function $f(x)$ can reconstruct from the following Lagrange interpolation

$$f(x) = \sum_{i \in S} s_i \cdot \triangle_{i,S}(x)$$

where $\triangle_{i,S}(x) = \prod_{j \in S, j \neq i} \frac{x-j}{i-j}$. So the secret can be recovered by $f(0) = c_0 = s$.

3 System and Security Model

3.1 System Model

There are four entities: The data owner, the data user, AAs, and the cloud server in the system as showed in Figure 1, the details are as follows.

Data owner: The data owner's work includes two parts:

- The data owner specifies the access tree, and encrypts the data and keywords using the specified access tree. Then the data owner uploads the ciphertext including encrypted data and encrypted keywords (namely index) to the cloud server.
- The data owner is responsible for issuing the identity secret keys to the data user after verifying the data user's identity.

Cloud server: It is responsible for storing encrypted data and performing search over the secure searchable index without knowing any information about data and the search query. When the cloud server is given the index and a search query, a successful search can be completed if and only if attributes contained in the trapdoor satisfy the access tree and the index match the query.

Data user: The data user's work includes two parts:

- When a data user joins in the system, the data user can use the issued secret keys to generate the legal search trapdoor, then sends it and a search query towards the cloud server.
- Once receiving the search result from the cloud server, the data user can access the encrypted data. If the attribute set satisfies the access tree, the encrypted data can be decrypted successfully by the data user.

AAs: They are in charge of distributing the attribute secret keys of the data user when AAs are given the data user's attribute set.

3.2 Algorithm Definition

Define $I = \{att_1, ..., att_n\}$ to be the universe attributes. This scheme consists of the following algorithms:

- *Setup* $(\lambda, N, I) \rightarrow (pp, PK_i, MK_i, PK_s, MK_s)$:

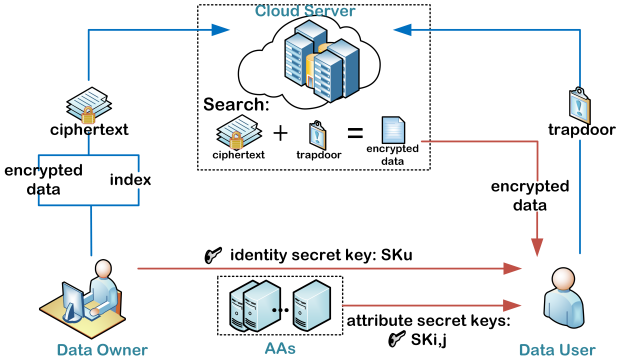


Figure 1: System model

  - *Initialization:* The system is produced at this stage. It inputs security parameter λ, the number of the authorities N and the attribute universe I, then returns the public parameters pp.
  - *AAs-setup:* Each AA_i generates the public and secret key pairs $(PK_i, MK_i)_{i \in \{1...N\}}$.
  - *Server-setup:* The cloud server generates the server public and secret key pair (PK_s, MK_s).

- *KeyGen* $(GID, S, MK_i) \rightarrow (blinded\_D_{i,j}, D_{i,j}, SK_u, SK_{i,j})$: The algorithm is divided into two phases: *DO-keygen* and *AAs-keygen* as follows.

  - *DO-keygen:* The data owner runs the randomized algorithm. It inputs the user's GID, then outputs the identity secret key SK_u.
  - *AAs-keygen:* Each AA_i runs the randomized algorithm. It inputs MK_i and the user's attribute set S, then outputs blinded secret keys $blinded\_D_{i,j}$, $D_{i,j}$ and attribute secret keys $SK_{i,j}$.

- *Encryption* $(M, K, pp, PK_s, W) \rightarrow (CT, \tau)$: The data owner runs the randomized algorithm. It inputs the message M, keywords K, pp, PK_s and the access tree W, then outputs the ciphertext CT, including the encrypted data $< \widetilde{C}, C, C_x, C_x^{'} >$ and the index τ.

- *Trapdoor* $(K, SK_{i,j}, blinded\_D_{i,j}, D_{i,j}, PK_s) \rightarrow (T_K)$: The data user runs the randomized algorithm. It inputs the keywords K, $SK_{i,j}$, $blinded\_D_{i,j}$, $D_{i,j}$ and PK_s, then outputs the keywords trapdoor T_K corresponding the access tree, and send it to the cloud server.

- *Search* $(T_K, \tau, MK_s) \rightarrow (CT, \perp)$: The cloud server matches the trapdoor T_K with encrypted keywords τ, then sends the relevant search results to the data user.

- *Decryption* $(CT, SK_{i,j}, SK_u) \rightarrow M$: The data user inputs CT, SK_u, $SK_{i,j}$, then outputs the message M, if the attribute set satisfies the access tree.

3.3 Security Model

3.3.1 Selective-Attribute Ciphertext Attack Game

We define the selective-attribute ciphertext attack experiment between the challenger $\mathcal{C}$ and the attacker $\mathcal{A}$ as follows.

- *Setup:* The attacker $\mathcal{A}$ gives the access structure W to the challenger $\mathcal{C}$. $\mathcal{C}$ runs the Setup algorithm and gives the public parameters pp and public keys PK_i to $\mathcal{A}$.
- *Phase 1:* The attacker $\mathcal{A}$ submits a set of attributes S for trapdoor queries and secret keys queries, where the attribute set S does not satisfy the access structure W. The challenger $\mathcal{C}$ answers secret keys SK for S.
- *Challenge:* The attacker $\mathcal{A}$ submits two equal length keywords w_0 and w_1. $\mathcal{C}$ randomly chooses $\nu \in \{0,1\}$, then an attribute ciphertext for w_ν under W and returns it to $\mathcal{A}$.
- *Phase 2:* Same as Phase 1.
- *Guess:* The attacker $\mathcal{A}$ outputs a guess $\nu^{'}$ of ν.

Definition 6. (Ciphertext privacy) *Our scheme is secure against selective-attribute ciphertext attack if for probability polynomial time attacker $\mathcal{A}$, there is a negligible advantage ϵ such that:*

$$Adv_{\mathcal{A}}^{cp} = |Pr[b = b^{'}] - 1/2| \leq \epsilon.$$

3.3.2 Indistinguishability of Trapdoor Game

We define the indistinguishability of trapdoor experiment between the challenger $\mathcal{C}$ and the attacker $\mathcal{A}$ as follows.

- *Setup:* The attacker $\mathcal{A}$ gives the challenge access structure W to the challenger $\mathcal{C}$. $\mathcal{C}$ runs the *Setup* algorithm and gives the public parameters pp and the sever public key PK_s to $\mathcal{A}$.
- *Phase 1:* The attacker $\mathcal{A}$ submits a set of attributes S for trapdoor queries, where the attribute set S does not satisfy the access structure W. The challenger $\mathcal{C}$ answers secret keys SK for S.
- *Challenge:* The attacker $\mathcal{A}$ submits two equal length keywords w_0 and w_1. $\mathcal{C}$ randomly chooses $\nu \in \{0,1\}$, then computes a trapdoor T for w_ν under W and returns it to $\mathcal{A}$.
- *Phase 2:* Same as Phase 1.
- *Guess:* The attacker $\mathcal{A}$ outputs a guess $\nu^{'}$ of ν.

Definition 7. (Trapdoor indistinguishability) *Our scheme satisfies trapdoor indistinguishability if for probability polynomial time attacker $\mathcal{A}$, there is a negligible advantage ϵ such that:*

$$Adv_{\mathcal{A}}^{ti} = |Pr[b = b^{'}] - 1/2| \leq \epsilon.$$

4 A Concrete Decentralized ABSE Scheme

4.1 Key Distribution Protocol

Firstly, we analysis the weakness to secret-keys-recovery attack in schemes [7, 11, 12]. Two multi-authority ABE schemes were proposed in [7, 11], in which could not resist secret-keys-recovery attack since the secret value r or g^r must be selected by the center authority (CA) and transmitted to each attribute authority (AA) through a secure channel. Combined with g^r and the ciphertext component g^{a_x}, CA can compute $e(g,g)^{ra_x}$ and $e(g,g)^{ra}$, (where a is the secret of root node, and a_x are the secret of leaf nodes), then decrypt successfully without knowing the private keys. It means that once the CA is broken, the whole system will collapse. In addition, in another multi-authority ABE scheme [12] based access tree, the secret value r is divided into r_k, where r_k is selected by multiple different AAs respectively. But the interaction must be performed between one attribute authority AA_i and another attribute authority AA_j $(i \neq j)$, during which r_i and r_j are delivered over secure channels. Similarly, this procedure depends on secure channels. In conclusion, above schemes can not resist to secret-keys-recovery attack once either AA or CA obtains the secret value r or g^r, unless AA and CA are trusted authorities.

To solve above problems, the key distribution protocol is introduced in our scheme. We reconstruct the ABE scheme and extend the KeyGen algorithm to a multiple authority scenario. In this system, the user needs to get the secret key components from a set of AA and the data owner. The key distribution procedure is shown as Figure 2. The detail will be presented as follows.

- *Step 1:* Each AA_i submits its signature $[sig_{AA_i}, s_i]$ to the data owner, where s_i is random seed from AA, which is used for verifying and preventing replay attack.
- *Step 2:* After verifying the AAs, the data owner computes parameter $Pu = P \cdot g^r$ for AAs to calculate the attribute secret keys, where P is the blinded factor and $r \in Z_p$ is selected randomly by the data owner. Once obtaining g^r, whoever can get the attribute secret keys what he/she wants. So it is necessary to blind g^r. Then Pu is encrypted with AA's public key $[enc_{pk-AA_i}, c]$.
- *Step 3:* AA decrypts and gets Pu, then calculates blinded attribute secret keys $blinded\_D_{i,j} = Pu \cdot H(att_j)^{r_{i,j}}$ and sends them to the data owner together with AA's signature.
- *Step 4:* The data owner verifies AA, then sends the unblinded attribute secret keys $D_{i,j}$ to AA.

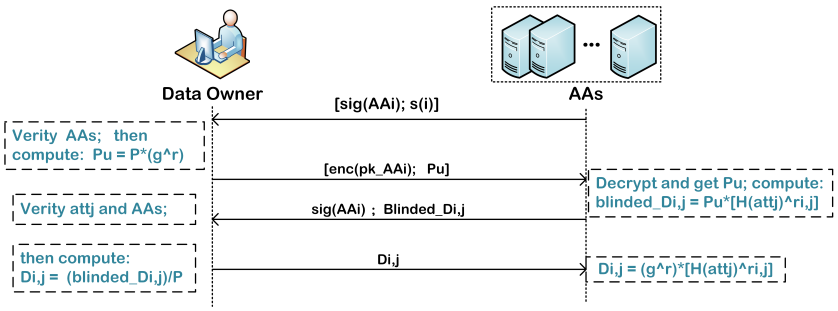


Figure 2: Key distribution protocol

4.2 Scheme Description

In this subsection, we introduce the implementation of each algorithm of the scheme as follows.

4.2.1 Setup Algorithm

- *Initialization:* Let G and G_T be two multiplicative cycle groups of prime order p. $e : G \times G \to G_T$ is the bilinear map, and g is the generator of group G. Choose randomly functions $F : S \times GID \to Z_p$, $H : \{0,1\}^* \to G$, $H_1 : \{0,1\}^* \to G$, $H_2 : \{0,1\}^* \to G_T$. Then it inputs security parameter λ, the number of the authorities N and the attribute universe I, then returns public parameters pp as follows:

$$pp =< e, p, g, G, G_T, F, H, H_1, H_2 >$$

- *AAs-setup:* Each AA_i randomly and independently chooses $y_i \in Z_p$ and random seed $s_i \in Z_p$, and computes $Y_i = e(g,g)^{y_i}$, then outputs the public and secret key pairs: $PK_i =< Y_i >$, $MK_i =< y_i, s_i >$.

- *Server-setup:* The cloud server selects randomly $a \in Z_p$, and returns the server public and secret key pair: $PK_c =< g^a >$, $MK_c =< a >$.

4.2.2 KeyGen Algorithm

- *DO-keygen:* The data owner selects randomly $r, \beta \in Z_p$ and a blinded parameter P, and computes $F(s_i, GID) = h_{i,u}$. Input random seed s_i from AAs, user's GID and $h_{i,u}$. Output the identity secret key SK_u.

$$SK_u = g^{\frac{r+\sum_{i=1}^{N} h_{i,u}}{\beta}}$$

- *AAs-keygen:*

  – Each AA_i invokes the key distribution protocol. Firstly, it obtains Pu and computes the blinded secret key component $blinded\_D_{i,j}$ as follows:

$$blinded\_D_{i,j} = Pu \cdot H(att_j)^{r_{i,j}}$$

  Then it receives the secret key component $D_{i,j}$. From $blinded\_D_{i,j}$, we can know that if g^r is given to AAs or user directly, any AA or user can forge arbitrary attribute keys since att_j is a binary string and $r_{i,j}$ is a random number. As shown in Figure 2, in order to prevent user from getting g^r, the data owner need to send user the final attribute keys by unblinding $D_{i,j}$.

  – Input MK_i, user's attribute set S, GID and $r_{i,j}$, where $r_{i,j} \in Z_p$ is selected randomly for each attribute $att_j \in S$. Output the attribute secret keys $SK_{i,j}$ as follows:

$$SK_{i,j} :< D_i = g^{\frac{y_i + h_{i,u}}{\beta}}; \forall att_j : D'_{i,j} = g^{r_{i,j}} >$$

4.2.3 Encryption Algorithm

The data owner runs the algorithm as Algorithm 1.

4.2.4 Trapdoor Algorithm

The trapdoor algorithm is run by the data user. Data user chooses randomly $\eta \in Z_p$ and secret value $\alpha \in Z_p$. Input PK_c, $blinded\_D_{i,j}$, $D'_{i,j}$ and keywords K, then output the trapdoor $T_K :< T_1, T_2, T_3, T_4 >$ as follows:

$$T_1 = H_1(K) \cdot H(PK_c{}^{\eta});\ T_2 = g^{\eta};$$

$$T_3 = (blinded\_D_{i,j})^{\alpha};\ T_4 = D'^{\ \alpha}_{i,j}$$

Here,the search trapdoor is structured by using blinded secret key components $blinded\_D_{i,j}$ instead of original secret key components $D_{i,j}$ used to decrypt, which increases the resistance to KG attack. In addition, the cloud server completes the search operation using its own secret value a to resist KG attack by preventing any part of the system from searching.

4.2.5 Search Algorithm

The algorithm is run by the cloud server as Algorithm 2.

4.2.6 Decryption Algorithm

The data user runs the algorithm. It inputs the ciphertext CT and SK_u and $SK_{i,j}$, and returns the result as follows. If the node x is a leaf node and $att_j \in S$, then computes :

$$F_x = \frac{e(D_{i,j}, C_x)}{e(D'_{i,j}, C'_x)} = \frac{e(g^r H(att_j)^{r_{i,j}}, g^{a_x})}{e(g^{r_{i,j}}, H(att_j)^{a_x})} = e(g,g)^{r a_x}$$

Algorithm 1 Encryption:

Begin
Input: pp, PK_i, PK_c, access tree W, message M, keywords K
Select randomly secret value $s \in Z_p$ for the root node t of access tree W
Compute: $\tau_W^{'} = H_2(e(H_1(K), PK_c^s)) \cdot e(g^s, Pu)$, $\tau_W^{''} = g^s, \widetilde{C} = M \cdot \prod_{i=1}^{N} Y_i{}^s, C = g_1{}^s$
while each node x in W(in a top-down manner, starting from t) **do**
  Define k_x be the threshold value of a node x
  if x is the root node t **then**
    Choose randomly a polynomial f_t with degree $d_t = k_t - 1$
    Set $f_t(0) = a_t = s$
    Set randomly d_t other points to completely define f_t
  else
    Choose randomly a polynomial f_x with degree $d_x = k_x - 1$
    Set $f_x(0) = f_{parent(x)}(index(x)) = a_x$
    Set randomly d_x other points to completely define f_x
  end if
end while
while each leaf node x in X (let X be the set of leaf nodes) **do**
  Compute $C_x = g^{a_x}$, $C_x^{'} = H(att_j)^{a_x}$
end while
Output: Ciphertext CT:
$CT :< \widetilde{C} = M \cdot \prod_{i=1}^{N} Y_i{}^s; C = (g^{\beta})^s;$
$\forall x(k_x = 1) : C_x = g^{a_x}, C_x^{'} = H(att_j)^{a_x};$
$\tau : (\tau_W^{'} = H_2(e(H_1(K), PK_c^s))e(g^s, Pu); \tau_W^{''} = g^s) >$
End

when $att_j \notin S$, it returns $\perp$.

If the node x is not a leaf node, for all child nodes z of node x, it outputs F_z. Define S_x be a $k_x - sized$ set of child nodes. If S_x does not exist, it returns $F_z = \perp$. Otherwise, the recursive computation is shown as follows:

$$\begin{aligned} F_x &= \prod_{z \in S_x} blinded\_F_z{}^{\triangle_{i,S_x'}(0)} \\ &= \prod_{z \in S_x} (e(g,g)^{ra_z})^{\triangle_{i,S_x'}(0)} \\ &= \prod_{z \in S_x} (e(g,g)^{rf_{parent(z)}(index(z))})^{\triangle_{i,S_x'}(0)} \\ &= \prod_{z \in S_x} e(g,g)^{rf_x(i)\triangle_{i,S_x'}(0)} \\ &= e(g,g)^{rf_x(0)} = e(g,g)^{ra_x} \end{aligned}$$

Recalling the Lagrange polynomial interpolation, the algorithm can decrypt the encrypted data as follows:

$$\begin{aligned} & \frac{\widetilde{C} \cdot e(SK_u, C)}{e(\prod_{i=1}^{N} D_i, C) \cdot F_t} \\ = & \frac{M \cdot e(g,g)^{\sum_{i=1}^{N} y_i s} \cdot e(g,g)^{(\sum_i h_{i,u} + r)s}}{e(g,g)^{\sum_{i=1}^{N}(y_i + h_{i,u})s} \cdot e(g,g)^{rs}} \\ = & M \end{aligned}$$

The data user can obtain the plaintext M if the attribute set S satisfies the access tree W.

Algorithm 2 Search:

Begin
Input: Trapdoor T_K, ciphertext CT, server secret key MK_c
while each leaf node x in W **do**
  if $att_j \in S$ (let att_j be the attribute related with the leaf node x) **then**
    Compute:
    $blinded\_F_x = \frac{e(T_3, C_x)}{e(T_4, C_x')} = \frac{e((P \cdot g^r H(att_j)^{r_{i,j}})^{\alpha}, g^{a_x})}{e(g^{r_{i,j}{}^{\alpha}}, H(att_j)^{a_x})} = e(g, Pu)^{a_x}$
  else
    Define: $blinded\_F_x = \perp$
  end if
end while
while each non-leaf node x in W **do**
  Define S_x be a $k_x - sized$ set of child node z such that $blinded\_F_z \neq \perp$
  if S_x is not found **then**
    Define: $blinded\_F_x = \perp$
  else
    Compute:
    $blinded\_F_x = \prod_{z \in S_x} blinded\_F_z{}^{\triangle_{i,S_x'}(0)}$
    $= \prod_{z \in S_x} (e(g, Pu)^{a_z})^{\triangle_{i,S_x'}(0)}$
    $= \prod_{z \in S_x} (e(g, Pu)^{f_{parent(z)}(index(z))})^{\triangle_{i,S_x'}(0)}$
    $= \prod_{z \in S_x} e(g, Pu)^{f_x(i)\triangle_{i,S_x'}(0)}$
    $= e(g, Pu)^{f_x(0)} = e(g, Pu)^{a_x}$
    where $S_x^{'} = \{\forall z \in S_x : index(z)\}$, $\triangle_{i,S_x'}$ is the lagrange coefficient
  end if
end while
if $blinded\_F_t = \perp$ (let t be the root node of W) **then**
  return 0
else
  Recursively compute:
  $blinded\_F_t = e(g, Pu)^{f_t(0)} = e(g, Pu)^s$
  Compute $G = H_2(e(T_1 / H(T_2{}^{MK_c}), \tau_W^{''}{}^{MK_c}))$
  if $G = \tau_W^{'} / blinded\_F_t$ hold **then**
    return CT
  else
    return $\perp$
  end if
end if
End

4.3 Security Proof

Theorem 1. *If BDH assumption holds in group (G, G_T), then our scheme is selective-attribute ciphertext attack secure in standard model.*

Proof. Suppose that there is a probabilistic polynomial time attacker $\mathcal{A}$ can attack our scheme with the advantage ϵ. We construct an algorithm $\mathcal{B}$ that can solve the BDH problem with $\epsilon^{'}$. Let $e: G \times G \to G_T$ be a bilinear map, where G is a multiplicative cycle group of prime order p, and g is the generator of G. Given $g^a, g^b, g^c \in G$, the algorithm $\mathcal{B}$ outputs Z. The challenger $\mathcal{C}$ flips a binary coin $\mu \in \{0,1\}$, if $\mu = 1$, $\mathcal{C}$ sets $Z = e(g,g)^{abc}$, otherwise chooses randomly $Z \in G_T$. $\mathcal{B}$ simulates $\mathcal{C}$ and interacts with $\mathcal{A}$ as follows.

- *Initialization:* The attacker $\mathcal{A}$ submits the challenged access tree W and a list of corrupted authorities AA^* to the challenger $\mathcal{C}$. $\mathcal{B}$ chooses randomly $\beta, r, P \in Z_p$ and sets $Pu = P \cdot g^r$, $g_1 = g^\beta$, then runs the Setup algorithm and sends Pu, g_1 to $\mathcal{A}$. For each attribute att_j, $\mathcal{B}$ chooses randomly $d_j, \beta_j \in Z_p$ and computes:

$$H(att_j) = \begin{cases} g^{d_j} & att_j \in S \\ g^{b\beta_j} = B^{\beta_j} & att_j \notin S \end{cases}$$

- *Authority Setup:* $\mathcal{B}$ randomly selects $AA_i^* \in \{AA_1, AA_2, ..., AA_N\} \setminus AA^*$.
  - For $AA_i \in AA^*$, $\mathcal{B}$ selects randomly $y_i \in Z_p$ and computes $Y_i = e(g,g)^{y_i}$. Then, $\mathcal{B}$ selects a random seed s_i for corrupted authorities AA_i. $\mathcal{B}$ sends $< y_i, s_i >$ and $< Y_i >$ to the attacker $\mathcal{A}$.
  - For $AA_i \notin AA^*$, $\mathcal{B}$ selects randomly $y_i \in Z_p$, and computes $Y_i = e(g,g)^{y_i}$. $\mathcal{B}$ selects a random seed s_i for the honest authority AA_i and gives $< Y_i >$ to $\mathcal{A}$.

- *Phase 1:* The attacker $\mathcal{A}$ queries for secret keys corresponding to attribute set S that does not satisfy W. $\mathcal{B}$ chooses randomly a function $F(\cdot)$ and sets parameter $h_{i,u} = F(s_i, GID)$, then computes $SK_u = g^{\frac{r+\sum_{i=1}^{N} h_{i,u}}{\beta}}$ and $D_i = g^{\frac{y_i + h_{i,u}}{\beta}}$. After receiving the key queries, $\mathcal{B}$ firstly defines a polynomial f_x for each node x of W, and f_x could be known completely if node x could be satisfied, otherwise $g^{f_x(0)}$ could be known. $\mathcal{B}$ sets $f_t(0) = a$ for each node x, then defines the final polynomial $Q_x(\cdot) = bf_x(\cdot)$ and $Q_t(0) = ab = s$. $\mathcal{B}$ randomly selects $r_{i,j} \in Z_p$, then computes $D_{i,j}^{'} = g^{r_{i,j}}$ and $D_{i,j}$ as follows:

$$D_{i,j} = \begin{cases} g^{d_j r_{i,j}} & att_j \in S \\ g \cdot g^{b\beta_j r_{i,j}} = g \cdot B^{\beta_j r_{i,j}} & att_j \notin S \end{cases}$$

  Finally, $\mathcal{B}$ returns the secret keys $< SK_u, D_i, D_{i,j}, D_{i,j}^{'} >$ to $\mathcal{A}$.

  Next, $\mathcal{A}$ queries the trapdoor for keyword w and access the random oracle H, H_1. The query process is similar to that of the paper [3], [16].

- *Challenge:* The attacker $\mathcal{A}$ submits two equal length keywords w_0 and w_1. $\mathcal{B}$ randomly chooses $\nu \in \{0,1\}$, and computes $\widetilde{C} = M \cdot \prod_{i=1}^{N} Y_i^s$; $C = g_1^s$; $\forall x (k_x = 1) : C_x = B^{a_x}, C_x^{'} = B^{d_x a_x}$; $\tau_W^{'} = H_2(e(H_1(w_\nu), PK_c{}^s)) \cdot Z; \tau_W^{''} = g^s$. Then $\mathcal{B}$ returns the ciphertext $< \widetilde{C}, C, C_x, C_x^{'} >$ to $\mathcal{A}$.

  Similar to [3], the attacker $\mathcal{A}$ try to analyze query to $H_2(t)$ and the pair $(t, H_2(t))$ in the $H_2(\cdot)$ list. Let $s = c$ and select randomly $\xi_\nu \in Z_p$, then can get:

$$t = e(H_1(w_\nu), g^{ac}) = e(g,g)^{ac(b+\xi_\nu)}$$

  $\mathcal{B}$ returns its guess $t/e(g,g)^{ac\xi_\nu}$ for $e(g,g)^{abc}$.

- *Phase 2:* Same as Phase 1.

- *Guess:* $\mathcal{A}$ outputs the guess $\nu^{'}$ of ν. And the advantage of $\mathcal{B}$ is at least $\epsilon^{'} = \epsilon / eq_T q_{H_2}$, where q_{H_2} and q_T are hash function queries to H_2 and trapdoor.

□

Hence, the proof of Theorem 1 is completed. Even an attacker can obtain trapdoor for any keyword what he chooses, the attacker cannot distinguish the encryption of two challenge keywords for which he does not obtain the challenge trapdoor.

Theorem 2. *If HDH assumption holds in group (G, G_T), then our scheme satisfies the trapdoor indistinguishability against a chosen keyword attack.*

Proof. Security proof is similar to that of [16]. Suppose that there is a probabilistic polynomial time attacker $\mathcal{A}$ that can break the trapdoor indistinguishability about our scheme with the advantage ϵ. We construct an algorithm $\mathcal{B}$ that can solve the HDH problem with $\epsilon^{'}$. Let $e: G \times G \to G_T$ be a bilinear map, where G is a multiplicative cycle group of prime order p, and g is the generator of G. Given $g, g^a, g^b \in G$ and $H: \{0,1\}^* \to G$, the algorithm $\mathcal{B}$ outputs Z. The challenger $\mathcal{C}$ flips a binary coin $\mu \in \{0,1\}$. If $\mu = 1$, $\mathcal{C}$ sets $Z = H(g^{ab})$. Otherwise chooses randomly $Z \in G$. $\mathcal{B}$ simulates $\mathcal{C}$ and interacts with $\mathcal{A}$ as follows.

- *Setup:* The attacker $\mathcal{A}$ gives the access tree W to $\mathcal{B}$. $\mathcal{B}$ randomly chooses $\gamma, \alpha \in Z_p$, then sets the server key pairs $(PK_c, SK_c) = (g^{a\gamma}, a\gamma)$ and the user secret key $SK_u = \alpha^{'}$.

- *Phase 1:* The attacker $\mathcal{A}$ submits keyword w_i except for the trapdoor w_0 and w_1 for trapdoor queries. $\mathcal{B}$ randomly chooses $\eta^{'}$, then computes $T_1 = H_1(w) H(g^{a\gamma\eta^{'}})$ and $T_2 = g^{\eta^{'}}$. The secret keys SK_u and $SK_{i,j}$ are generated similar to Theorem 1. $\mathcal{B}$ returns the trapdoor $T_{w_i} :< T_1, T_2, T_3, T_4 >$ to $\mathcal{A}$.

- *Challenge:* The attacker $\mathcal{A}$ submits two equal length keywords w_0 and w_1. $\mathcal{B}$ randomly chooses $\nu \in \{0,1\}$, then computes a trapdoor $T_{w_\nu}{}^*$ as follows:

$$T_1{}^* = H_1(w_\nu) Z; \;\; T_2{}^* = g^{\eta^{'}};$$

Table 1: The comparison of our scheme and related works

Scheme	Authority-number	Center-authority	Searchable-encryption	Whether g^r hidden	KG attack
[2]	Single	✗	✗	✗	✗
[11]	Multi	✓	✗	✗	✗
[14]	Single	✗	✓	✗	✗
[29]	Multi	✗	✓	–	✗
Our	Multi	✗	✓	✓	✓

Table 2: Comparison of computation costs

Scheme	Data Owner (encryption)	Data User (trapdoor)	Cloud Server (search)
[14]	$(3 + K_{index} + 2N_W)E_G + N_W \cdot \mathcal{O}(\mathcal{H}_1)$	$(2N_S + 3)E_G$	$(4+2N_S)e+N_S \cdot E_{G_T}+(N_S+K_{test})E_G$
[29]	$(2 + K_{index} + N_W)E_G + 2\mathcal{O}(\mathcal{H}) + e$	$(2 + N_S)E_G + 2\mathcal{O}(\mathcal{H})$	$(1+2N_S)e+2\mathcal{O}(\mathcal{H})+2E_{G_T}$
Our	$(2+N+2N_W)E_G+K_{index} \cdot \mathcal{O}(\mathcal{H}_1)+\mathcal{O}(\mathcal{H}_2)$	$3E_G+K_{index} \cdot \mathcal{O}(\mathcal{H}_1)+\mathcal{O}(\mathcal{H})$	$(1 + 2N_S)\hat{e} + N_S \cdot E_{G_T} + \mathcal{O}(\mathcal{H}_2) + \mathcal{O}(\mathcal{H})$

$$T_3{}^* = (blinded\_D_{i,j})^{\alpha'};\ T_4{}^* = D'_{i,j}{}^{\alpha'}$$

where Z is a component of the HDH problem. $\mathcal{B}$ returns the trapdoor $T_{w_\nu}{}^*$ to $\mathcal{A}$. Note that $T_{w_\nu}{}^*$ is a valid challenge trapdoor for w_ν, if $Z = H(g^{ab})$.

- *Phase 2:* Same as Phase 1.
- *Guess:* The attacker $\mathcal{A}$ outputs a guess ν' of ν. If $\nu' = \nu$, $\mathcal{B}$ outputs $\mu' = 1$, namely $Z = H(g^{ab})$. Otherwise, it outputs $\mu' = 0$, namely $Z \in G$. The advantage of $\mathcal{B}$ is $\epsilon' = \epsilon/2$ in this game.

□

Hence, the proof of Theorem 2 is completed.

Theorem 3. *If DBDH assumption holds in group (G, G_T), then our multi-authority ABSE scheme satisfies the indistinguishability of messages against chosen plaintext attack.*

Proof. An ABSE scheme is derived from an ABE scheme. The scheme [7] is extended to our scheme, which has been proved to be secure against chosen plaintext attack. So it can be showed that our scheme is also confidential and the detailed proofs are omitted here. □

5 Performance Analysis

The theoretical comparisons between previous schemes and our scheme are conducted in Table 1, and they are shown that this proposed scheme is much more abundant. Compared with schemes [2,11,14,29], our scheme is multi-authority ABSE with some properties, such as access control, keywords search, hiding g^r, resist to KG attack, and resist strongly to secret-keys-recovery attack, so that our scheme is more flexible and efficient.

The paper evaluates the time of encryption, trapdoor and search theoretically as Table 2. Let E_G and E_{G_T} be time of an exponential operation in the group G and G_T respectively. The number of attributes in the access policy and user's attribute set are denoted by N_W and N_S, respectively. N is the number of the attribute authorities and I is the number of attributes in the system. K_{test} shows the number of keywords require for search test and K_{index} shows the number of keywords require for encryption. $\hat{e}$ is the time of computing a pairing function e. The time of calculating a hash function $H(\cdot)$ is denoted by $\mathcal{O}(\mathcal{H})$. $\mathcal{O}(\mathcal{H}_1)$ and $\mathcal{O}(\mathcal{H}_2)$ represent the time of computing hash functions $H_1(\cdot)$ and $H_2(\cdot)$, respectively.

To compare the performance of those schemes more intuitively, we give here an empirical comparison of computation costs in ours, and the results with the latest the work of Miao *et al.* (2017) [14] and the work of Zhu *et al.* (2019) [29]. We conduct our experiments on a Windows machine with 3.40 GHz Intel(R) Core(TM) i3-3240 CPU and 4 GB RAM. The code uses Pairing Based Cryptography library to achieve the access control scheme, which supports pairing operation. Type A pairings are used in the simulation, which are constructed on the curve over the field for some prime q. The pairing is symmetric, where the order of groups is 160 bits, the base field size is 512 bits. All that the length of an element in each group G and the target group G_T is set to 512 bits. The results in Figure 3 reveal the computation costs of the encryption, trapdoor and search algorithm, which the time grows linearly with the number of attributes involved in the access policy and the user's attribute set in ours. On the whole, our scheme has a comparative advantage.

6 Conclusion

Our scheme finds a way to extend the single-authority ABKS scheme to decentralized multi-authority ABKS scheme, which realizes flexible access control and keyword search without a trusted CA. Especially, the search trapdoor is structured by blinded secret key components in our scheme instead of original secret key components used for decryption, which increases the resistance to KG attack. In addition, vulnerable item g^r is hidden by blinded it based on the key distribution protocol to resist secret-keys-recovery attack.

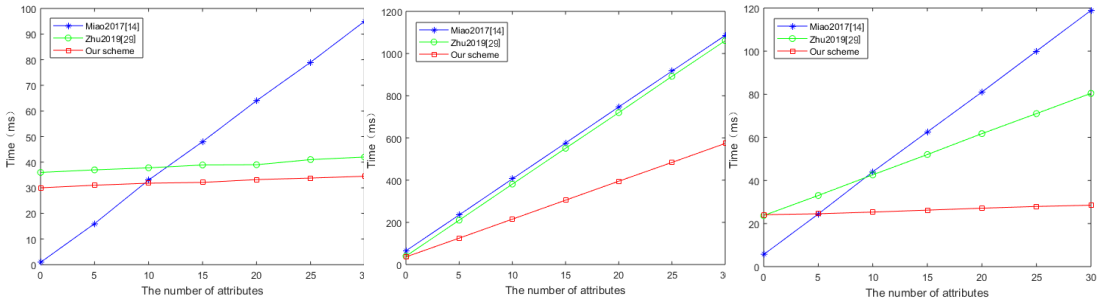


Figure 3: The time cost of the encryption, search and trapdoor algorithm

Acknowledgments

This work was supported in part by the National Cryptography Development Fund under Grant (MMJJ20180209), the National Key Research and Development Program of China under Grants No. 2017YFB0802002, International S&T Cooperation Program of Shaanxi Province No. 2019KW-056, the Plan For Scientific Innovation Talent of Henan Province under Grant 184100510012, and the Program for Science and Technology Innovation Talents in the Universities of Henan Province under Grant 18HASTIT022.

References

[1] M. Abdalla, M. Bellare, D. Catalano, and *et al.*, "Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions," in *Annual International Cryptology Conference*, pp. 205–222, 2005.

[2] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *IEEE Symposium on Security and Privacy (SP'07)*, pp. 20–23, 2007.

[3] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and *et al.*, "Public key encryption with keyword search," in *Advances in Cryptology*, pp. 506–522, 2004.

[4] M. Chase and S. S. M. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in *Proceedings of the ACM Conference on Computer and Communications Security*, pp. 121–130, 2009.

[5] S. M. Chow, "New privacy-preserving architectures for identity attribute-based encryption," in *Dissertation*, 2010. ISBN: 978-1-124-33119-5.

[6] H. Cui, Z. Wan, R. H. Deng, G. Wang, and Y. Li, "Efficient and expressive keyword search over encrypted data in cloud," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 3, pp. 409–422, 2018.

[7] N. Doshi and D. Jinwala, "Hidden access structure ciphertext policy attribute based encryption with constant length ciphertext," in *The 2nd International Conference on Computer and Communication Technology*, pp. 515–523, 2011.

[8] T. Feng and J. Guo, "A new access control system based on CP-ABE in named data networking," *International Journal of Network Security*, vol. 20, no. 4, pp. 710–720, 2018.

[9] T. Feng, X. Yin, Y. Lu, J. Fang, and F. Li, "A searchable cp-abe privacy preserving scheme," *International Journal of Network Security*, vol. 21, no. 4, 2019.

[10] J. Han, Y. Yang, J. K. Liu, and *et al.*, "Expressive attribute-based keyword search with constant-size ciphertext," *Soft Computing*, vol. 22, pp. 5163–5177, 2018.

[11] X. F. Huang, Q. Tao, B. D. Qin, and *et al.*, "Multi-authority attribute based encryption scheme with revocation," in *IEEE 24th International Conference on Computer Communication and Networks*, pp. 1–5, 2015.

[12] T. Jung, X. Y. Li, Z. Wan, and *et al.*, "Control cloud data access privilege and anonymity with fully anonymous attribute-based encryption," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 190–199, 2015.

[13] V. Kuchta and O. Markowitch, "Multi-authority distributed attribute-based encryption with application to searchable encryption on lattices," in *International Conference on Cryptology in Malaysia*, pp. 409–435, 2017.

[14] Y. Miao, J. Ma, X. Liu, and *et al.*, "Practical attribute-based multi-keyword search scheme in mobile crowdsourcing," *IEEE Internet of Things Journal*, vol. 5, no. 4, 2017.

[15] H. Qian, J. Li, Y. Zhang, and J. Han, "Privacy-preserving personal health record using multi-authority attribute-based encryption with revocation," *International Journal of Information Security*, vol. 14, no. 6, 2015.

[16] H. S. Rhee, J. H. Park, W. Susilo, and *et al.*, "Trapdoor security in a searchable public-key encryption scheme with a designated tester," *Journal of Systems and Software*, vol. 83, no. 5, pp. 763–771, 2010.

[17] Y. Rouselakis and B. Waters, "Efficient statically-secure large-universe multi-authority attribute-based encryption," in *International Conference on Financial Cryptography and Data Security*, pp. 315–332, 2015.

[18] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp.457–473, 2005.

[19] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Proceeding of IEEE Symposium on Security and Privacy*, pp. 44–55, 2000.

[20] H. Wang, X. Dong, and Z. Cao, "Multi-value-independent ciphertext-policy attribute based encryption with fast keyword search," in *IEEE Transactions on Services Computing*, pp. 1–1, 2017.

[21] H. Wang, D. He, J. Shen, and *et al.*, "Fuzzy matching and direct revocation: A new CP-ABE scheme from multilinear maps," *Soft Computing*, vol. 22, pp. 2267–2274, 2018.

[22] S. Wang, D. Zhao, and Y. Zhang, "Searchable attribute-based encryption scheme with attribute revocation in cloud storage," *PLoS ONE*, vol. 12, no. 8, 2018.

[23] K. Yang, K. Zhang, X. Jia, and *et al.*, "Privacy-preserving attribute-keyword based data publish-subscribe service on cloud platforms," *Information Sciences*, vol. 387, pp. 116–131, 2017.

[24] H. Yin, J. Zhang, Y. Xiong, and *et al.*, "CP-ABSE: A ciphertext-policy attribute based searchable encryption scheme," *IEEE Access*, vol. 7, no. 99, 2019.

[25] L. Zhang, G. Hu, Y. Mu, and *et al.*, "Hidden ciphertext policy attribute-based encryption with fast decryption for personal health record system," *IEEE Access*, no. 99, pp. 1–1, 2019. DOI: 10.1109/ACCESS.2019.2902040.

[26] L. Zhang, J. Ren, and *et al.*, "Privacy-preserving multi-authority attribute-based data sharing framework for smart grid," *IEEE Access*, vol. 8, pp. 23294–23307, 2020.

[27] L. Zhang and Y. Shang, "Leakage-resilient attributebased encryption with CCA2 security," *International Journal of Network Security*, 2019. DOI: 10.6633/IJNS.2019-838. (`http://ijns.jalaxy.com.tw/contents/ijns-v22-n6/ijns-2020-v22-n6-p838-0.pdf`)

[28] L. Zhang and H. Yin, "Recipient anonymous ciphertext-policy attribute-based broadcast encryption," in *International Conference on Information Systems Security*, pp. 329–344, 2018.

[29] B. Zhu, J. Sun, J. Qin, and *et al.*, "Fuzzy matching: Multi-authority attribute searchable encryption without central authority," *Soft Computing*, vol. 23, pp. 527–536, 2019.

Biography

Juan Ren received the B.S. degree in mathematics from Nanchang Hangkong University, China, in 2017. She received the M.S. degree in mathematics from Xidian University, China, in 2020. Her current interests include applied cryptography and cloud security.

Leyou Zhang received the M.S. and Ph.D. degrees from Xidian University, in 2002 and 2009, respectively. He is currently a Professor with Xidian University. His current research interests include cryptography, network security, cloud security, and computer security.

Baocang Wang received the B.S. degree in Computational Mathematics and the Application Software, the M.S and the Ph.D. degrees in cryptography from Xidian University in 2001, 2004, and 2006, respectively. He is currently a professor with the School of Telecommunications Engineering, Xidian University. His main research interests include public key cryptography, wireless network security, and cloud computing security.

Public Key Infrastructure Traditional and Modern Implementation

Ohoud Albogami, Manal Alruqi, Kholood Almalki, and Asia Aljahdali
(Corresponding author: Aisa Aljahdali)

College of Computer Science and Engneering, Unversity of Jeddah, Saudi Arabia
(Email: aoaljahdali@uj.edu.sa)

(Received Mar. 25, 2020; Revised and Accepted Oct. 10, 2020; First Online Feb. 16, 2021)

Abstract

The public key infrastructure (PKI) method is used to implement strong authentication, data encryption, and digital signatures. The PKI traditional approaches use certificate authorities (CAs) or web of trust (WoT) models; these approaches have security flaws. An emerging solution for constructing secure PKIs is blockchain. Blockchain is a distributed public ledger that works as transaction records. The development of blockchain-based PKIs has been proposed in several studies. In theory, blockchain meets many PKI requirements and addresses some security problems of traditional approaches. This paper explains the traditional and blockchain-based methods for implementing PKI and discusses their advantages and disadvantages. This paper also analyzes PKI approaches by comparing their features and limitations based on several criteria.

Keywords: Blockchain; Certificate Authority (CA); Public Key Infrastructure (PKI)

1 Introduction

A public key infrastructure (PKI) is the primary building block of many applications that rely on secure and reliable authentication, such as digital signatures and encryption for email, smart cards, and network connections. A PKI ensures that a particular entity is bound to its public key, usually by relying on trusted key servers maintained by certificate authorities (CA) [23]. These authorities issue a certificate for a domain or person that publicly and verifiably binds this entity to a specific key. A standard format for such certificates is X.509 [10]. Traditional PKI setups are mostly centralized and face some problems, such as malicious certificates that can remain undetected and allow attackers to act as a man in the middle [26].

Similarly, the revocation of keys relies on a centralized list maintained by only a few entities, implies a significant amount of trust put into a relatively small CAs. In recent years, the misuse of trust has led to distrusting certificates from specific CAs altogether [13].

One approach toward more transparency in managing certificates has been proposed by [15] and is referred to as log-based PKIs. The proposed public log allows the audit of CA activity for the process of issuing, managing, and revoking certificates but does not provide a fully decentralized approach. The advent of blockchain technology has advanced the concept of such a public log. Blockchain technology presents a mechanism for a public, decentralized, tamperproof, complete, and available list of records. A large number of blockchain-based, decentralized theoretical approaches, for example, [1, 11, 16, 19], have been discussed. They intend to deal with the challenges of traditional PKIs. Implementations of proposed approaches come with different storage types, permission models, and support for certificate formats.

This paper intends to investigate the modern and traditional implementation of PKI, deeply studying the two different approaches and presenting their advantages and limitations.

The paper is organized as follows: Section 2 gives an overview of PKI, Section 3 presents traditional approaches for implementing PKI, while Section 4 investigates the modern approaches for implementing PKI. Finally, a discussion related to the comparison of PKI implementations is presented.

2 Public Key Infrastructure

A PKI is a set of roles, procedures, hardware, and software that manage, distribute, store, and revoke digital certificates and public-key encryption. The goal of a PKI is to securely facilitate the automated transfer of information for various network activities such as sending and receiving emails, internet banking, and e-commerce. PKI confirms the identity of the parties involved in the communication and validates the information being transferred for activities where multiple rigorous proofs are required, not for simple passwords that are inadequate as authentication methods.

A PKI binds public keys with respective identities of entities (users or organizations). The binding is estab-

lished through registration and issuance of certificates that may be carried out by an automated process or under human supervision, depending on the assurance level of the binding [10].

A trusted party called a certification authority (CA) can use the PKI element to establish ownership of a public key. CA issues signing certificates that indicate and bind the identity of the certificate subject to the public key contained in the certificate. The CA uses its private key to sign the certificate. The certificate signing process enables the receiver to verify that the public key was not tampered with or corrupted during transit. The CA hashes the contents, encrypts the hash by using its private key, and includes the encrypted hash in the certificate. The receiver verifies the certificate by decrypting the hash using the CA public key, implementing a separate hash of the certificate, and comparing the two hashes. If they match, the receiver can be sure that the certificate and the public key it contains have not been altered.

3 Traditional Approaches for PKI Implementations

Two traditional approaches used to implement the PKI are certificate authority (CA) and a web of trust (WoT). This section discusses both approaches and their advantages and disadvantages.

3.1 Certificate Authority (CA)

A certificate authority (CA) is an approved entity that distributes and manages digital certificates for a network of users. A digital certificate is a digital document that has been signed by the private key of a trusted authority. The digital certificate that CA issues contain the public key and the identity of the owner. The CA validates and authenticates the identity of the user requesting for the certification by verifying if the public key that will be in the certificate belongs to the user who will own this certificate. This process is called certificate validation [3].

Recent research [4,20] called CA-based PKI as centralized PKI because the CA adopts a centralized infrastructure. The users can trust the CA by verifying the CA's signature. Consequently, users will assume that certificate information is accurate, and the public key belongs to the user identified in the certificate. Several web services are protected through keys signed by CAS.

The CA issues a digital certificate to authorize another CA to distribute certificates that can issue a digital certificate for another CA, forming a chain of trust. Certificates can then be traced backward through this chain. The chained CA certificates are called intermediate CA or sub CA certificates. The top-level CA certificate is called a root CA certificate. Self-signed certificates may be used internally in a large company or used by a small company that does not want the expense of using a CA. A CA's root certificate is self-signed by the CA and is used as a trust anchor in certificate chains [3].

3.1.1 The X.509 Certificate

X.509 is a standard for a digital certificate that is widely used in PKI. The X.509 digital certificate structure is shown in Figure 1. Certificate X.509 has different fields, depending on the version used. The required fields for all versions are version number, serial number, name of the entity associated with the public key (subject), issuer name, validity period, and public key. All this information is signed using the CA's private key. To validate a certificate, a relying party uses the CA's public key to verify the signature on the certificate, checks that the time falls within the validity period, and may also consult a server associated with the CA to ensure that the CA has not revoked the certificate [21].

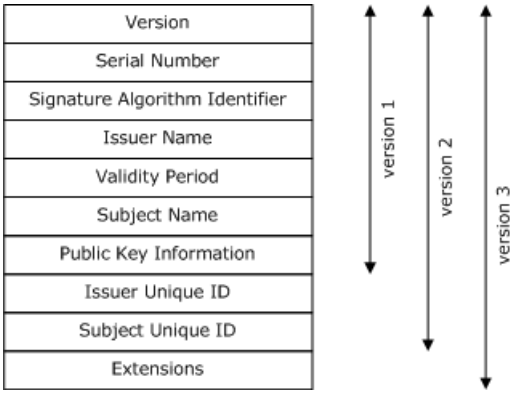


Figure 1: X.509 certificate structure [9]

The advantages of using CA in PKI are as follows:

1) The CA's digital certificate can authenticate the identity of the entity and many enterprise networks and applications using this type of certificate [26].

2) The integrity of the certificate information is guaranteed by verifying the CAs.

3) Integrity: integrity is guaranteed as long as the CA's signature on the digital certificate can be verified.

4) The signature in the certificate also guarantees non-repudiation. Non-repudiation means that the CA who signed the certificate cannot deny it has issued this certificate.

The limitations of using Digital Certificates in public key infrastructure are:

1) CA is vulnerable because of its centralized structure, which could lead to a single point of failure where the whole structure will be affected once a root CA is attacked or tampered with [4,20].

2) There is a concern for the process of certificate verification that uses more than one CA's root public keys. If the attackers add their public keys to that chain of CAs, attackers then issue certificates that will be treated as legitimate certificates [6].

3) CA is highly exposed to different forms of MITM (man-in-the-middle) attacks such as ARP spoofing, DNS spoofing, HTTPS spoofing, and man-in-the-browser.

4) Identification of an anonymous entity that has requested a digital certificate from a CA leaves serious risk for the verifier of the certificate. As a result, the verification process requires a set of verification methods. However, none of these methods can completely guarantee the authenticity of the entity [4,6].

5) In 2017, Symantec, one of the largest CAs, issued a large number of falsified certificates. Google Chrome 70 has stopped support for all certificates issued by Symantec and its affiliates [5].

3.2 Web of Trust (WoT)

In the Pretty Good Privacy (PGP) encryption program, a new concept is introduced named web of trust by Phil Zimmermann in 1991 [18]. The main goal is to authenticate the binding between a public key and the owner of the key. The PKI certificate, which is the centralized hierarchical concept, is only introduced by a CA. Unlike the PKI certificate, WoT is a decentralized public key where each one of the participants in the ecosystem can introduce the public keys of other participants. Any participant in the PGP system is viewed as a CA from the PKI viewpoint. Users of PGP can select the public keys of other users and assign them with different levels of trust. These levels of trust indicate how trustworthy the signature (introduction) of the certificate holder is when he signs public key certificates of other participants. PGP offers four levels of trustworthiness [25]:

1) Full (level 4): The signature of the certificate holder on other users' certificates is fully trusted.

2) Marginal (level 3): The signature of the certificate holder on other users' certificates is trusted to some extent, but it is preferred to find a fully trusted signature.

3) Untrustworthy (level 2): Ignoring signatures on other users' certificates is mandatory if the certificate holder is not trustworthy.

4) Don't know (level 1): There are doubts about the certainty of the certificate holder's signature trustworthiness of other users' certificates. In this case, to send protected information, it is possible to create a "chain of trust" a path from one user to another when confirming the identity is required.

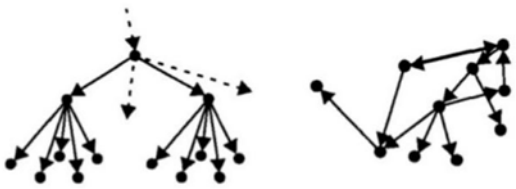


Figure 2: Primary key infrastructure vs. Web of trust [25]

This will cause the publication of a decentralized web of trust for all public keys. As mentioned, each user has a collection of the users' public keys in the ring. In the web of trust, each user encrypts his message, using the recipient's public key and only the private key of the recipient can decrypt the message to ensure confidentiality, and each user digitally signs the information with its own private key when he wants to send a message, then when they verify it using the sender public key to ensure the integrity of the message and that the message was not tampered with and it actually came from the true intended recipient [22].

One of the advantages of using web of trust in public key infrastructure is removing the probability of a central point of failure in PKI's centralized approaches because of its nature as a decentralized system [24]. The limitations of using web of trust in public key infrastructure are:

1) With scalability problem, if a user wants to trust another user not in his group of trusted users, but in one of his group of trusted users, he can simply trust that user and build a secure communication, which is not always a safe way to trust a user.

2) At first, new users must meet in person with another user already in the network of WoT to verify their identities and sign their public key certificates. Therefore, it is difficult for new and remote users to join the network without going through this process [25].

3) In case one of the users lost his private key or the private key gets compromised, WoT provide no way for key revocation. The user has a solution to choose another user on the network to revoke his certificate. It is up to the browser for revocation in some cases [22].

4 Modern Approaches for PKI Implementations

Modern approaches have incorporated blockchain technology with the PKI. This section analyzes two approaches of PKI using blockchain and their advantages and disadvantages.

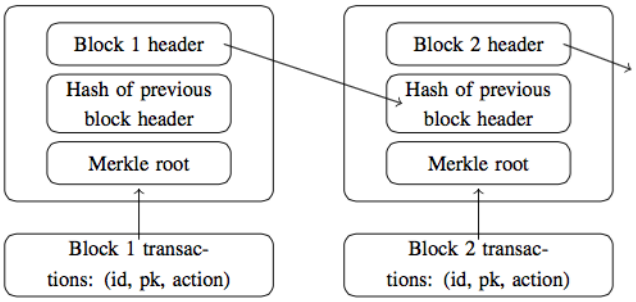


Figure 3: Blockchain PKI structure [9]

4.1 Blockchain-based PKI

A blockchain is a decentralized public ledger to which events are posted and verified by network members. The validation process is called mining in which members compete to complete some proof of work, usually a cryptographic challenge. Blockchain was first introduced as the transaction record for the Bitcoin cryptocurrency. Many blockchains for PKI have been developed, such as the Namecoin blockchain on which Certcoin and PB-PKI are built. Namecoin works as a decentralized domain name server (DNS), which, unlike the Bitcoin blockchain, can store data suitable for larger applications.

The structure of blockchain-based PKI is illustrated in Figure 5. The process of registration, update, and revocation is accomplished by sending a transaction that contains the public key and identity to the blockchain. In the blockchain, each block includes its hash and the hash of previous blocks that creates a reliable ledger that can only be modified by mining the majority of the network. The block can also contain the Merkle root, a hash of a set of transactions. This Merkle root can be used to securely verify transactions, eliminating the need to download the entire blockchain for verification [2]. Blockchain-based PKI has the following advantages:

1) Blockchain is decentralized. No central authority or third-party stores or controls the information. Instead, the information stores and controls the members of the networks.

2) PKIs using blockchain removes the potential points of failure created using CAs.

3) The transaction ledger is unchangeable. Once the transaction is recorded, it cannot be removed or altered.

4) Blockchain-based PKI provides the certificate transparency (CT) property to improve CA-based PKI security through public logging and monitoring of certificates.

5) Blockchain-based PKI also has potential advantages over WoT-based PKI, where the need to establish trust results in a high barrier to entry. The amount of work required to build a web that proves "trustworthiness" to a usefully large proportion of the network is significant. In blockchain-based PKI, entities do not require this web of attesting members, so the work needed to perform as a network member is removed [2].

6) The interaction for a blockchain can be zero-knowledge proofs, where some propositions about the transactions can be proved without revealing all its information [12].

Blockchain-based PKI has the following limitations:

1) Blockchain-based PKI does not provide privacy awareness. Therefore, building a privacy blockchain-based PKI is a complicated task that may have multiple conflicts in its requirements.

2) High resource consumption, such as CPU memory, especially in the mining process [18].

3) Blockchain-based PKIs have a master authority in charge of authentication and trust. The master authority becomes the central part of the network security and the critical point of vulnerability that attackers attack [18].

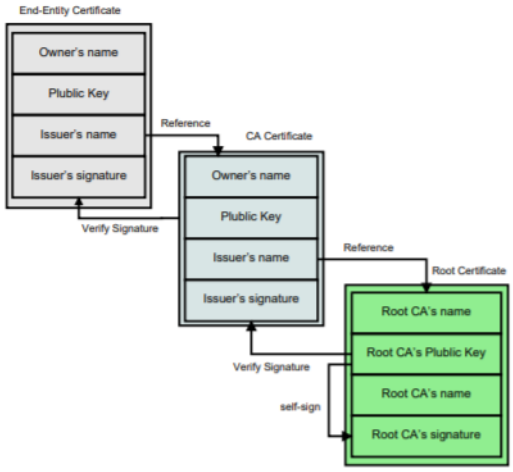


Figure 4: Chain of trust [24]

4.2 Blockchain-based PKI using X.509 Extension

Before introducing blockchain-based PKI, we must briefly discuss the chain of trust to understand types of CAs and to simplify the idea of blockchain. We already discussed CA. We defined CA as a third-party issuing certificates to anyone or any website to guarantee the confidentiality and integrity of the communicating entities' messages [24]. When a user logs in to any social media platform through

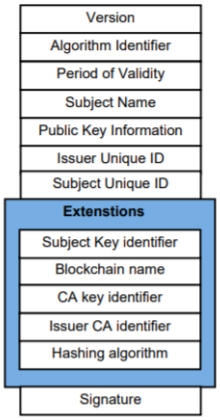


Figure 5: The X.509 hybrid certificate structure [3]

a browser, the browser first validates the platform certificate. Each browser usually has a list of known CAs already trusted and accepts certificates only from those trusted CAs. Root CA and sub CA, which is trusted by root CA, signatures are the only CAs that can issue a certificate that will be trusted to be used [24].

Figure 4 demonstrates how the chain of trust works. The web browser checks the validity of the end-entity certificates, if it's not issued by trust CA, the browser moves forward to check the validity of the CA that issued the certificate to the end entity, and so on, until the browser finally finds a trusted CA or an error is displayed [24]. Blockchain is blocks linked with each other using cryptography, with each block containing the hash of the previous block, a timestamp, and transaction data. Blockchain is a decentralized approach, so it solves the problem of single points of failure that occur in CAs. Blockchain-based PKI is basically an X.509 certificate (Figure 4) with an extension filed contains information about PKI.

The X.509 hybrid certificate structure works with the three types of certificates mentioned before, root CA, sub CA and end-user CA. Blockchain-based PKI is a hierarchy of hybrid certificates and it contains the following fields: Certificate, issued by, issued to, contract ID, and issuer CA ID. Blockchain-based PKI works as follows: the root CA certificate is issued and signed by the root CA and no issuer CA ID, the sub CA must be issued by the root CA and the issuer CA ID is the root CA contract ID. There could be more than one sub CA between root CA and end-user CA. The end-user CA must be issued by the sub CA and the issuer CA ID is the sub CA contract ID. The end-user CA has no contract ID because of the fact that the end user cannot issue certificates.

Blockchain-based PKI has the following advantages over the traditional PKI:

1) Blockchain-based PKI provides a certificate revocation mechanism, and only the parent CA that issued the certificate has the privilege to revoke the certificate and that makes Blockchain-based PKI reliable; because any modification in the network's nodes every other node will be notified [24].

2) The validation process of CAs and certificates are simple and fast [24].

3) Provides a high level of protection against Man in The Middle attack; because when one CA revokes or publishes a public-key of a website or domain on the blockchain th3e modification will be distributed across thousands of nodes which makes it impossible for anyone to tamper the public-key [24].

Table 1: The blockchain hybrid certificate [24]

Cert.	Issued By	Issued To	CA Contract ID	Issuer CA ID
RootCA	RootCA	RootCA	0x1234xxxx	0x00000000
SubCA	RootCA	SubCA	0x5631xxxx	0x1234xxxx
EndUser	SubCA	End user	-	0x5631xxxx

Blockchain-based PKI has the following limitations:

1) Due to the blockchain nature, as the blockchain's size increases more space needed, which may affect the performance [24].

2) The blockchain operation cost depends mainly on the price of the cryptocurrency, for example: In May 2017 Ether price was 85.43 dollars growing 8 times just in 7 months apart December 2017 to be 729.01 dollars [24].

3) If the user lost his/her account's password of the blockchain platform, his/her account becomes irrevocable and he/she will lose the right to access and modify certificates authority data.

5 Discussion

In this section, we analyze the previous PKI approaches by comparing their features and limitations based on several criteria, including system structure, management framework, validation process, revocation process, certificate transparency, level of protection, scalability, privacy, trust, and performance. Table 2 shows a summary of the compression between PKI approaches.

System Structure: The traditional approach CA-based PKI is centralized since it relies on a trusted third party to control the process of issuing, validating, and revoking the certificate. Therefore, the CA is subject to bottleneck, single point of failure, and different attacks because of its centralized structure.

In contrast, the WoT is a decentralized structure in which each participant can introduce the public keys of other participants. The modern approach is also decentralized based on blockchain technology, where a public ledger's linking identity with the public key is distributed over a peer-to-peer network. Decentralization does not have a single point of failure and solves security issues of the central authority.

Table 2: Comparing the discussed techniques based on different factors

Features-approach	**CA**	**WOT**	**PB-PKI**	**Blockchain-based PKI using X059**
System Structure	Centralized	Decentralized	Decentralized	Decentralized
Management Framework	Organized but no real-time monitoring	Complex and no real-time monitoring	Real-time monitoring	Real-time monitoring using a smart contract
Validation process	Simple and fast	Complicated and time-consuming	Simple and fast	Simple and fast through The Smart contract or Web service
Revocation process	Cumbersome, not instant and revocation lists are not immutable	No way for direct revocation	Revocations instantly and the revocation lists are immutable	Revocations instantly and the revocation lists are immutable
Certificate Transparency (CT)	Does not use CT	Does not use CT	Use CT	Use CT
Security (Level of protection)	Low level of protection and exposed to different attacks	Low level of protection and exposed to different attacks	High level of protection	High level of protection
Trust	Has trust issues	Different levels of trust	Trustable	Trustable
Privacy	privacy	Does not consider privacy	High level of privacy	Does not consider privacy
Scalability	No concerns	not always reliable	significant concerns	significant concerns
Performance	Reasonable	Affected by some factors	Affected by some factors	Affected by some factors

Management framework: CA-based PKI is a popular and commonly used approach compared with other methods. The CA has evolved over the years, which makes the management framework in CA well designed, manageable, and organized. Thus, the management process of CA is more precise and adaptable. However, the CA still does not provide real-time monitoring. WoT is the less popular approach because of the complexity in the framework management and registration process. The modern approaches provide real-time monitoring, but the PB-PKI [2] is not suitable for identity management because of its strict privacy and transparency requirements. The management process of blockchain-based PKI [24] is performed using a smart contract for each CA that makes the management of framework straightforward because the smart contract is stored in the blockchain, accessible to every peer in the network and cannot be tampered with.

Validation process: In traditional approaches, the CA's validation process is considered simple with few steps and not time-consuming. Conversely, the WoT model is complicated and time-consuming because new users must meet in person with another user already in the WoT network. In the modern implementation of PKI, both approaches perform a simple validation process without revealing all information in the certificate validation in [24] using the smart contract or Web service.

Revocation process: The CA can revoke the certificate, but the process of revocation is cumbersome and not instant. The CA's revocation lists are not immutable and can be recreated with a different content. In WoT, it does not have a way for direct revocation. The only one solution is to choose another user on the network to revoke the certificate. For modern approaches, they can revoke the certificate instantly and the revocation lists are immutable.

Certificate transparency (CT): The modern implementations of PKI use the certificate transparency (CT) while the traditional PKI implementations do not use it. The CT is an Internet standard providing public logs that record all certificates issued. CT goals are to monitor, auditing, and detecting mistakenly or maliciously issued certificates [14].

Security (level of protection): The security level of the traditional approaches is considered low since the CA and WoT are highly exposed to different attacks,

such as MITM. In many scenarios, CAs had been attacked and issued falsified certificates. The security level of blockchain is high since it has not been attacked until now. Both blockchain and WoT rely on a decentralized structure. However, blockchain is more secure than WoT because it uses a timestamp, immutable ledger, encryption, and consensus protocol such as proof of work and proof of stack.

Trust: CA has trust issues because it was exposed to different attacks. In some cases, the user or organization needs to trust multiple certification centers. In WoT, the users assign different levels of trust (from one to four) to other users. These levels of trust indicate how trustworthy the signature of the certificate holder is. The modern approaches are trusted for many reasons. such peer-to-peer network, transactions being visible and stored in all peers, and trust given only to the parent CA that issued the certificate.

Privacy: The CAs have some level of privacy, but some of the privacy requirements are not included. WoT and blockchain-based PKI [24] do not consider privacy, and a transaction's information is publicly available to the network participants. On the other hand, PB-PKI's privacy requirements are considered in the design phase, which provides a high level of privacy to PB-PKI.

Scalability: There are significant concerns about the scalability in the blockchain-based PKI because of the increase in the chain's size that may affect other aspects of the blockchain [7].

WoT scalability is not always reliable. The users can trust and join other users, not in their group of trusted users. For example, if user A has B in his trusted group and B has C in his trusted group, then user A can trust C. The scalability of CA is better and more efficient when compared with the other approaches.

Performance: The performance here means the time consumed, the consumption of resources, and storage overhead. The CA's performance is reasonable in terms of the consumption of time and resources. CA-efficient storage keeps certificates on individual devices. PKI-based blockchain has factors that affect performance, such as the decentralization system, peer-to-peer networks, and consensus algorithms in which the participants perform most of the work. Thus, blockchains cannot ensure fast and stable data transfer as centralized systems CA.

In the end, the most critical question is, what is the best implementation of PKI' From our point of view, security is the most important thing to consider when configuring a PKI. The system needs a guaranteed and secure management of public keys. In a modern approach, the PKI model using blockchain technology provides a higher level of security than other approaches and removes the potential points of failure created by the use of CAs. Indeed, the modern approach faces some limitations and challenges. However, blockchain is a recent technology that has gained much attention. We believe that blockchain's scalability and performance issues will be overcome soon, especially because of the emergence of a new generation of blockchain 3.0 [17] that focuses on solving these problems.

6 Conclusion

The PKI method is used to implement strong authentication, data encryption, and digital signatures. The traditional approaches of PKI use CAs and WoT models. These approaches have security flaws. An emerging solution to constructing secure PKIs is blockchain. This paper investigates the modern and traditional implementations of PKI. It studies these approaches and presents their advantages and limitations. The paper also provides a comparison between all approaches based on various criteria, such as system structure, management, revocation, validation, privacy, security, and performance. For future research, we will conduct experiments for all approaches against several criteria, evaluate their security, and measure their performance.

References

[1] N. Alexopoulos, J. Daubert, M. Muhlhauser, S. M. Habib, "Beyond the hype: On using blockchains in trust management for authentication," in *IEEE Trustcom/BigDataSE/ICESS*, 2017. DOI: 10.1109/Trustcom/BigDataSE/ICESS.2017.283.

[2] L. M. Axon, and M. Goldsmith, "PB-PKI: A privacy-aware blockchain-based PKI," in *International Conference on Security and Cryptography*, 2016. (`https://doi.org/10.5220/0006419203110318`)

[3] P. Black, R. Layton, "Be careful who you trust: Issues with the Public Key Infrastructure," in *The Fifth Cybercrime and Trustworthy Computing Conference*, 2014. DOI: 10.1109/CTC.2014.8.

[4] Y. Chu, *et al.*, "SS-DPKI: Self-signed certificate based decentralized public key infrastructure for secure communication," in *Digest of Technical Papers - IEEE International Conference on Consumer Electronics*, 2020. DOI: 10.1109/ICCE46568.2020.9043086.

[5] DigiCert, *Replace Your Symantec SSL/TLS Certificates.* (`https://www.digicert.com/blog/replace-your-symantec-ssl-tls-certificates/`)

[6] C. Ellison and B. Schneier, "Ten risks of PKI: What you're not being told about public key infrastructure," *Computer Security Journal*, vol. 16, no. 1, pp. 1-7, 2000.

[7] I. Eyal, *et al.*, "Bitcoin-NG: A scalable blockchain protocol," *Cryptography and Security*, 2015. arXiv:1510.02037.

[8] K. Isirova, O. Potii, "Decentralized public key infrastructure development principles," in *IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, pp. 305-310, 2018.

[9] Mike Jacobs, Michael Satran, *X.509 Public Key Certificates*, May 31, 2018. (`https://docs.microsoft.com/en-us/windows/win32/seccertenroll/about-x-509-public-key-certificates`)

[10] A. T. Kaimov, A. T. Kaimov, *Public Key Infrastructure: A Survey*, 2018. (`https://repository.kbtu.kz/xmlui/handle/123456789/75`)

[11] E. Karaarslan and E. Adiguzel, "Blockchain based DNS and PKI solutions," *IEEE Communications Standards Magazine*, vol. 2, no. 3, pp. 52-57, 2018.

[12] A. E. Kosba, A. Miller, "The blockchain model of cryptography and privacy-preserving smart contracts," in *IEEE Symposium on Security and Privacy (SP'16)*, 2016. DOI: 10.1109/SP.2016.55.

[13] D. Kumar, Z. Wang, M. Hyder, J. Dickinson, G. Beck, D. Adrian, J. Mason, Z. Durumeric, J. A. Halderman, M. Bailey, "Tracking Certificate Misissuance in the Wild," in *IEEE Symposium on Security and Privacy (SP'18)*, 2018. DOI: 10.1109/SP.2018.00015.

[14] B. Laurie, "Certificate transparency," *Communications of the ACM*, vol. 57, no. 10, pp. 40-46, 2016.

[15] B. Laurie, A. Langley, E. Kasper, *Certificate Transparency*, RFC 6962, 2013. (`http://www.rfc-editor.org/info/rfc6962`)

[16] R. Longo, F. Pintore, G. Rinaldo, M. Sala, "On the security of the blockchain BIX protocol and certificates," in *International Conference on Cyber Conflict*, 2017. DOI: 10.23919/CYCON.2017.8240338.

[17] D. D. F. Maesaa, P. Mori, "Blockchain 3.0 applications survey," *Journal of Parallel and Distributed Computing*, vol. 138, pp. 99-114, 2020.

[18] A. Moinet, B. Darties, J. L. Baril, "Blockchain based trust & authentication for decentralized sensor networks," *Cryptography and Security*, 2017. arXiv:1706.01730.

[19] H. Orman, "Blockchain: The Emperors New PKI?," in *IEEE Internet Computing*, vol. 22, no. 2, pp. 23-28, 2018.

[20] B. Qin, *et al.*, "Cecoin: A decentralized PKI mitigating MitM attacks," *Future Generation Computer Systems*, vol. 107, pp. 805-815, 2020.

[21] J. Vacca, *Computer and Information Security Handbook*, 2009. eBook ISBN: 9780080921945.

[22] S. Wilson, "Some limitations of web of trust models," *Information Management & Computer Security*, vol. 6, no. 5, pp. 218-220, 1998.

[23] P. W. Wong, N. Memon, "Secret and public key image watermarking schemes for image," *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1593-1601, 2001.

[24] A. Yakubov, *et al.*, "A blockchain-based PKI management framework," in *IEEE/IFIP Network Operations and Management Symposium*, 2018. DOI: 10.1109/NOMS.2018.8406325.

[25] A. Yakubov, W. M. Shbair and R. State, "BlockPGP: A blockchain-based framework for GPG key servers," in *The Sixth International Symposium on Computing and Networking Workshops*, pp. 316-322, 2018.

[26] J. Yu, M. Ryan, "Evaluating web PKIs," in *Software Architecture for Big Data and the Cloud*, 2017. DOI:10.1016/B978-0-12-805467-3.00007-7.

Biography

Ohoud Albogami received her bachelor's degree in information technology (IT) from King Abdul-Aziz University (KAU), KAS, in 2017. She is currently a master's Student at Jeddah University (JU), KAS. Her current research interest includes clustering, Trust computing systems in vehicular ad hoc networks, and blockchain.

Manal Alruqi graduated from King Abdul-Aziz University with a Bachelor of Science in Information Technology. She is currently a master's student in Computer Science at Jeddah University. Her research interests include Blockchain Technology, Distributed Systems Security, Artificial Intelligence, and Machine Learning.

Kholood Almalki received her undergraduate degree in Information Technology at King Abdul-Aziz University, Faculty of Computing and Information Technology, in 2017. Currently, she is pursuing a master's degree in Computer Science at Jeddah University. She has published a scientific paper in Human-Computer Interaction titled "Anti-procrastination Online Tool for Graduate Students Based on the Pomodoro Technique."

Asia Othman Aljahdali received her Ph.D. degree in computer science at Florida State University in 2017. And a master's degree in information security in 2013. Later on, she worked at King Abdul-Aziz University as assistance Professor. Then, she worked at university of Jeddah as an assistance professor in cybersecurity department. Currently, beside her academic work, she works as cybersecurity consulate for the administration of cybersecurity in Jeddah university. Her current research interests include information security, cryptography, data hiding, network security, IoT security, and Cloud security.

An Enhanced Differential Private Protection Method Based on Adaptive Iterative Wiener Filtering in Discrete Time Series

Dan Zheng, Lei Meng, Shoulin Yin, and Hang Li
(Corresponding author: Hang Li)

Software College, Shenyang Normal University
Shenyang 110034, China
(Email: 8871346@qq.com; 352720214@qq.com; lihangsoft@163.com)

(Received Nov. 4, 2019; Revised and Accepted May 15, 2020; First Online Feb. 16, 2021)

Abstract

Although many proposed researches on differential privacy protection in correlation time series have made great progress, there are still some problems. Because different methods are based on different models and rules. There is no uniform attack model, their privacy protection intensity cannot be compared and measured horizontally. This paper designs an attack model for the differential privacy in correlation time series based on adaptive iterative wiener filtering. Experimental results show that the attack model is effective and provides an uniform measurement for the privacy protection with different methods.

Keywords: Adaptive Iterative Wiener Filtering; Differential Privacy Protection; Time Series

1 Introduction

Time series is a sequence formed by chronological arrangement of certain statistical indicators. As an important form of data storage and distribution, time series are ubiquitous in many fields. Data mining of time series can obtain rich information, which is of great significance for government decision-making, enterprise management and public services [3, 6].

Data mining [22, 25] for time series can bring a lot of conveniences, because the time series contains personal sensitive information, the results of data mining may reveal personal privacy. Therefore, how to publish personal data and ensure that sensitive information is not leaked becomes the focus of researchers. In response to the above problems, Yin [26] proposed a differential privacy protection framework. It had a strict mathematical axiomatization model and was independent of the attacker's background knowledge. It was an important privacy protection method in the current privacy protection field.

Differential privacy mechanism [8, 11] is essentially a noise disturbance mechanism, which initially aims at the privacy leakage of independent data. It defines a global sensitivity function, namely the maximum impact of a single record on the data set, and calculates the noise level added to the original data according to the global sensitivity function. In correlation data, correlation will increase the global sensitivity of differential privacy. If the noise is designed according to the original differential privacy mechanism, the corresponding noise level will increase leading to the reduction of data availability.

Therefore, designing a differential privacy protection mechanism that satisfies the security and availability of correlation data has become the important emphasis. As a typical correlation data, time series contains some differential privacy protection methods which are mainly divided into two categories:

1) Modeling-based method;

2) Transformation-based method [12, 27].

The modeling-based method reconstructs the sensitivity function by establishing a correlation model. The correlation model mainly includes Markov [19], Bayesian [7] and other probability models and coefficient matrix model. There are two main methods based on transform, one is to transform the time series of correlation into an independent sequence of another domain. The representative algorithms are discrete Fourier transform (DFT) [1] and discrete wavelet transform (DWT). The other extracts the correlation characteristics of time series and is characterized by a set of independent characteristics. The representative algorithm has the data feature extraction method such as principal component analysis [13]. The existing correlation time series differential privacy protection method reconstructs the sensitivity function by establishing different correlation models or data transformations to reduce the correlation of time series to bring additional noise.

Aiming at the above two problems, Naskar [17] presented DNA encoding and channel shuffling for secured encryption of audio data. Naskar [15] showed a secured key-based (k, n) threshold cryptography scheme, where key as well as secret data was shared among set of participants. Then a robust image encryption technique using dual chaotic map was proposed in reference [18]. And there are some other proposed methods [2, 14, 16, 28]. Therefore, this paper designs an attack model for the correlation time series differential privacy. According to the principle of filtering in signal processing, an adaptive iterative wiener filter is designed to filter out the noise added by the correlation time series differential privacy protection mechanism and calculate the change of privacy protection intensity of the existing methods before and after filtering to provide their privacy protection intensity with an unified measurement. The contributions of this paper mainly include the following three aspects.

- Since the Laplace noise of the existing method design is independent and identically distributed, unlike the time series with correlation, this provides an opportunity to design an attack model to filter out part of the noise.
- In order to filter out the Laplace noise in the noisy sequence, an optimal filter is designed as the attack model, which increases the probability of the attacker's success.
- The attack model proposed in this paper can provide an unified measurement for the privacy protection strength of each correlation time series differential privacy protection method.

The remainder of this paper is organized as follows. Section 2 introduces some related works. In Section 3, we give the differential privacy protection definition which will be used in the later. We detailed describe the attack model in Section 4. Experiments and analysis are conducted in Section 5. Conclusions and remarks are given in Section 6.

2 Differential Privacy Protection

2.1 Differential Privacy

The main idea of differential privacy [10, 20] is to add noise to each record in data set D, so that the probability of data leakage is controlled within a certain range. The formal definition of differential privacy is as follows.

Definition 1. *$\varepsilon - difference$ privacy. Given the random algorithm K, and all possible output sets S of K. For a given data set D and any adjacent data set D' with a maximum difference of one record, if algorithm K satisfies,*

$$Pr[K(D) \in S \leq e^{\varepsilon} \times Pr[K(D') \in S]. \quad (1)$$

Algorithm K provides $\varepsilon - difference$ privacy protection for the output results. Through privacy budget ε, the query results of the adjacent data sets D and D' with a maximum difference of one record are indistinguishable within a certain probability range. Where, if ε is smaller, the data security is higher.

Since the differential privacy mechanism is essentially a noise disturbance mechanism, it is generally adopted by Laplace mechanism to add noise into the original data set, which satisfies $\varepsilon - difference$ privacy.

Definition 2. *Laplace mechanism. For the query function $f : D \to R$, the random algorithm K provides $\varepsilon - difference$ privacy protection.*

$$K(D) = f(D) + Lap(\lambda). \quad (2)$$

where $Lap(\lambda)$ is the noise obeying the Laplace distribution, and λ is calculated as,

$$\lambda = \frac{\Delta f}{\varepsilon} \quad (3)$$

Where the global sensitivity Δf measures the maximum change in output S after removing a record in D, which is defined as follows.

Definition 3. *Global sensitivity. For the query function $f : D \to R$, the global sensitivity of f is:*

$$\Delta f = \max_{D,D'} ||f(D) - f(D')||_p \quad (4)$$

where R represents the real space of the map, p denotes the Norm distance measuring Δf.

2.2 Problem Statement

The correlation of time series will increase the global sensitivity of differential privacy. However, existing methods only propose various privacy protection models from the perspective of decreasing the global sensitivity of correlation time series. In fact, the correlation of sequences can be used by attackers to improve the probability of successful attack. And the probability of successful attack will increase.

The time series is processed by differential privacy to obtain the noisy sequence X'. After querying X, X' and the adjacent sequence X" of the original time series, the probability density distribution curves $K(X)$, $K(X')$ and $K(X$") of the query results are respectively obtained. Since X is correlated with each other. The noise N added by the differential privacy mechanism is independently and identically distributed, part of the noise can be filtered out by an adaptive iterative wiener filter to obtain a sequence. Compared with $K(X)$, the probability density distribution curve $K(A)$ of the filtered query result is closer to the probability density distribution curve $K(X)$ of the original time series.

3 Attack Model

3.1 Attack Model Principle

Proposed attack model is a filter-based correlation time series differential privacy attack model. Due to the difference, the noise added by the privacy mechanism is small, so the correlation of the time series before and after filtering does not change so much. Assuming that the correlation of the original time series is known, since the added noise is an independent and identically distributed Laplace sequence, the correlation time series can be regarded as a short-term stationary process, the attacker can filter out part of the noise by the filter to increase the probability of its attack being successful.

The noise sequence of X' is obtained by Laplace mechanism adding noise N in the original time series X, namely,

$$X' = X + N. \tag{5}$$

Where $X = [x(1), x(2), \cdots, x(k)]^T$, $X' = [x'(1), x'(2), \cdots, x'(k)]^T$, $N = [n(1), n(2), \cdots, n(k)]^T$. When an adaptive iterative wiener filter with an impulse response of $h(k)$ is passed, it can be known from the relevant knowledge of signal and filtering in the system that the filtered sequence $\tilde{X}$ is:

$$\tilde{X} = H^T X'. \tag{6}$$

Therefore, the noise filter N is:

$$N' = X' - \tilde{X}. \tag{7}$$

3.2 Solution of Filter Impulse Response

Since wiener filter can filter out the independently distributed noise from the stationary process, the solution process of filter impulse response $h(k)$ is expounded by taking the classical wiener filter as an example, and $h(k)$ can be obtained from the impulse response vector H.

According to the Wiener-Hough equation, the solution of the wiener filter impulse response vector is,

$$P^T = H^T R. \tag{8}$$

Where R is the autocorrelation function of X', P is the cross-correlation function of X and X', $H = [h(1), h(2), \cdots, h(k)]^T$.

Therefore, the impulse response vector of wiener filter is,

$$H = R^{-1} P. \tag{9}$$

Since noise sequence N is white noise sequence, its autocorrelation function is,

$$R_n = \delta(k). \tag{10}$$

So the autocorrelation function R of noisy sequence X' and the cross-correlation function P of original time series and noisy sequence X' can be obtained. The solution formula is,

$$R = E[X' X'^T]. \tag{11}$$

$$P = E[X' X']. \tag{12}$$

Where R is a column vector and P is a square matrix. They are substituted into Equation (9). And we get the impulse response vector H of wiener filter. The detailed algorithm processes are as Algorithm 1.

Algorithm 1 Filter the noise in the noisy time series

1: Input. Original time series X and noise time series X'.
2: Output. Filtered time series $\tilde{X}$.
3: Step 1. Calculate the autocorrelation function R of X, and the cross-correlation function P of X and X'.
4: Step 2. Design an optimal filter impulse response vector H according to R and P to filter noise as much as possible in X'.
5: Step 3. Use the optimal filter to filter out the independent identically distributed noise in X' and obtain the filtered time series.
6: Step 4. Return $\tilde{X}$.

Algorithm 1 describes the working process of the attack model. In here, the most important part is the solution of the impulse response vector H of the filter.

3.3 Intensity Assessment

This section evaluates the change in privacy protection strength of the correlation time series differential privacy protection method under the attack model designed in this paper. Since,

$$\frac{Pr[K(X) \in S]}{Pr[K(X') \in S]} = \frac{Pr[K(N) \in S - X]}{Pr[K(N') \in S - X']} \tag{13}$$

Therefore, it is only necessary to analyze the Laplace noise sequence N through the attack model. After the noise expression, the change of the privacy protection intensity before and after filtering can be obtained.

First, we analyze the noise representation of Laplace noise after passing through the Wiener filter as shown in Theorem 1.

Theorem 1. *The Laplace noise sequence N consisting of m points noise with a scale parameter λ is passed through a Wiener filter with an impulse response vector of H. The output sequence $\tilde{N}$ approximates a Gaussian distribution with a variance of $\frac{2m\lambda^2}{|H|^2}$. That is, $\tilde{N} \propto N(0, \frac{2m\lambda^2}{|H|^2})$.*

Proof. According to the knowledge of filtering in signal processing, after the noise sequence N passing through the adaptive iterative wiener system with impulse response $h(k)$, the output sequence is,

$$\tilde{N} = H^T N. \tag{14}$$

□

Therefore, $\tilde{n}(k) = \sum_{k=-\infty}^{\infty} h(k)n(j-k)$, the impulse response $h(k)$ can be seen as the weight coefficient of $n(k)$. $\tilde{n}(k)$ is the weighted adaptive iterative wiener combination of $n(k)$. According to the properties of the Laplace probability density function, it can be seen that $n(k)$ is an independent and identically distributed Laplace sequence, and the scale parameter is $\tilde{\lambda} = \frac{\lambda}{|H|}$.

Filter generally contains many adders, and it can be known from the central limit theorem, if random variable sequence N with expect μ, variance σ^2 is independently and identically distributed. When the number of variables m of the sequence is sufficiently large, the sum $\sum_{k=1}^{m} n(k)$ of the first m terms of the sequence N approximates a Gaussian distribution with expect $m\mu$, variance $m\sigma^2$, that is,

$$\sum_{k=1}^{m} n(k) \propto N(m\mu, m\sigma^2). \tag{15}$$

Therefore, the Laplace noise sequence N passes through the Gaussian distribution of the output sequence N after the filter. Since the variance of the Laplace noise sequence N is $D[N] = 2\lambda^2$, the variance of output sequence N is,

$$D[\tilde{N}] = 2m\tilde{\lambda}^2 = \frac{2m\lambda^2}{|H|^2} \tag{16}$$

The value of the Laplace noise added in the differential privacy protection method is 0. That is, $\mu = 0$. So

$$\tilde{N} \propto N(0, \frac{2m\lambda^2}{|H|^2}). \tag{17}$$

According to Theorem 1, the noise sequence added by the Laplace mechanism approximates the Gaussian distribution through the output sequence of the Wiener filter, and obtains the mean and variance of the output sequence. The filtered privacy protection strength is shown in Theorem 2.

Theorem 2. *It uses the differential privacy protection mechanism by attacking the correlation of time series model, the intensity of privacy protection* $\varepsilon' = \frac{(R^{-1}P)^2}{2m}\varepsilon^2$, *where* $R = [X'X'^T]$, $P = E[X'X']$. *m is the noise points.*

Proof. Because the Gaussian noise can guarantee $\delta - approximate$ $\varepsilon - difference$ privacy, when $\varepsilon > \sqrt{ln(1/n)/\sigma^2}$, Gaussian noise can guarantee $1/\sigma^2 - unidentifiability$, σ^2 is variance. Because the $\varepsilon > 0$, and the δ value is small, so the inequation is correct in the general case. □

Take count query as an example, its global sensitivity is $\Delta f = 1$, and $\lambda = \frac{\Delta f}{\varepsilon}$, it can be concluded that the privacy protection intensity after filtering is,

$$\varepsilon' = \frac{1}{\sigma^2} = \frac{|H|^2}{2m\lambda^2} = \frac{|H|^2}{2m}\varepsilon^2 \tag{18}$$

And from Equation (9), the privacy protection intensity after attack is,

$$\varepsilon' = \frac{(R^{-1}P)^2}{2m}\varepsilon^2 \tag{19}$$

4 Experiment and Analysis

The experimental environment is Windows 10, 2.2GHz, 62.0 GB of memory and Matlab R2017a. Each experiment runs 500 times. In order to evaluate the effectiveness of the attack model presented in this paper and the existing differential privacy protection methods of correlation time series, four correlation time series data sets are selected from the four fields including transportation, medical care, network and finance (Trajectory, Diabetes, NetTrace and Amazon Access Samples) [24]. The Trajectory has the strongest correlation, and the Amazon Access Samples has the weakest correlation.

4.1 Experiment Process

- For four original time series data set X, we use WT, DFT, CIM, Bayesian and Markov correlation time sequence to perform difference privacy protection respectively. Set the budget of the privacy ε and obtain four noise sequences X'.
- From Equations (11) and (12), we get the autocorrelation function R of X', and the cross-correlation function P of X and X'. It is substituted into type (9) and gets the filter impulse response vector H.
- It is substituted into type (6) and obtains the attacked time sequence $\tilde{X}$.
- Query X, X', $\tilde{X}$ and their adjacent data sets.

  A set of query functions F containing 1000 random linear queries is set, and the total number of query functions is expressed as $|F|$. For Trajectory, the result of the query returns a number of attributes whose value is greater than a fixed value. For Diabetes, the query results return the mean of each indicator. For NetTrace, the query results return the number of internal and external host connections. For Amazon Access Samples, the query results return the number of users that might be supported. The probability of query results is all within the range of $[0, 1]$.
- Calculate the probability density function of the query result. The actual privacy protection intensity and the effective privacy protection intensity after the attack model can be obtained from Equation (20).

$$\frac{Pr[K(X) \in S]}{Pr[K(X') \in S]} \leq e^{\varepsilon} \tag{20}$$

  If the ε is smaller, the intensity of privacy protection is higher. Dwork [26] pointed out that when privacy budget $\varepsilon \leq 1$, better privacy protection can be achieved. Therefore, the range of privacy budget ε set in this experiment is in $[0.1, 0.9]$, and query results are counted every interval 0.2.

- The availability measurement of proposed method in this paper. The probability density function of four time sequence data sets and its adjacent data set query results are calculated. The accuracy of the query results is measured by the mean square error (MSE).

$$MSE = \frac{1}{|F|} \sum_{F_i \in F} (\tilde{F}_i(X) - F_i(X))^2. \quad (21)$$

If the MSE is lower, then the data availability is higher.

4.2 Privacy Protection Strength Evaluation

Firstly, this section calculates the actual privacy protection intensity of the existing differential privacy protection method of correlation time series. Secondly, the effective privacy protection degree of each method is calculated under the attack model.

In order to evaluate the impact of the correlation background knowledge possessed by the attacker on the privacy protection intensity, attackers with all correlation background knowledge and without correlation background knowledge adopt the attack model proposed in this paper and low-pass filter respectively to attack the four time series protected by $\varepsilon - difference$ privacy at the same time. According to the Equation (20), the effective privacy protection intensity ε'' after attacking can be obtained.

It can be seen from the Figure 1, the known correlation background knowledge of the attacker in the $\varepsilon - difference$ privacy after four time series of attacks, under different ε, the actual strength of privacy protection ε'' is lower than irrelevant background knowledge knowledge of the attacker. For example, when the Trajectory sequence is attacked, $\varepsilon = 0.7$, the attacker with the relevant background gets an ε'' of 3.221. The attacker with no relevant background knowledge gets 1.343.

The experimental results show that the attackers with correlation background knowledge and the optimal filter designed in this paper have a higher probability of success than the attackers with low filter and no correlation background knowledge.

The four time series and their adjacent sequences after differential privacy protection of each correlation time series are queried, and the actual privacy protection intensity is calculated according to the experimental Step 5 and expressed by ε'. As it can be seen from Figure 2, the actual privacy protection intensity of all sides is different when protecting the same data set.

For Trajectory, the actual intensity of privacy protection with the Markov approach is 0.345, while it is 0.826 with the WT approach when $\varepsilon' = 0.1$. And similar trends have been observed in other data sets. In addition, it can be observed that when protecting the same time series, the ε' of Markov, Bayesian and CIM is lower than WT

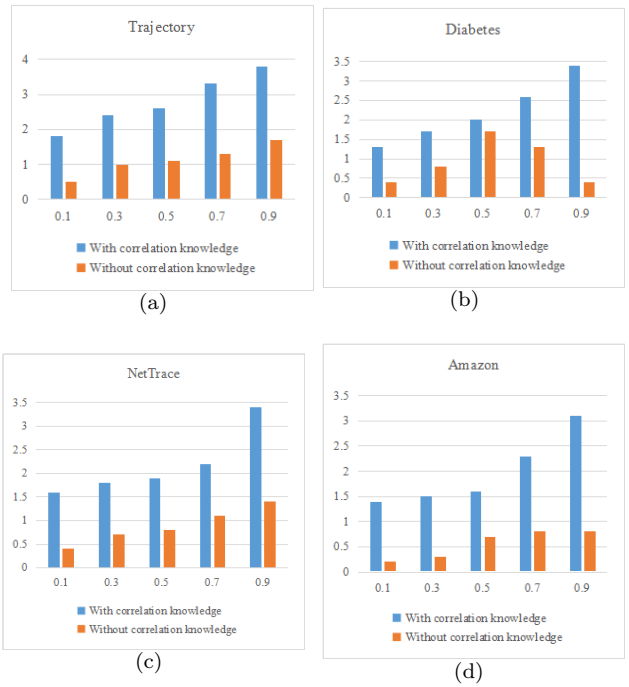


Figure 1: The influence with and without relevant background knowledge on privacy protection (horizontal coordinate is privacy budget and vertical coordinate is privacy protection strength)

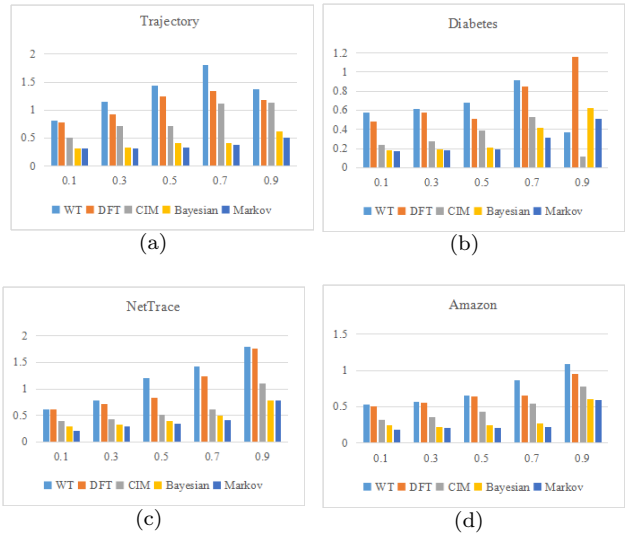


Figure 2: The comparison of actual privacy protection intensity with each method (horizontal coordinate is privacy budget and vertical coordinate is actual privacy protection strength)

and DFT. This indicates that the privacy protection intensity of the model-based approach (Markov, Bayes and CIM) is higher than that of the transformation-based approach (WT and DFT).

This section calculates the effective privacy protection intensity ε'' of each method under the attack model according to experimental Step 5. Compared to Figure 2, the value in Figure 3 is higher.

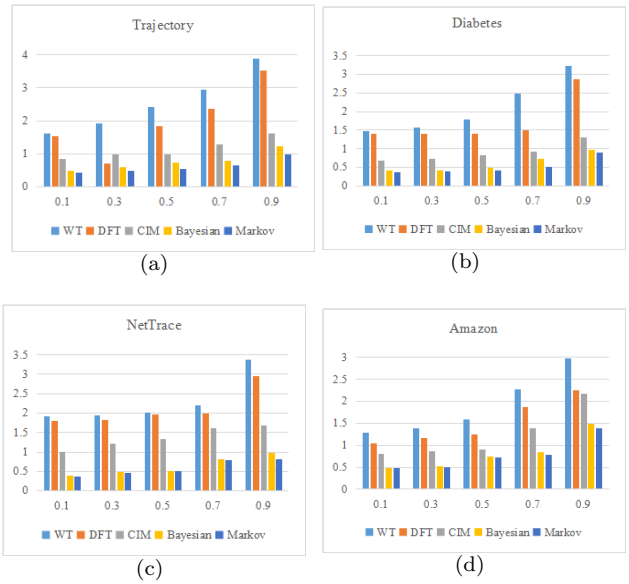


Figure 3: The comparison of actual privacy protection intensity with each method in Step 5 (horizontal coordinate is privacy budget and vertical coordinate is effective privacy protection strength)

For Trajectory, when $\varepsilon = 0.5$, $\varepsilon'' = 0.98$ of CIM, while in Figure 2, the $\varepsilon' = 0.65$ of CIM. It can be inferred from the experimental results that the privacy protection intensity of each method under the attack model is reduced.

In order to make the effect of the attack model proposed in this paper more intuitive, we conduct the experiment of privacy protection intensity with each method before and after the attack changes, when $\varepsilon = 0.7$. The experimental results in Figure 4 are the comparison between the actual and effective privacy protection strengths under the proposed attack model in this paper.

For the Trajectory, the actual privacy intensity ε' of the CIM approach is 1.18, and the effective privacy intensity ε'' is 1.39. Similarly, for Diabetes, the actual privacy intensity ε' of the DFT approach is 0.95, and the effective privacy intensity ε'' is 1.65. Experimental results indicate that the proposed attack model requires a smaller privacy budget.

Then we make comparison experiments with other related methods including LDP [21], TPTID [5], TSL [4] in terms of MSE and time consumption. The results are as Table 1.

Table 1 also shows that the proposed method has better result than other methods.

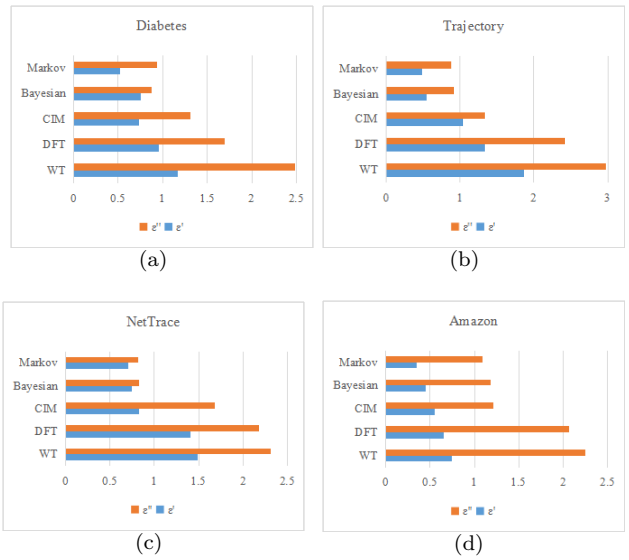


Figure 4: The comparison of privacy protection intensity before and after attack

Table 1: Comparison results

Method	MSE	Time/s
LDP	12.85	5.8
TPTID	10.26	4.6
TSL	9.57	3.7
Proposed	8.95	2.1

5 Conclusions

In order to solve the problem that there is no uniform attack model for the differential privacy protection methods of correlation time series, and the privacy protection intensity of different methods cannot be compared and measured horizontally, this paper designs an attack model for the differential privacy of correlation time series from the perspective of signal processing. This paper assumes that the attacker knows all the background knowledge of correlation, designs an adaptive iterative wiener filtering according to the principle of filtering in signal processing to filter out the noise added in the differential privacy protection mechanism of correlation time series, and calculates the change of privacy protection intensity before and after filtering with the existing protection methods. Experiment results show that under the attack model presented in this paper, the effective privacy protection degree of each method for the correlation time series is greatly decreased. In the future, we will study more private protection methods and apply them into practical engineering.

Acknowledgments

The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] M. Abassi, O. Khlaief, O. Saadaoui, *et al.*, "Real-time implementation of discrete fourier transform phase analysis and fault tolerant control for PMSM in electric vehicles," *Compel International Journal for Computation & Mathematics in Electrical & Electronic Engineering*, vol. 37, no. 1, pp. 432-447, 2018.

[2] C. A. J. Castelo, M. C. Dahowitt, O. P. Almeida, *et al.*, "Comparative determination of the probability of landslide ocurrences and susceptibility in central quito, ecuador," in *The Fifth International Conference on eDemocracy & eGovernment (ICEDEG'18)*, pp. 136-143, 2018.

[3] J. Gao, J. Li, and Y. Li, "Approximate event detection over multi-modal sensing data," *Journal of Combinatorial Optimization*, vol. 32, pp. 1002-1016, 2016.

[4] K. Haiyan, Z. Shuxuan, J. Qianqian, "A method for time-series location data publication based on differential privacy," *Wuhan University Journal of Natural Sciences*, vol. 24, no. 2, 107-115, 2019.

[5] Z. Hu, Y. Jing, J. Zhang, "Trajectory privacy protection method based on the time interval divided," *Computers & Security*, vol. 77, pp. 488-499, 2018.

[6] L. C. Huang, L. Y. Tseng, M. S. Hwang, "A reversible data hiding method by histogram shifting in high quality medical images," *Journal of Systems and Software*, vol. 86, no. 3, pp. 716-727, Mar. 2013.

[7] M. S. Hwang, C. C. Lee, S. K. Chong and J. W. Lo, "A key management for wireless communications," *International Journal of Innovative Computing, Information and Control*, vol. 4, no. 8, pp. 2045-2056, Aug. 2008.

[8] M. S. Hwang, C. C. Lee, W. P. Yang, "An Improvement of mobile users authentication in the integration environments," *International Journal of Electronics and Communications*, vol. 56, no. 5, pp. 293-297, Sep. 2002.

[9] W. Jiang, C. Xie, Z. Zhang, "Wishart mechanism for differentially private principal components analysis," *Computer Science*, 2015. arXiv:1511.05680.

[10] T. Lin, L. Hang, L. Jie, Y. Shoulin, "An efficient and secure Cipher-Text retrieval scheme based on mixed homomorphic encryption and multi-attribute sorting method under cloud environment," *International Journal of Network Security*, vol. 20, no. 5, pp. 872-878, 2018.

[11] J. Liu, S. L. Yin, H. Li and L. Teng, "A density-based clustering method for k-anonymity privacy protection," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 8, no. 1, pp. 12-18, Jan. 2017.

[12] D. Lv, S. Zhu, "Correlated differential privacy protection for big data," in *IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA'18)*, pp. 1011-1018, 2018.

[13] A. Malhotra, I. D. Schizas, V. Metsis, "Correlation analysis-based classification of human activity time series," *IEEE Sensors Journal*, vol. 99, pp. 1-1, 2018.

[14] D. Martin, A. M. Amado, A. G. G. Lvez, *et al.*, "FTIR spectroscopy and DFT calculations to probe the kinetics of β-carotene thermal degradation," *The Journal of Physical Chemistry A*, vol. 123, no. 25, pp. 5266-5273, 2019.

[15] P. K. Naskar and A. Chaudhuri, "Secured secret sharing technique based on chaotic map and DNA encoding with application on secret image," *The Imaging Science Journal*, vol. 64, no. 8, pp. 460-470, 2016.

[16] M. Nazari, S. H. Nazari, F. Zayeri, *et al.*, "Estimating transition probability of different states of type 2 diabetes and its associated factors using Markov model," *Primary Care Diabetes*, vol. 12, no. 3, pp. 245, 2018.

[17] P. K. Naskar, S. Paul, D. Nandy, *et al.*, "DNA encoding and channel shuffling for secured encryption of audio data," *Multimed Tools Appl*, vol. 78, pp. 25019-25042, 2019.

[18] P. K. Naskar, *et al.*, "A robust image encryption technique using dual chaotic map," *International Journal of Electronic Security and Digital Forensics*, vol. 7, no. 4, pp. 358-380, 2015.

[19] S. H. Shin, S. Kim, Y. H. Seo, "Development of a fault monitoring technique for wind turbines using a hidden markov model," *Sensors*, vol. 18, no. 6, pp. 1790, 2018.

[20] Y. Sun, S. Yin, J. Liu, and L. Teng, "A certificateless group authenticated key agreement protocol based on dynamic binary tree," *International Journal of Network Security*, vol. 21, no. 5, pp. 843-849, 2019.

[21] J. Wang, Y. Wang, G. Zhao, *et al.*, "Location protection method for mobile crowd sensing based on local differential privacy preference," *Peer-to-Peer Networking and Applications*, vol. 12, pp. 1097-1109, 2019.

[22] C. C. Wu, S. J. Kao, and M. S. Hwang, "A high quality image sharing with steganography and adaptive authentication scheme," *Journal of Systems and Software*, vol. 84, no. 12, pp. 2196-2207, 2011.

[23] X. Xiao, G. Wang, J. Gehrke, "Differential privacy via wavelet transforms," *IEEE 26th International Conference on Data Engineering (ICDE'10)*, 2010. arXiv:0909.5530.

[24] W. Xiong, Z. Xu and H. Wang, "Privacy level evaluation of differential privacy for time series based on filtering theory," *Journal on Communications*, vol. 38, no. 5, pp. 172-181, 2017.

[25] S. Yin and J. Liu, "A K-means approach for map-reduce model and social network privacy protection," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 7, no. 6, pp. 1215-1221, Nov. 2016.

[26] C. Yin, L. Shi, R. Sun, *et al.*, "Improved collaborative filtering recommendation algorithm based on differential privacy protection," *The Journal of Supercomputing*, vol. 7, pp. 1-14, 2019.

[27] C. Yin, J. Xi, R. Sun, *et al.*, "Location privacy protection based on differential privacy strategy for big data in industrial internet-of-things," *IEEE Transactions on Industrial Informatics*, vol. 99, pp. 1-1, 2017.
[28] T. Zhu, P. Xiong, G. Li, *et al.*, "Correlated differential privacy: Hiding information in non-IID data set," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 2, pp. 229-242, 2015.

Dan Zheng biography. Dan Zheng received the B.Eng. degree from Shenyang Normal University, Shenyang , Liaoning province, China in 2016. Now, she is studying for Master degree in Software College, Shenyang Normal University. Her research interests include Multimedia Security, Network Security, Filter Algorithm and Data Mining.

Lei Meng biography. Lei Meng biography. He is a full associate professor of the Kexin software college at Shenyang Normal University. He has research interests in wireless networks, cloud computing and network security. Email:8871346@qq.com

Shoulin Yin biography. He received the B.Eng. degree from Shenyang Normal University, Shenyang , Liaoning province, China in 2016. Now, he is a doctor in Harbin Institute of Technology. His research interests include Multimedia Security, Network Security, image processing and Data Mining. Email:352720214@qq.com.

Hang Li biography. Hang Li obtained his Ph.D. degree in Information Science and Engineering from Northeastern University. Hang Li is a full professor of the Kexin software college at Shenyang Normal University. He is also a master's supervisor. He has research interests in wireless networks, mobile computing, cloud computing, social networks, network security and quantum cryptography. Prof. Li had published more than 30 international journal and international conference papers on the above research fields. Email:lihangsoft@163.com

Long Sequence Speech Perception Hash Authentication Based on Multi-feature Fusion and Arnold Transformation

Yi-Bo Huang[1], He-Xiang Hou[1], Man-Hong Fan[1], Wei-Zhao Zhang[1], and Qiu-Yu Zhang[2]
(Corresponding author: Yi-Bo Huang)

College of Physics and Electronic Engineering, Northwest Normal University[1]
No. 967, An-ning East Road, Lanzhou 730070, China
(Email: huang yibo@foxmail.com)
School of Computer and Communication, Lanzhou University of Technology[2]
No.287, Lan-Gong-Ping Road, Lanzhou 730050, China
(Received Oct. 23, 2019; Revised and Accepted May 8, 2020; First Online Feb. 16, 2021)

Abstract

To improve the discrimination and robustness of the existing speech authentication and solve low security in the process of mobile communication transmission, a novel long sequence speech perceptual hash authentication algorithm based on multi-feature fusion and Arnold transform is proposed. Firstly, the wavelet low-frequency logarithmic energy spectra of the pre-processed speech signals and the feature matrix of the low-frequency MFCC are extracted. Secondly, the two sets of feature matrices are transformed into binary hash long sequences. Finally, the two long hash sequences are fused into a novel long hash sequence after Arnold encryption to complete the hash matching. The proposed algorithm adopts a hash long sequence, which significantly improves the discrimination of existing algorithms. When each frame of the speech signal is converted into a binary hash sequence of 8 bits, the algorithm's robustness is virtually balanced. Experimental results show that compared with the existing speech authentication algorithms, the proposed algorithm has better comprehensive performance and ensures the security of the hash sequence in the transmission process.

Keywords: Arnold Transform; Discrimination; Perceptual Hash Long Sequence; Speech Authentication; Wavelet Logarithmic Energy Spectrum

1 Introduction

In the face of today's massive data processing requirements, hash technology has drawn much attention on its efficient storage and search capabilities. Speech perceptual hash is mainly explored to map the raw speech into a sequence of binary codes while preserving the similarity structure of the original data. Existing speech authentication algorithms use short hash sequences to easily map multimedia signals with different perceived content to the same perceived hash value, resulting in a low discrimination of the algorithm. Since the opening of network communication channels may cause leakage of important information, the security of hash sequences faces serious challenges. Therefore, it is especially important to improve the discrimination and security of the perceptual hash authentication algorithm [5, 9, 17, 19].

At present, the features extracted from speech signals include short-term energy, short-term zero-crossing rate, Mel-frequency cepstral coefficient (MFCC) [13] , linear prediction coefficient [6], cochleagram [7], spectral entropy [22], discrete wavelet transform (DWT) [23], spectral centroid [21], spectrogram [25], and multiple fusion features. Li *et al.* [14] proposed an audio hash scheme based on non-negative matrix factorization (NMF) of modified discrete cosine transform (MDCT) coefficients. The algorithm has good robustness, especially compression aspects such as MP3 and AAC, but its processing efficiency is greatly reduced. Zhang *et al.* [22] proposed an efficient perceptual hash based on LP-MMSE for speech authentication. The algorithm has highly efficiency, but its anti-collision and performance at the MP3 compression is poor. Jiang *et al.* [11] proposed an audio fingerprinting extraction algorithm based on lifting wavelet packet and improved optimal-basis selection. Although the algorithm has strong robustness and efficiency, the features of the speech data it reflected are fragmentary and have certain limitations. Huang *et al.* [10] proposed a strong robustness hash algorithm of speech perception based on tensor decomposition model. The algorithm is robust against background noise and flexible model building. The discrimination of the algorithm needs to be further improved. In [3], the speech authentication algorithm used a ternary hash sequence instead of a binary hash sequence, and the

hash construct proved to be flexible. The algorithm is not only robust to content preserving operations, but also highly efficient.

In order to ensure the security of the speech authentication algorithm in mobile communication transmission, the encryption methods of the existing algorithms include equal length sequence key encryption, logistic chaotic encryption, measurement matrix combined with logistic encryption, *etc.* These encryption algorithms are relatively complex and require simplification of the encryption algorithm while ensuring its security. Zhang *et al.* [23] combined the measurement matrix with the logistic chaotic sequence. The algorithm obtains better security and efficiency, however, it has poor discrimination. The experimental signal is required to satisfy the sparse condition then the algorithm has no universality. The main reason for the low discrimination are that the existing perceptual hash algorithm is to represent a frame of speech signals with a binary hash sequence of 1 bit "0" or "1", compressing multidimensional features of one-dimensional features produces a shorter hash sequence in lower anti-collision. In [24], each frame of speech signal is represented by a 4 bits binary hash sequence of "0" or "1", which get a good discrimination. The algorithm simply compares the influence of different length hash sequences in the discrimination of the algorithm. It does not deeply study the features of hash long sequences, and does not consider the security of the algorithm.

Aiming at the above problems, we propose a long-sequence speech perceptual hashing authentication algorithm based on multi-feature fusion and Arnold transform. The algorithm in this paper not only considers distinguishability and robustness, but also guarantees the security of hash long sequences. In this paper, the multi-dimensional features of speech signals are expanded. Each frame of speech signal is represented by a binary hashing sequence of 8 bits "0" or "1". The resulting hash-length sequence has high anti-collision. MFCC takes advantage of the non-linear characteristics of human hearing, and the discrete wavelet transform conforms to the frequency analysis characteristics of human ear basilar membrane, which has a good robustness for various content preserving operations. The Arnold transform not only has a simple transformation, but also has a good encryption effect.

2 Related Theory Introduction

2.1 MFCC

The MFCC parameter takes into account the auditory features of the human ear. It transforms the spectrum into a non-linear spectrum based on the Mel-frequency scale, and then converts it to the cepstrum domain. The MFCC parameters have good recognition performance and noise immunity due to the full consideration of human hearing features without any assumptions. The relationship between Mel-frequency and speech frequency is

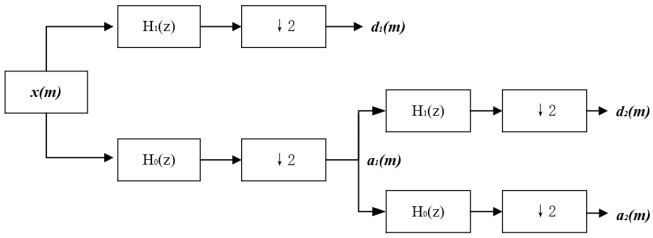


Figure 1: Binary decomposition of signal

expressed in Equation (1):

$$f_{mel} = 2595 \times lg(1 + f/700). \tag{1}$$

Due to the non-linear correspondence between the Mel-frequency and the Hz frequency, the calculation accuracy of the MFCC decreases from the increase in the frequency. Therefore, the algorithm only used the low-frequency MFCC in the application. The medium and high frequency MFCC is discarded. This paper also uses the MFCC of low-frequency.

2.2 DWT

The DWT can decompose the speech into sub-bands of different frequency ranges, and the sub-bands can further divide the smaller sub-bands. The signal needs to be discretized to perform DWT, and the discrete signal is transformed as shown in Equation (2):

$$DWT(j,k) = \frac{1}{\sqrt{|2^j|}} \int_{-\infty}^{\infty} x(t)\psi(\frac{t - k2^j}{2^j})dt, \tag{2}$$

where, 2^j and $k2^j$ are scale parameters and time-shift parameters respectively, and ψ are wavelet functions. The idea of the DWT method is multi-resolution analysis of signals, which essentially decomposes the signals by frequency band. The signal decomposition method can be equal frequency band division, or a binary decomposition can be used [15]. This paper uses binary decomposition. Figure 1 shows the specific decomposition process of the second-order DWT. The proposed algorithm only uses the low-frequency signal of the signal for analysis because some high-frequency signals contain some noise signals.

2.3 Arnold Transform

The Arnold transformation is proposed by V. J. Arnold in the study of ergodic theory, also known as cat face transformation. The Arnold transform is intuitive, simple, and periodic, and is very easy to use. It is widely used for image scrambling encryption. The Arnold transform is divided into equal length Arnold transform and non-equal length Arnold transform. In this paper, the equal-length Arnold transform is adopted. The expression of the two-dimensional Arnold of (x_n, y_n) is:

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} mod(N),$$

where, x_n, y_n represents the position of the value in the matrix before the transformation, $x_{(n+1)}, y_{(n+1)}$ represents the position of the value after the transformation, a, b is a parameter, n represents the number of current transformations, N is the length or width of the matrix (this article takes the same length Arnold transformation, does not discuss the case where the length and width are not equal), mod is the modulo operation.

Since the transformation is an iterative process, if the position (x, y) is converted multiple times, it will return to the original position after T iterations. T is called the transform period and it depends on the parameters a, b and N. The general algorithm chooses a, b and n as the key, and this paper sets $a = 1, b = 1$ and only keeps n as the key. The movement of the values in the matrix has periodicity, and T, a, b and N(the size of the original matrix) are related [2, 20]. Whenever the value changes, a completely different Arnold map is generated. After several multiplications, the correlation of the values in the matrix will be completely chaotic. iterations.

3 The Proposed Scheme

The flow of long-sequence speech perceptual hashing authentication algorithm based on multi-feature fusion and Arnold transform is shown in Figure 2. The algorithm steps in this paper are divided into perceptual feature extraction and hash match.

3.1 Perceptual Feature Extraction

Step 1: Pre-processing. Pre-processing includes pre-emphasis, framing, and windowing. The unprocessed speech signal $s(t)$ is discretized to obtain $s(n)$, and then the pre-emphasis is processed to obtain the signal $x(n)$. Pre-emphasis can increase the features of the speech signals high-frequency components, eliminate the influence of noise in the speech sounding process, and flatten the signal spectrum, which is advantageous to further spectrum analysis.

$$x(n) = s(n) - a * (n-1),$$

where, a is a pre-emphasis coefficient, and its value ranges from 0.9 to 1.0. Then, the processed signal is framed and windowed, where in the window function selects a Hamming window to smooth the edge of the frame. The length of frame is k. It is supposed that the speech $x(n)$ is divided into n frame, and signal $x_i(m) = \{x_i(m)|i = 1, 2, \cdots, n, m = 1, 2, \cdots, k\}$ is obtained.

Step 2: MFCC of low-frequency. First, the fast Fourier transform (FFT) is performed on each frame of the time domain signal $x_i(m)$ to obtain a frequency domain signal $A_i = \{A_i(m)|i = 1, 2, \cdots, n, m = 1, 2, \cdots, k\}$. Then logarithmic function is used to obtain logarithmic energy spectrum of the each frame of the frequency domain signal B_i.

$$B_i(m) = log\left\{|A_i(m)|^2 + 1\right\}.$$

Using the Mel filter bank to find the Mel frequency signal $C_i(j)$.

$$C_i(j) = \sum_{m=1}^{k} B_i(m)H_j(m), 1 \le j \le J,$$

where, $C_i(j)$ is the Mel frequency energy ,$H_j(m)$ represents the bandpass filter, J represents the number of filters, j represents the jth filter. Finally, the coefficient transformed by the Mel filter is subjected to discrete cosine transform to obtain the MFCC.

$$D_i(l) = \sqrt{\frac{2}{J}} \sum_{j=1}^{J} C_i(j) cos\left[\frac{\pi l(2j-1)}{2J}\right], \quad 1 \le l \le M$$

where, M is the total number of lines. The MFCC feature matrix of low-frequency $G_1(L, n)$ is extracted from the MFCC feature matrix $D(M, n)$.

Step 3: Wavelet low-frequency coefficient. The frame signal $x_i(m)$ is subjected to four-stage wavelet decomposition to obtain a low-frequency coefficient matrix $E(M, n)$ (The dimension of the wavelet low-frequency coefficient matrix is the same as the dimension of the MFCC feature matrix). Extracting front L row matrix of wavelet low-frequency coefficient as the feature matrix $F(L, n)$ (Same as low-frequency MFCC feature matrix dimension L).

Step 4: Wavelet low frequency logarithmic energy spectrum. Calculating the logarithmic energy for the wavelet low-frequency feature matrix $G_2(L, n)$.

$$G_2(L, n) = log\left\{|F(L, n)|^2 + 1\right\}.$$

Step 5: Constructing the hash long sequence.

V_1 Constructing the low-frequency Mel-hash long sequence: The arrangement and fusion of each dimension feature $g_i(1, n)$ of the $G_1(L, n)$ matrix to obtain the one-dimensional matrix T_1.

$$T_1 = [g_1, g_2, \cdots, g_L].$$

Binary hashing construction is performed by T_1, the low frequency Mel hashing long sequence $H_1(1, L \times n)$ is obtained.

$$H_1(1, L \times n) = [L_1(1), L_1(2), \cdots, L_1(L \times n)]$$

The previous column vector is subtracted from the current column vector in the parameter matrix. If it is greater than 0, the data of the

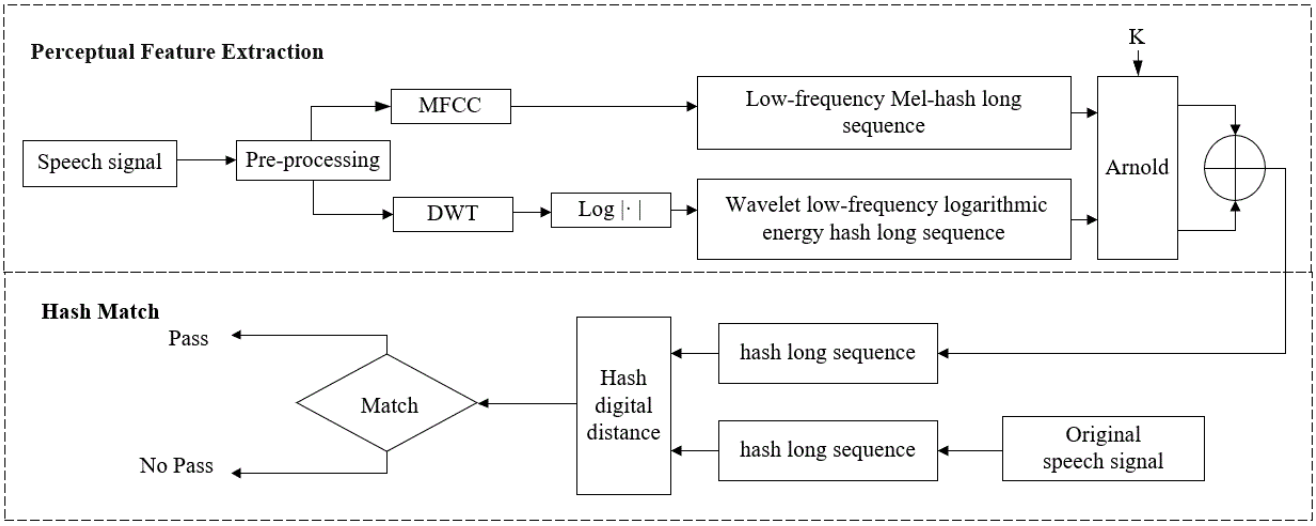


Figure 2: The flow chart of proposed algorithm

current column becomes 1, otherwise the data of the current column is 0. $h_1(1)$ is set to 0.

$$h_1(i) = \begin{cases} 1, & if \quad T_1(i) > T_1(i-1) \\ 0, & Otherwise \end{cases}$$

where, i represents each column of the matrix T_1,$i = 2, \cdots, L \times n$.

V_2 Constructing the wavelet low-frequency logarithmic energy hash long sequence: The wavelet low-frequency coefficient matrix $G_2(L, n)$ is converted into an one-dimensional matrix T_2, and then the hash sequence construction as in V_1 is performed to obtain a wavelet low-frequency logarithmic energy hash long-sequence $H_2(1, L \times n)$.

Step 6: Arnold transform. The low-frequency Mel hash long sequence H_1 is converted into a square matrix P_1 in which the number of rows and columns are both p, and then the transformation of Arnold is performed, and finally the transformed matrix Q_1 is obtained.

$$P_1(p,p) = \begin{bmatrix} P_1(1,1) & P_1(1,2) & \cdots & P_1(1,p) \\ P_1(2,1) & P_1(2,2) & \cdots & P_1(2,p) \\ \vdots & \vdots & \ddots & \vdots \\ P_1(p,1) & P_1(p,2) & \cdots & P_1(p,p) \end{bmatrix}$$

$$Q_1(p,p) = Arnold(P_1(p,p), k)$$

$$Q_1(p,p) = \begin{bmatrix} Q_1(1,1) & Q_1(1,2) & \cdots & Q_1(1,p) \\ Q_1(2,1) & Q_1(2,2) & \cdots & Q_1(2,p) \\ \vdots & \vdots & \ddots & \vdots \\ Q_1(p,1) & Q_1(p,2) & \cdots & Q_1(p,p) \end{bmatrix}$$

where, k is not only the number of Arnold transform, but also the key of the algorithm. The transformed $Q_1(p,p)$ is restored to the one-dimensional matrix $R_1(1, L \times n)$. In the same way, the wavelet low-frequency logarithmic energy hash long sequence H_2 is transformed, and finally the transformed matrix $R_2(1, L \times n)$ is obtained.

3.2 Hash Match

The low-frequency Mel-hash long sequence R_1 and the wavelet low-frequency log energy hash long sequence R_2 are spliced and fused to obtain a hash long sequence $H(1, N) = [R_1, R_2]$ of the speech signal. Speech authentication matches the given speech to the original speech. The normalized Hamming distance $d(:,:)$ of the perceptual hash sequence generated by the speeches $s1$ and $s2$ is BER. The calculation formula is shown as follows:

$$d\left(h_{s1}, h_{s2}\right) = \sum_{i=1}^{N} \left(\left|h_{s1} - h_{s2}\right|\right) / N, \tag{3}$$

where, d is BER, h_{s1} and h_{s2} correspond to the perceptual hash values generated by speech clip $s1$ and $s2$, and N is the length of the perceptual hash values. The probability of the appearance of "0" and "1" sequence is equal in theory, and the average normalized hamming distance is $N/2$.

This paper uses hypothesis testing to evaluate the speech authentication system, which is described as follows:

W_0**:** If two speech clips $s1$ and $s2$ are the same clip, then: $d \leq \tau$.

W_1**:** If two speech clips $s1$ and $s2$ are the different clip, then: $d > \tau$.

By setting the size of matching threshold τ, the perceptual hashing sequence mathematical distance of the

speech clips $s1$ and $s2$ are compared. If the two mathematical distances $d \leq \tau$, and their perceptual content are treated as the same, the certification is passed, otherwise it doesn't pass the certification.

4 Experimental Results and Analysis

The experimental speech data comes from the Texas Instruments and Massachusetts Institute of Technology (TIMIT) speech database and the Text to Speech (TTS) speech database. There are different 1280 speech clips in experimental database recorded. The format of each speech clip is wav with the length 4 s, which is of the form of 16 bits PCM, mono and sampled at 16 kHz. Each speech signal is divided into 361 frames, and each frame of speech signal is represented by an 8 bits 0 or 1 binary hash sequence, and the length of the speech hash sequence is 2888 bits.

The operating experimental hardware platform is Intel(R) Core(TM) i5-7500 CPU, 3.40 GHz, with computer memories of 4G. The operating software environment is MATLAB R2018b of Windows 7 system.

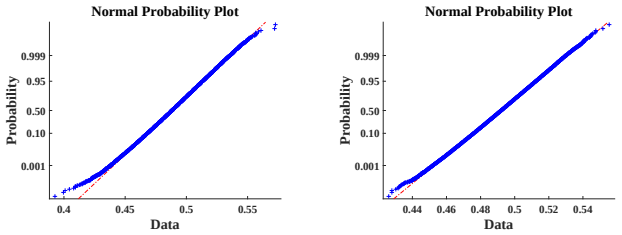
Parameter settings: $n = 361, p = 38, L = 4, M = 16, N = 2888$.

4.1 Discrimination Test and Analysis

Discrimination is mainly used to evaluate the reliability of the algorithm for distinguishing different speech contents read by different or same persons. BER is a basic indicator for testing the digital distance of the hash algorithm and evaluating the performance of the algorithm in binary form. BER refers to the proportion of the number of error bits in the total number of bits, and the normalized Hamming code distance is calculated as Equation (3).

BER of the perceptual hash value of different speech content basically obeys a normal distribution. In this paper, the pairwise comparison of the perceived hash values of 1200 speech segments yields 719,400 BER data. The distribution law is shown in Figure 3.

According to the De Moivre-Laplace central limit theorem, the hamming distance is approximate obeying normal distribution ($\mu = p$, $\sigma = \sqrt{p(1-p)/N}$, N is the number of bits in a hashing sequence). The closer the BER distribution curve is to the normal distribution, the better the randomicity and collision resistance of the perceptual hashing sequence. It can be concluded from Figure 3 that as the length of the sequence increases, the closer the BER curve is to the theoretical curve, the better the discrimination and collision resistance. The algorithm in this paper uses a long hash sequence. When the sequence length is 2888 bits, the overall performance is optimal. According to the central limit theorem of De Moivre-Laplace, the normal distribution parameters of different length hashing sequences can be calculated. The specific parameters are shown in Table 1.



(a) 1444 bits (b) 2166 bits

(c) 2888 bits (d) 3610 bits

Figure 3: The BER normal distribution diagram

As shown in Table 1 and Figure 4, as the length increases, the theoretical value of the normal distribution parameter is closer to the actual value, which means that the algorithm is feasible. When the sequence length of the algorithm in this paper adopts 2888 bits, it has good discrimination and randomness. In order to verify the correctness of the experiment, the FAR and FRR of the algorithm can be calculated by Equations (4) and (5).

$$FAR(\tau) = \int_{-\infty}^{\tau} \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} dx, \quad (4)$$

$$FAR(\tau) = 1 - \int_{-\infty}^{\tau} \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} dx, \quad (5)$$

where, FAR is the false accept rate, FRR is the false rejection rate, τ is the perceived authentication threshold, μ is the BER mean, σ is the BER standard deviation. The higher the FRR value, the weaker the robustness. The higher the FAR value, the worse the discrimination. Table 2 compares the misrecognition rates of different long hash sequence algorithms, and Table 3 compares the misrecognition rate of different algorithms.

Table 1: Normal distribution parameters

parameter	N	Theoretical	Actual
μ	1444 *bits*	0.5	0.4880
μ	2166 *bits*	0.5	0.4919
μ	2888 *bits*	0.5	0.4939
μ	3610 *bits*	0.5	0.4951
σ	1444 *bits*	0.0131	0.0155
σ	2166 *bits*	0.0107	0.0125
σ	2888 *bits*	0.0093	0.0107
σ	3610 *bits*	0.0083	0.0096

As shown in Table 2, the smaller the matching threshold τ is, the smaller the FAR value is. When the matching

Table 2: FAR of hash sequences of different lengths

τ	1444 *bits*	2166 *bits*	2888 *bits*	3610 *bits*
0.20	1.842×10^{-77}	4.262×10^{-121}	6.398×10^{-165}	1.025×10^{-209}
0.25	1.420×10^{-53}	7.236×10^{-84}	2.689×10^{-144}	2.093×10^{-145}
0.30	3.405×10^{-34}	1.410×10^{-53}	4.715×10^{-73}	5.755×10^{-93}
0.35	2.607×10^{-19}	3.237×10^{-30}	3.538×10^{-41}	2.185×10^{-52}

Table 3: FAR of the different algorithms

τ	[23]	[22] [13]	[14]	
0.20	1.405×10^{-24}	1.395×10^{-19}	1.111×10^{-05}	1.019×10^{-21}
0.25	1.215×10^{-17}	5.656×10^{-14}	2.715×10^{-04}	1.696×10^{-15}
0.30	6.166×10^{-12}	2.146×10^{-09}	1.682×10^{-03}	2.080×10^{-10}
0.35	1.874×10^{-7}	7.788×10^{-06}	9.999×10^{-03}	1.916×10^{-06}

threshold $\tau = 0.35$, there are approximately 3.538 speech clips misjudged in 1×10^{41} speech clips, it means that the generated hash sequences will not be the same, which greatly improves the collision resistance of the algorithm. When N is 3610 bits, only 3.092 of each 1×10^{52} speech segment is misidentified. As the length of the hashing sequence increases, the higher the recognition rate. However, the magnitude of the FAR reduction is decreasing, and the robustness is also weakening. The algorithm in this paper fully balances the FAR and the FRR. The hash sequence length 2888 bits is used to achieve the optimal performance of the algorithm. Feature fusion can not only effectively reduce the FAR and FRR, but also improve the robustness and distinguishability of the algorithm.

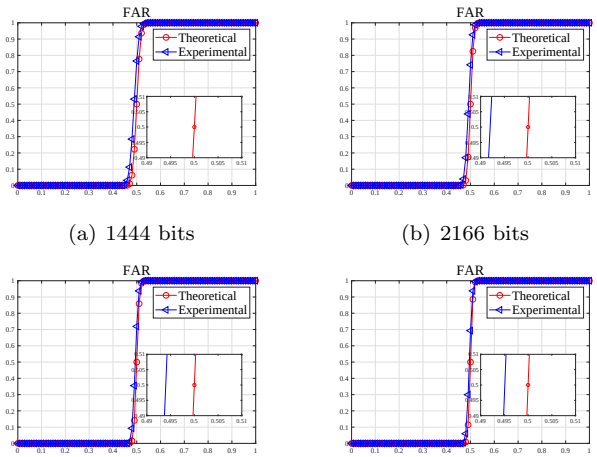
According to the data in Table 2 and Table 3, When τ is 0.35, [13,14,22,23] can completely discriminate between speech and content preserving operations, but FAR of this paper is much lower than the above algorithm. Only 3.538 of the 1×10^{41} speech data is wrong. The algorithm of this paper is 1.9×10^{34} times of [23], 4.5×10^{34} times of [22], 3.5×10^{39} times of [13], and 1.8×10^{35} times of [14]. It is not difficult to conclude that FAR of this algorithm is far lower than other algorithms, which also shows that the algorithm has strong anti-collision and has very good discrimination.

Entropy rate (ER) is a comprehensive evaluation index of discriminative perception hash algorithm, which mainly overcomes the shortcomings of the algorithm being susceptible to sequence size. The larger the value, the stronger the recognition ability, which can be calculated by Equations (6) and (7).

$$ER = -\left[q log_2 q + (1-q) log_2 (1-q)\right], \tag{6}$$

$$q = \frac{1}{2}\left(\sqrt{\frac{|\sigma^2 - {\sigma_1}^2|}{\sigma^2 + {\sigma_1}^2}} + 1\right), \tag{7}$$

where, σ and σ_1 are theoretical and experimental standard deviation of BERs respectively, q is experimental mean value.



(a) 1444 bits (b) 2166 bits

(c) 2888 bits (d) 3610 bits

Figure 4: FAR curve of hashing sequences of different lengths

Table 4: ER of hashing sequences of different lengths

Hash sequence length	ER
1444 bits	0.8762
2166 bits	0.8857
2888 bits	0.8970
3610 bits	0.8931

Table 5: ER of the different algorithms

Algorithms	ER
[23]	0.9187
[22]	0.9732
[13]	0.6794
[14]	0.5449

As can be seen from Table 4, when the hash sequence size is 2888 bits, the entropy rate is the highest, which is the best distinguishing. By comparing Tables 3 and 4, the ER of this algorithm is slightly lower than that of [22,23], it is much higher than [13,14].

Through the above analysis, the algorithm of this paper has a very good distinction.

4.2 Robustness Test and Analysis

In order to evaluate the robustness of the proposed algorithm, content preserving operations are performed on each speech in the speech data, including twelve operations such as echo, resampling, noise, and filters. For the 1200 speech segments in the above speech library, the content preserving operations shown in Table 6 are performed, and various BERs of the proposed algorithm are obtained, as shown in Table 7.

As can be seen from Table 7, it can be seen that the average BER values of the proposed algorithm are less and the maximum value is 0.2240. Therefore it denotes that the proposed algorithm has better robustness. When the algorithm is manipulated on the filter, average BER is relatively high, because the low-pass filter has an effect on the speech spectrum, which causes average BER to be higher than the other content preserving operations. Because the algorithm uses the combination of low-frequency MFCC and wavelet low-frequency logarithmic energy spectrum, the increase and decrease of volume have little effect on the hash sequence of the algorithm, and it indicates that the volume adjustment has little effect on the BER mean of the algorithm. Since the resampling, narrowband noise and echo operations do not change the speech spectrum significantly, the BER average of these operations is small, and the corresponding holding operations are very robust. When the MP3 operation is performed, the BER value is small, indicating that the algorithm has strong robustness to the compression of the speech.

As shown in Table 7, the average BER of various content preserving operations increases in addition to noise, as the length of the hash sequence increases. When adding 30db of noise, the average BER of the hash sequence length 2888 bits is the smallest. When adding 50db of noise, the average BER of the algorithm in this paper is also relatively small. As the length of the hash sequence increases, the robustness decreases. The length of the hash sequence in this paper is 2888 bits, which fully balances the discrimination and robustness.

According to BER data obtained in Table 7, FAR and FRR are obtained, and FAR-FRR curve is plotted. Figure 5 shows the FRR-FAR curves for different length hash sequences. The length of the hash sequence used in this paper is longer than the length of the traditional hash sequence, and the FRR-RAR curves of different sequence lengths do not intersect, which fully demonstrates the scientificity of the hash long sequence.

As shown in Table 8, the average BER of the algorithm are smaller than the three algorithms compared with [6, 14, 22], which can show that the proposed algorithm has good robustness to the content preserving operations. This algorithm works best in volume adjustment and resampling, and it is far lower than the other three algorithms in MP3 operation. In terms of low-pass filter, the effect of this paper is slightly lower than [14, 22]. A pairwise comparison of the perceived hash values of 1200 segments of speech yielded 719,400 BER data.

As indicated in the result in Figure 6(c) and (d), FRR and FAR are intersected. The discrimination and robustness cannot be solved well regardless of the threshold value, so [6, 14] cannot be very wonderful to discriminate between the same processed speech and different content speech. Compared with [22], the obvious advantage of the proposed algorithm is that the threshold can be selected within a large range of 0.280 to 0.440. Although FAR and FRR of [22] does not intersect, the threshold selection does not balance the robustness and discrimination well.

It can be seen from Table 9, the wavelet low-frequency energy coefficient is relatively poor in FIR filtering. The BER reaches 0.3278, while the low-frequency MFCC is only 0.0975. The robustness of the feature fusion is significantly improved, and the BER is reduced to 0.1957. In terms of echo, narrowband noise and MP3 compression, the BER of the low frequency MFCC is above 0.20, which indicates that the MFCC is poor in this respect, and the wavelet low frequency energy coefficient is relatively robust in terms of echo and so on. The algorithm combines these two characteristics, and the BER of the three aspects is relatively low. Overall, this paper has strong robustness.

Comparing Figure 6(a) with Figure 7, FAR of the three algorithms are very close and all have good distinguishability. The FRR-FAR curves of the low-frequency MFCC long-sequence algorithm have intersections, so the threshold selection does not balance the discrimination and robustness well. The threshold of wavelet low-frequency logarithmic energy coefficient long sequence algorithm is 0.440 to 0.450. Compared with the proposed algorithm, the selection space is much smaller. Through the above analysis, the proposed algorithm has better robustness.

4.3 Efficiency Analysis

Efficiency is also a relatively important evaluation standard in speech authentication. In order to evaluate the computational efficiency of the perceptual hash authentication algorithm in this paper, 100 speech segments are selected from the speech library for the calculation of authentication efficiency, and finally the average running time of the algorithm is counted.

As shown in Table 10, the efficiency of the proposed algorithm is relatively good, 1.17 times that of [24], 1.70 times of [6], and 2.89 times of [7]. However, compared with [22], there is some gap, which is 0.10 times slower than [22]. Because the hash sequence is used for speech

Table 6: Content preserving operations

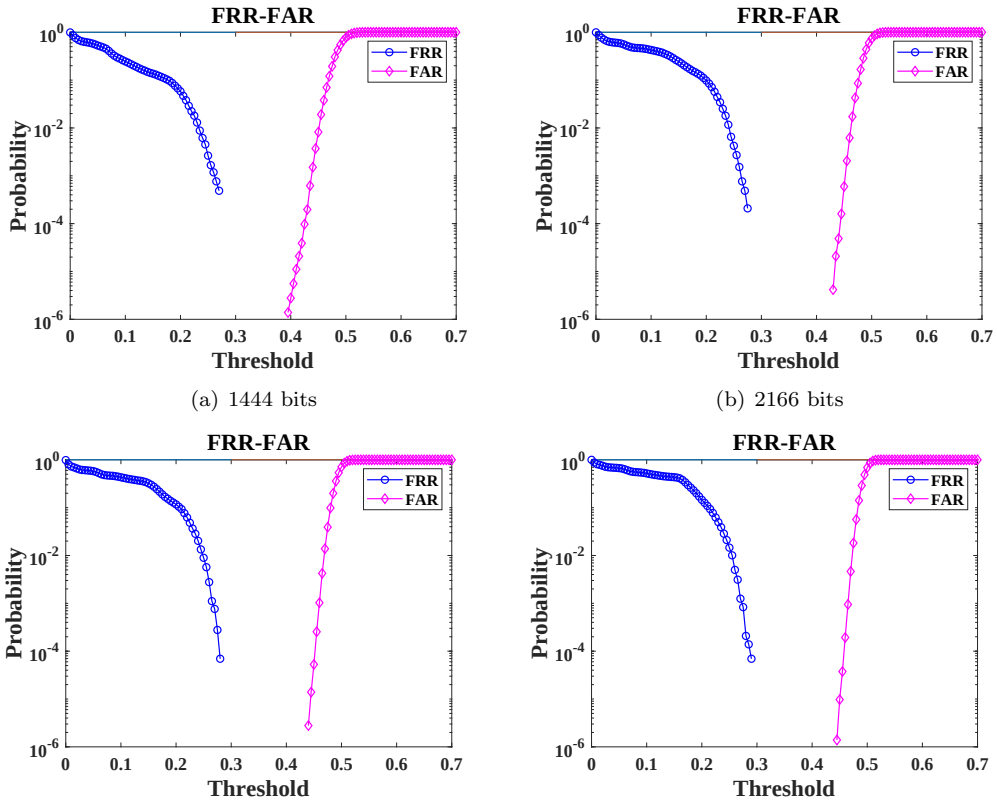
Operating means	Operation method	Abbreviation
Volume Adjustment 1	Volume down 50%	V.1
Volume Adjustment 2	Volume up 50%	V.2
FIR Filter	12 order FIR low-pass filtering, Cutoff frequency of 3.4 kHz	F.I.R
Butterworth Filter	12 order Butterworth low-pass filtering, Cutoff frequency of 3.4 kHz	B.W
Resampling 1	Sampling frequency decreased to 8 kHz, and then increased to 16 kHz	R.8→16
Resampling 2	Sampling frequency increased to 32 kHz, and then dropped to 16 kHz	R.32→16
Echo Addition 1	Superimposed attenuation 30%, delay 100 ms,initial strength were 10% of the echo	E.A1
Echo Addition 2	Superimposed attenuation 60%, delay 300 ms, initial strength were 25% of the echo	E.A2
Narrowband Noise 1	SNR=30 dB narrowband Gaussian noise, center frequency distribution in $0 \sim 4$ kHz	G.N1
Narrowband Noise 2	SNR=50 dB narrowband Gaussian noise, center frequency distribution in $0 \sim 4$ kHz	G.N2
MP3 Compression 1	Re-encoded as MP3, and then decoding recovery, the rate is 48 k	M.48
MP3 Compression 2	Re-encoded as MP3, and then decoding recovery, the rate is 128 k	M.128

Table 7: Comparison of average BER of hashing sequences of different lengths

Hash sequence length	1444 bits	2166 bits	2888 bits	3610 bits
Operating means	Average BER			
V.1	0.0085	0.0087	0.0092	0.0098
V.2	0.0036	0.0039	0.0042	0.0046
F.I.R	0.1705	0.1871	0.1957	0.2017
B.W	0.2021	0.2168	0.2240	0.2291
R.8→16	0.0654	0.0672	0.0685	0.0700
R.32→16	0.0080	0.0077	0.0078	0.0080
E.A1	0.0482	0.0566	0.0624	0.0671
E.A2	0.1234	0.1562	0.1684	0.1807
G.N1	0.1525	0.1359	0.1320	0.1322
G.N2	0.0427	0.0411	0.0418	0.0433
M.48	0.0727	0.1309	0.1607	0.1892
M.128	0.0109	0.0117	0.0126	0.0136

Table 8: Comparison of average BER of different algorithms

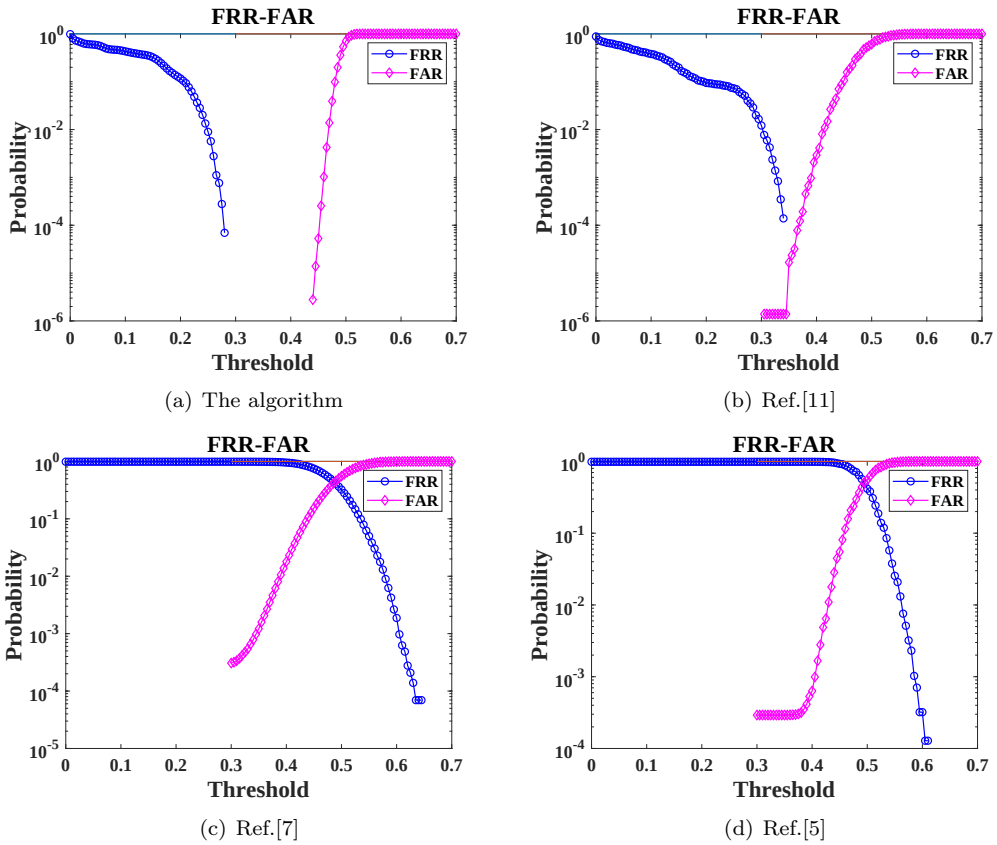
Algorithm	Proposed algorithm	[22]	[6]	[14]
Operating means	Average BER			
V.1	0.0092	0.0047	0.1761	0.0630
V.2	0.0042	0.0455	0.1469	0.0002
F.I.R	0.1957	0.1248	0.3668	0.1821
R.8→16	0.0685	0.0074	0.1567	0.0217
R.32→16	0.0078	0.0910	0.3766	0.0128
E.A2	0.1684	0.1109	0.2132	0.1700
G.N1	0.1320	0.1570	0.3883	0.0346
M.48	0.1607	0.2952	0.4835	0.4852
M.128	0.0126	0.2248	0.4817	0.4851



(a) 1444 bits (b) 2166 bits

(c) 2888 bits (d) 3610 bits

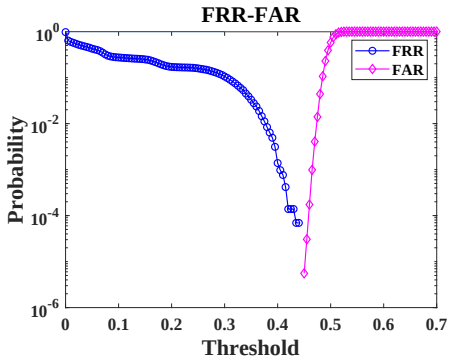
Figure 5: The FRR-FAR curve of the content preserving operations of different length hashing sequences



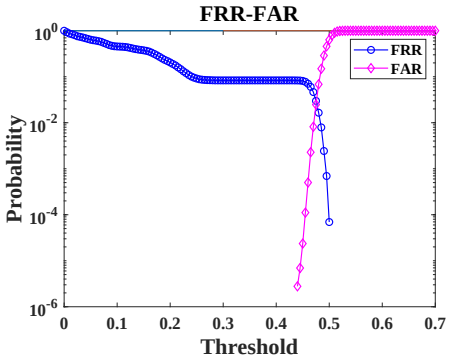
(a) The algorithm (b) Ref.[11]

(c) Ref.[7] (d) Ref.[5]

Figure 6: The FAR-FRR curves of different perceptual hashing algorithm



(a) Wavelet low frequency logarithmic energy



(b) Low frequency MFCC

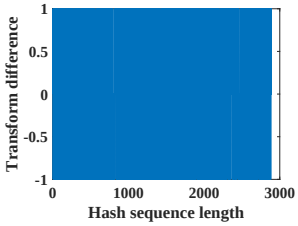
Figure 7: Comparison of the same hash sequence length FRR-FAR curves under different features

authentication, the algorithm structure is relatively complex and the running time is relatively long, which leads to the relatively low efficiency of this paper. Although shorter hash sequences are adopted in [6, 7, 24], its algorithm structure is complex, resulting in low-efficiency.

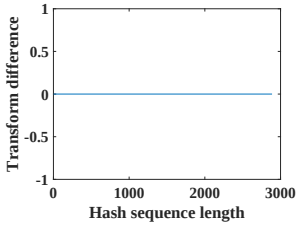
In summary, the proposed algorithm of this paper has certain efficiency and meets the requirements of real-time voice communication quality.

4.4 Security Analysis

In order to improve the security of the algorithm, the Arnold transform is used for scrambling encryption, and the number of scrambling is used as the key. In order to verify the security of the algorithm, a piece of speech of

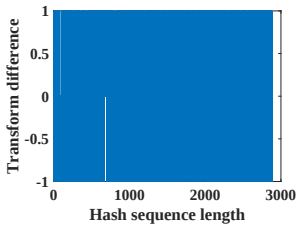


(a)

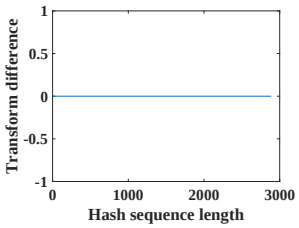


(b)

Figure 8: Hash sequence difference graph. (a) Arnold transform hashing sequence difference graph, (b) Arnold inverse transform hashing sequence difference graph



(a)



(b)

Figure 9: Hash sequence difference graph. (a) Different key hashing sequence difference graph, (b) Same key hashing sequence difference graph

the speech library is randomly selected for testing.

After the Arnold transform, the correlation between the original sequences is disrupted. After the inverse Arnold transform, the state of the original sequence can be restored. Arnold transform encryption is not only secure, but also flexible. In Figure 8(a), The difference between the transformed sequence and the original sequence is 1, 0, -1, the difference between the two is large, indicating that the algorithm is more secure. In Figure 8(b), there is no difference between the inverse transformed sequence and the original sequence, and the difference between the two is all zero, indicating that the original sequence can be restored intact after the inverse transformation by Arnold.

As shown in Figure 9, if the keys are different, the original hash sequence cannot be obtained after the Arnold inverse transforms from the encrypted sequence. Only when the same key is used, the encrypted sequence obtains the original hash sequence. Figure 9 further demonstrates that the algorithm is highly secure and has good encryption effect.

After the hash matrix is scrambled by the Arnold transform, thus realizing the initial hiding of information. At the same time, the number of scrambling can be used as the key of the hash matrix, thereby further enhancing the security and confidentiality of the algorithm, and ensuring the security of the speech signal transmission in the channel.

5 Conclusions

In this paper, a long sequence speech perceptual hash authentication algorithm based on multi-feature fusion and Arnold transform is proposed, which solves the problem of discrimination, robustness and security existing in speech authentication. Not only the discrimination of the algorithm has been greatly improved, but also has a strong comprehensiveness. Especially in the case of volume adjustment and resampling, it has strong robustness and solves the problem of poor MP3 compression robustness. In addition, this paper uses Arnold transform to scramble and encrypt the transmitted signal to improve the security of signal transmission.

The structure of the proposed algorithm is relatively

Table 9: Comparison of different characteristic BER

Algorithm	Proposed algorithm	Wavelet low frequency logarithmic energy spectrum	Low frequency MFCC
Operating means	Average BER		
V.1	0.0092	0.0023	0.0205
V.2	0.0042	0.0010	0.0101
F.I.R	0.1957	0.3278	0.0975
R.8→16	0.0685	0.3021	0.1248
R.32→16	0.0078	0.0235	0.0151
E.A2	0.1684	0.0026	0.2195
G.N1	0.1320	0.1740	0.1999
M.48	0.1607	0.0774	0.2138
M.128	0.0126	0.1575	0.0292

Table 10: Comparison of operating efficiency of algorithms (average running time)

Algorithms	Hashing sequence length	Working frequency	Average time
Proposed algorithm	2888 bits	3.40GHz	0.3133s
[22]	266 bits	2.30GHz	0.0310s
[24]	256 bits	2.50GHz	0.3681s
[7]	360 bits	3.20GHz	0.5323s
[6]	-	3.30GHz	0.9008s

complex, the efficiency is low, and the robustness against filter interference is poor. The following research will optimize the algorithm structure, combined with support vector machine and other models to improve the efficiency. Further, the characteristics of other features to construct hash long sequence will be explore, and the speech tampering and approximate recovery will be study.

Acknowledgments

This work is supported by the National Natural Science Foundation of China(No.61862041), Youth Science and Technology Fund of Gansu Province of China(No.1606RJYA274).

References

[1] F. Bao and W. H. Abdulla, "A new time-frequency binary mask estimation method based on convex opti-mization of speech power," *Speech Communication*, vol. 97, pp. 51–56, 2018.

[2] S. I. Batool and H. M. Waseem, "A novel image encryption scheme based on Arnold scrambling and Lucas series," *Multimedia Tools and Applications*, vol. 78, no. 19, pp. 27611–27637, Oct. 2019.

[3] T. Bui, D. Cooper, J. Collomosse, *et al.*, "Tamper-proofing video with hierarchical attention autoencoder hashing on blockchain," *IEEE Transactions on Multimedia*, no. 99, pp. 1–1, 2020.

[4] S. Chen, R. Feng, Y. Zhang and C. Zhang, "Aerial image matching method based on HSI hash learning," *Pattern Recognition Letters*, vol. 117, pp. 131–139, 2019.

[5] N. Chen and W. G. Wan, "Robust speech hash function," *ETRI Journal*, vol. 32, no. 2, pp. 345–347, 2010.

[6] N. Chen and W. G. Wang, "Robust speech hash function," *ETRI Journal*, vol. 32, no. 2, pp. 345–347, 2010.

[7] N. Chen, H. D. Xiao, J. Zhu, J. J. Lin, Y. Wang and W. H. Yuan, "Robust audio hashing scheme based on cochleagram and cross recurrence analysis," *Electron Lett*, vol. 49, no. 1, pp. 7–8, 2013.

[8] P. P. Dahake, K. Shaw and P. Malathi, "Speaker dependent speech emotion recognition using MFCC and support vector machine," in *International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT'15)*, pp. 9–12, Aug. 2015.

[9] A. Ghasemzadeh and E. Esmaeili, "A novel method in audio message encryption based on a mixture of chaos function," *International Journal of Speech Technology*, vol. 20, no. 4, pp. 829–837, Dec. 2017.

[10] Y. B. Huang and Q. Y. Zhang, "Strong robustness hash algorithm of speech perception based on tensor decomposition model," *Journal of Software Engineering*, vol. 11, no. 1, pp. 22–31, 2017.

[11] Y. T. Jiang, C. X. Wu, K. F. Deng and Y. Wu, "An audio fingerprinting extraction algorithm based on

lifting wavelet packet and improved optimal-basis selection," *Multimedia Tools and Applications*, vol. 78, pp. 30011—30025, Nov. 2018.

[12] Y. Jiao, L. Ji and X. Niu, "Robust speech hashing for content authentication," *IEEE Signal Processing Letters*, vol. 16, no. 9, pp. 818–821, Sep. 2009.

[13] J. Li, T. Wu and H. Wang, "Perceptual hashing based on correlation coefficient of MFCC for speech authentication," *Journal of Beijing University of Posts and Telecommunications*, vol. 38, no. 2, pp. 89–93, 2015.

[14] J. F. Li, T. Wu and H. X. Wang, "Perceptual hashing based on NMF and MDCT coefficients," *Chinese Journal of Electronics (in Chinese)*, vol. 24, no. 3, pp. 579–583, July 2015.

[15] D. R. Nayak, R. Dash and B. Majhi, "Block-based discrete wavelet transform-singular value decomposition image watermarking scheme using human visual system characteristics," *IET Image Processing*, vol. 10, no. 1, pp. 34–52, Jan. 2016.

[16] S. Rameshnath and P. K. Bora, "Perceptual video hashing based on temporal wavelet transform and ran-dom projections with application to indexing and re-trieval of near-identical videos," *Multimedia Tools and Applications*, vol. 78, no. 13, pp. 18055–18075, 2019.

[17] D. Renza, J. Vargas, D. M. Ballesteros, "Robust speech hashing for digital audio forensics," *Applied Sciences*, vol. 10, no. 1, pp. 249, 2020.

[18] K. M. Singh, A. Neelima, T. Tuithung, *et al.*, "Robust perceptual image hashing using SIFT and SVD," *Current Science*, vol. 117, no. 8, pp. 1340, 2019.

[19] D. Slimani and F. Merazka, "Encryption of speech signal with multiple secret keys," in *International Conference on Natural Language and Speech Processing (ICNLSP'18)*, vol. 128, no. 2018, pp. 79–88, 2018.

[20] L. Sun, J. Xu, S. Liu, S. Zhang, Y. Li and C. Shen, "A robust image watermarking scheme using Arnold transform and BP neural network," *Neural Computing and Applications*, vol. 30, no. 8, pp. 2425–2440, Oct. 2018.

[21] J. Williams, J. Rownicka, "Speech replay detection with X-vector attack embeddings and spectral features," *Computation and Language*, 2019. (`arXiv: 1909.10324`)

[22] Q. Y. Zhang, W. J. Hu, S. B. Qiao and T. Zhang, "An efficient voice perception hash authentication algorithm based on LP-MMSE," *Journal of Huazhong University of Science and Technology (Natural Science Edition)*, vol. 44, no. 12, pp. 127–132, 2016.

[23] Q. Y. Zhang, S. B. Qiao, Y. B. Huang and T. Zhang, "A high-performance speech perceptual hashing authentication algorithm based on discrete wavelet transform and measurement matrix," *Multimedia Tools and Applications*, vol. 77, no. 16, pp. 21653–21669, Aug. 2018.

[24] Q. Y. Zhang, P. F. Xing, Y. B. Huang, R. H. Dong and Z. P. Yang, "An efficient speech perceptual hashing authentication algorithm based on wavelet packet decomposition," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 6, no. 2, pp. 311–322, Mar. 2015.

[25] Q. Y. Zhang, T. Zhang and S. B. Qiao, "Spectrogram-based efficient perceptual hashing scheme for speech identification," *International Journal of Network Security*, vol. 21, no. 2, pp. 259–268, Mar. 2019.

Biography

Yi-bo Huang received Ph.D candidate degree form Lanzhou university of technology in 2015, and now working as a Associate Professor in the college of physics and electronic engineering in northwest normal university, He main research interests include Multimedia information processing, information security, speech recognition.

He-xiang Hou received the BS degrees in communication engineering from Dezhou University, Shandong, China, in 2018. His research interests include audio signal processing and application, multimedia authentication techniques.

Man-hong Fan received M. Sc. degrees in Circuits and system from Northwest Normal University, Lanzhou, China, in 2012. His research interests include computer measurement and control.

Wei-zhao Zhang received M. Sc. degrees in Circuits and system from Northwest Normal University, Lanzhou, China, in 2011. His research interests include speech acoustics and speech signal processing.

Qiu-yu Zhang (Researcher/Ph.D supervisor), graduated from Gansu university of technology in 1986,and then worked at school of computer and communication in Lanzhou university of technology. He is vice dean of Gansu manufacturing information engineering research center, CCF senior member, a member of IEEE and ACM. His research interests include network and information security, information hiding and steganalysis, multimedia communication technology.

Guide for Authors
International Journal of Network Security

IJNS will be committed to the timely publication of very high-quality, peer-reviewed, original papers that advance the state-of-the art and applications of network security. Topics will include, but not be limited to, the following: Biometric Security, Communications and Networks Security, Cryptography, Database Security, Electronic Commerce Security, Multimedia Security, System Security, etc.

1. Submission Procedure

Authors are strongly encouraged to submit their papers electronically by using online manuscript submission at http://ijns.jalaxy.com.tw/.

2. General

Articles must be written in good English. Submission of an article implies that the work described has not been published previously, that it is not under consideration for publication elsewhere. It will not be published elsewhere in the same form, in English or in any other language, without the written consent of the Publisher.

2.1 Length Limitation:

All papers should be concisely written and be no longer than 30 double-spaced pages (12-point font, approximately 26 lines/page) including figures.

2.2 Title page

The title page should contain the article title, author(s) names and affiliations, address, an abstract not exceeding 100 words, and a list of three to five keywords.

2.3 Corresponding author

Clearly indicate who is willing to handle correspondence at all stages of refereeing and publication. Ensure that telephone and fax numbers (with country and area code) are provided in addition to the e-mail address and the complete postal address.

2.4 References

References should be listed alphabetically, in the same way as follows:
For a paper in a journal: M. S. Hwang, C. C. Chang, and K. F. Hwang, ``An ElGamal-like cryptosystem for enciphering large messages," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 445--446, 2002.

For a book: Dorothy E. R. Denning, *Cryptography and Data Security*. Massachusetts: Addison-Wesley, 1982.

For a paper in a proceeding: M. S. Hwang, C. C. Lee, and Y. L. Tang, ``Two simple batch verifying multiple digital signatures," in *The Third International Conference on Information and Communication Security (ICICS2001)*, pp. 13--16, Xian, China, 2001.

In text, references should be indicated by [number].

Subscription Information

Individual subscriptions to IJNS are available at the annual rate of US$ 200.00 or NT 7,000 (Taiwan). The rate is US$1000.00 or NT 30,000 (Taiwan) for institutional subscriptions. Price includes surface postage, packing and handling charges worldwide. Please make your payment payable to "Jalaxy Technique Co., LTD." For detailed information, please refer to http://ijns.jalaxy.com.tw or Email to ijns.publishing@gmail.com.