Design and Achievement of Security Mechanism of API Gateway Platform Based on Microservice Architecture

Yang Dawei, Gao Yang, He Wei, Li Kai
State Grid Xinjiang Information and Telecommunication Company, Urumqi, Xinjiang 83500

Abstract: To solve the current security problems of the microservice architecture, the security mechanism of the API gateway platform using the microservice platform is designed and implemented. To solve the corresponding security problems, a new API platform security mechanism is designed. In the data transmission, the hybrid encryption algorithm hybridized by encryption algorithm of Rivest Shamir Adleman (RSA) and encryption algorithm of Advanced Encryption Standard (AES) is used for data encryption. An API gateway between the client and the microservice is introduced as the only entrance for users to access the microservice, avoiding the complexity of the client logic caused by the direct communication between the client and the server. After the introduction of the API gateway, the API-level service operation authority can be controlled; the security of the front-end and back-end transmission data is improved without affecting the transmission performance. The algorithm designed takes longer time than the AES encryption algorithm, but shorter than the RSA encryption algorithm, which indicates that the time-consuming of the algorithm designed is acceptable, showing the feasibility of the algorithm. The content can provide an important theoretical basis for the security of API gateway platform of microservice architecture.

1. Introduction
With the advancement of science and technology in the information age, the traditional monolithic architecture application development system fails to renovate and repair applications in most enterprises. The design concept of microservice architecture is proposed [1, 2] under this background. The microservice architecture aims to propose a new architecture style which is a large complex software application is composed of one or more microservices [3]. Each microservice of a complete system only needs to focus on completing one task, which can greatly improve working efficiency of the whole system. Although the concept of microservice architecture can solve the problem of system efficiency of the platform, the increase of the number of microservices leads to the low utilization rate and poor stability of the system, which also causes more security problems [4]. Microservice architecture style is an emerging trend in software engineering, which allows the construction of highly scalable and flexible systems. However, we can only provide limited insight into the specific security issues of microservice systems with current technology.

Esposito et al. (2017) discussed the requirements and impacts of security, privacy, and discussed the existing methods. Finally, a security manager architecture was proposed for the management and exchange of cloud-based healthcare related data [5]. Since 2014, micro service architecture (MSA) has been widely deployed by Google, Netflix, Twitter and other large companies. Nguyen and Baker (2019) verified the relevant security by using microservice architecture on Spring framework, Spring security framework and Oauth2.0, and showed the effectiveness of Spring security framework and OAuth2.0 in protecting Spring based API (Application Program Interface) [6]. In this paper, we will solve and
improve the security problems after the introduction of microservice architecture. We will study the security mechanism of API gateway platform and solve some practical security problems.

2. Methods

2.1 Requirement Analysis on API Gateway
In the platform studied in this paper, the main tasks of the API gateway include the transmission of user requests, record of the platform traffic and related log information, etc., that is to say, the API gateway is equivalent to the bridge between users and platforms. Under the microservice architecture, the API gateway also needs to meet the functions of system status monitoring and user flow control to serve users during the request process. The functions of the API gateway are as follows:

1. The API gateway needs to obtain the user’s identity the first time, and then transfer it to the server side to user authentication.
2. The API gateway needs to respond to the user’s relevant requests and record the users’ request logs. In other words, it needs to record the abnormal requests of users, including multiple requests ended in failures. After recording the problems, the administrator will handle the abnormal request logs.
3. API gateway needs to record the relevant information of each application, such as call time, request result, etc.
4. The API gateway needs to control the flow. In other words, it needs to control the number of requests and calls per second.
5. According to the user's request information, the user’s request is forwarded to the corresponding server.

When facing the administrator, the API gateway needs to complete the following tasks:

1. Manage the service authority, configure the situation that each service calls other service interfaces to avoid duplication between two services.
2. Through the configuration of various services, the system is monitored to check whether there are abnormal phenomena in the current system. In case there are abnormal phenomena, the relevant staff will be reminded in time.
3. According to the administrator’s settings, the system traffic is limited. Through the configuration required by the administrator, the traffic quota of other services or applications is limited.
4. In order to ensure the security of the whole system, it is necessary to configure the internal and external URL (uniform resource locator).

2.2 Platform Security Requirements Analysis
In the architecture of the platform, the data transmission process of the front and rear end is easy to be intercepted. If the data packet is intercepted, it will lead to data leakage, and there is a big security risk. Therefore, in terms of security, we need to improve the overall data security. Data encryption is one of the commonly used methods. One of the most common used methods of digital signature is to verify whether the data is modified.

There is a certain correlation between the data transmission security of the whole system and the security encryption algorithm used, but the degree of confidentiality of the key is also one of the important factors determining the security level of the security system. Therefore, a certain degree of confidentiality operation is required for the key to achieve a higher security level. The key encryption can be realized by encrypting the generation of the key and can adjust the use and storage of the key.

2.3 The Design of Overall Framework of Security Mechanism
Direct transmission of JSON plaintext is the main way of data transmission at the front and back of the platform. This method is easy to be cracked by attackers, resulting in data loss or modification, which leads to great security risks in the process of data transmission. Therefore, this paper improves the security mechanism, and the overall framework after improvement is shown in Fig. 1:
As can be seen from the above figure, the security mechanism designed in this paper has been improved from two aspects of data and authentication system. Firstly, in terms of data, AES (Advanced Encryption Standard) algorithm and RSA (Rivest Shamir Adleman) are used for mixed encryption and then data transmission, which improves the security of data transmission process. Then the API gateway platform is used as a communication tool between users and microservice module to solve the backend security problem.

2.4 Design of API Gateway
Microservice architecture API gateway is mainly composed of request proxy subsystem and management subsystem. In the request proxy subsystem, the API gateway platform needs to distinguish the user’s identity information from their access requests and forward the identity information to the server for user’s identity authentication. If users pass the authentication, their accesses are allowed. If users are not authenticated, their accesses are denied.

In terms of management subsystem, since each microservice has its own management authority, the main work of API in the management subsystem is to configure related resources, including flow control, URL configuration, etc.

2.5 Implementation of Data Mixed Encryption
Advanced Encryption Standard (AES) is the most common symmetric encryption algorithm (this encryption algorithm is used for We-chat small program encryption transmission) [7,8]. Symmetric encryption algorithm means that the same key is used for encryption and decryption. The specific encryption process is shown in Fig. 2

The plaintext in the figure above represents the data that has not been encrypted, and the key represents the password used to encrypt the plaintext. In the symmetric encryption algorithm designed in this paper, the key required for encryption is the same as the key used for decryption. The key is generated after the discussion between the data sender and the data receiver, but the key cannot be transmitted directly on the network, otherwise, the key is likely to be leaked. The AES encryption function is as follows. Assuming that the AES encryption function is E, then P represents plaintext, K
represents key, and C represents ciphertext. The encrypted data is immediately encrypted:

\[ C = E(K, P) \] (1)

It can be seen from the above formula that if the plaintext and key are input as the parameters of the encryption function, the encrypted function will output the encrypted ciphertext.

RSA mainly uses the mathematical problem of Large Integer Decomposition to design, and skillfully uses the concept of number theory. Nowadays, only short RSA key can be solved in a powerful way [9]. The steps of RSA encryption are as follows:

1. Two unequal prime numbers a and b are randomly selected (in practical application, the larger the prime number is, the higher the encryption level of the whole system is).
2. The product N of two prime numbers can be figured out.
3. The Euler function is calculated, and the binary length of N is taken as the length of the key.
4. An encryption key e can be randomly selected. The formula here can be shown as \( 1 < e < \varphi(N), \gcd(e, \varphi(N)) = 1 \);
5. The decryption key d is obtained according to \( ed = 1 \mod \varphi(N), 0 \leq d \leq N \);
6. Publish the encryption key (e, N).
7. Release the decryption key (d, N).

The specific steps of the hybrid encryption algorithm designed in this paper are as follows:

1. A 128-length encryption key K is randomly generated by the sender. AES algorithm is used to encrypt the plaintext, and the encrypted ciphertext C is obtained.
2. The receiver creates the RSA key while encrypting the data of the sender. The key is divided into private key and public key. The self-preserve private key is sent to the sender through the network as a public key.
3. The sender of the data encrypts the key K by using the public key sent by the receiver to get the encrypted key K.
4. The sender sends the encrypted ciphertext and the encrypted key to the receiver.
5. The receiver decrypts K' with its self-preserve key and then obtains the key B.
6. The receiver uses the key to decrypt the ciphertext and get the plaintext.

3. Results and Discussion

3.1 API Gateway Performance Verification

This paper only verifies the application level flow control and request forwarding function of API gateway. In the aspect of application level flow control, the initial state of the research is that the service has been associated with application information, but the API traffic policy has not been set. First, enter the API management page in the microservice platform, enter the query rate per second setting interface, set the query rate per second to 10, start 20 threads, and call the API at the same time.

The verification results of application level flow control of API gateway are as follows. When the API is faced with 20 call requests, the flow control is carried out on the 20 call requests. The results show that 10 call requests display frequent request information, and 10 call requests can return the corresponding results normally. From the above research results, we can see that it is effective to call API gateway platform between users and microservices.

The test results of request forwarding capability of platform API gateway are shown in Fig. 3:
From the above test results, we can see that there is a high consistency between the expected performance and the actual performance of the API gateway, especially in the first four steps, the expected performance is completely consistent with the actual performance, which indicates that the API gateway has good request forwarding ability.

### 3.2 Data Security Analysis Results

In the method part above, we have explained the data mixing encryption algorithm in detail, studied the feasibility of the algorithm designed in this paper. Using 128 byte-data as transmission data, AES key length is 128 bits, RSA key length is 1024 bits, respectively. In the process of AES encryption, RSA encryption and mixed encryption, the program is executed 1000 times and 10000 times respectively. The time required can be figured out. The result is shown in Fig. 4:
Because the hybrid encryption algorithm designed in this paper has better encryption effect, if it has good performance in program execution time, it means that the algorithm designed in this paper has certain feasibility. From the results shown in the figure above, we can see that the algorithm designed in this paper takes longer time than AES encryption algorithm, but shorter than RSA encryption algorithm. The time consumption of the proposed algorithm is acceptable, which shows the feasibility of the proposed algorithm.

4. Conclusion
Since there are some data security problems in the current microservice architecture, this paper puts forward the corresponding solutions to the problems after putting forward the relevant security risks. Firstly, in the data aspect, AES algorithm and RSA algorithm are used for mixed encryption, and then the data transmission is carried out to improve the security in the data transmission process. Secondly, the API gateway is used as a communication tool between users and microservice modules to solve the back-end security problem. After the research, after the overall structure improvement, it can realize the service API level operation authority control; without affecting the transmission performance, it improves the security of the front and back-end transmission data. Although this study has made some achievements, there are still some deficiencies. With the development of various technologies, there will be more and more types of network attacks in the future, so the security of data will be more and more difficult to be guaranteed. The hybrid encryption algorithm designed in this paper is also difficult to resist all attacks, so the relevant content needs further researches to improve the security of the platform. Secondly, the key management scheme designed in this paper is relatively simple. To ensure the security of the platform, other more secure key management schemes can be used.

Author
Yang Dawei ,1986.10, Shandong, Master, Engineer, Technical Supervisor

References
[1] Pan Xiaoyang, Huang Xiaofang. Design [J.] of security control mechanisms for micro-service frameworks Journal of Southwest University of Science and Technology.
[2] Han Daoqi. Design of Micro-Service Architecture and Security System [J.] and Electronic Technology and Software Engineering ,2019,000(002):199-201.
[3] Platinum Snow, Jin Wang, Ma Tao, et al. the practice of Network Security level Protection based on Micro-Service Architecture [J.]; and Information Security and Technology, 2019, 010(003): 23-27.
[4] Gong Minghao, Liang Jinchun, Yao Yingying, et al. Container-based Fusion Media Micro-Service Architecture Security Threats and Protection Methods [J]. Radio and Television Information ,2019(5):64-67.
[5] Esposito C, Castiglione A, Tudorica C A, et al.Security and privacy for cloud-based data management in the health network service chain: a microservice approach[J]. IEEE Communications Magazine, 2017,55(9): 102-108.
[6] Nguyen Q, Baker O F. Applying Spring Security Framework and OAuth2To Protect Microservice Architecture API[J]. JSW,2019,14(6): 257-264.
[7] Almeida W H C, de Aguiar Monteiro L, Hazin R R, et al.Survey on microservice architecture-security, privacy and standardization on cloud computing environment[J]. ICSEA 2017, 2017: 210.
[8] Zhang Yajuan, Han Yinxue, Liu Xiaoyang. APP[J]. of account password management based on AES algorithm Computer Technology and Development ,2019,029(008): 125-129.
[9] Li Ying, Zhao Rui, Cao Yu, et al. Research on RSA Encryption Algorithm [J.]; and Smart Computers and Applications ,2020,010(003):166-168.