*, [1] Wei Zheng
[1] Tao Liu
[1] Lihong Hao
[2] Jiwei Tang

Design of Privacy Protection System for Target Search Based on Selection Algorithm and Sustainable Iteration Index

Journal of Electrical Systems

Abstract: - The data self-destruct method based on privacy cloud proposed in this paper can realize the automatic destruction of cloud data after expiration, and based on this method, a self-destruct image protection system is implemented. This project intends to combine the adaptive scale invariant Feature conversion (SIFT) extraction of ciphertext images with binary SIFT to ensure the confidentiality of the images uploaded to the server in the ciphertext space. The security protection of data is realized using Paillier homomorphic encryption technology. The feature of SIFT is extracted from the encrypted region, and it is expressed in binary form to reduce the amount of computation and memory overhead. The experimental results show that this method can effectively prevent the automatic destruction of images after expiration, so as to ensure that the user's personal information is not leaked.

Keywords: Data Self-Destruction; Cloud Data; Privacy; Electronic Image.

I. INTRODUCTION

Cloud technology provides individuals and enterprises with low-cost, high-quality network services, so as to better meet the growing needs. However, because of the nature of cloud computing, user data is permanently stored, copied, used and archived to the cloud. User misoperation, illegal operation, hackers, cloud computing service providers mining, stealing user data, etc., may cause the risk of user data privacy disclosure. In the past decade, electronic imaging has become the most typical application scenario in the cloud computing environment. In the 1990s, video server vendors could only reserve a few hundred megabytes of storage per account, and they had to pay for more [1]. But now, people can get gigabytes of pictures for free. Human beings transmit and receive all kinds of information through electronic images, such as address books, personal notes, schedules, business contracts, electronic bills, social network status announcements, etc. Electronic images are the most commonly used means of communication in today's enterprises, units, and government departments. As the digital image storage capacity becomes larger and larger, users often forget to manually erase the image data [2]. The risk of this is that if the user's picture account is leaked, the personal information of the individual will be exposed. In addition, even if the user manually removes the image from the mailbox, the user's image data will be preserved in the cloud service provider or the cloud service provider's server. In recent years, problems such as user privacy disclosure and trade secret exposure caused by photo disclosure have caused great harm to user privacy and enterprise economic rights and interests [3]. Therefore, these data stored in the cloud must have a life cycle, only within this life cycle, can be allowed to use. Personal data in the cloud will self-destruct at the end of its useful life. Data self-destruction is the ultimate measure to ensure data security and privacy.

At present, a lot of research has been done on the data self-destruction technology on the Internet. Researchers have studied a new data destruction mechanism based on Hash algorithm. Symmetric key is used to encrypt the user's private information, and it is divided into n groups through secret sharing mechanism, and then allocated to a large distributed hash table network [4]. Therefore, without relying on trusted servers and human intervention, the DHT algorithm periodically removes the key portion, making the ciphertext unreadable. Some researchers have perfected the Vanish algorithm, encrypting DHT systems as well as making them more resistant to conventional encryption detection and brute force attacks. This project intends to study a deterministic deduplication algorithm suitable for cloud computing environment, that is, key derived tree is used to structure and manage it, and then distributed to DHT in a similar way to DHT protocol, so as to achieve the purpose of reducing data certainty [5]. However, some scholars have found that Sybil attacks may occur in VuzeDHT networks, that is, attackers can obtain a sufficient number of keys before the data expires, so as to reconstruct the decryption key. Researchers have proposed SafeVanish (SafeVanish) scheme, which protects against Sybil jump attacks based on Sybil detection by increasing the length of key branches. This project intends to study an ISDS system that integrates identity-based cryptography with DHT network, encrypts symmetric keys using IBE

[1] * Dadu River-hydropower Development Co. Ltd, sichuan, chengdu, 610041, China
2 CHN Energy Dadu River Big Data Services Co., Ltd., Sichuan, chengdu, 610041, China
*Corresponding author: Wei Zheng

method, and assigns their ciphertext to DHT network to resist Sybil attacks. Web Service is a new distributed computing model based on the Internet, which is very suitable to exist in an autonomous and open form on the Internet. However, the open, dynamic and autonomous characteristics of Web services determine their application and disclosure in the network [6]. With the increase of user privacy information infringement cases, the protection of privacy information has attracted more and more attention from users, especially in the case of Web service composition, service providers combine Web services to form larger granularity web services, so as to achieve complex business logic. User privacy information is exposed to member services through Web service composition. There is no agreement on the use of private information between users and member services, so it is difficult to ensure that the private information can be exposed and used according to the wishes of users during the implementation of the combination.

The existing data destruction methods are based on the loss of private keys, and the ciphertext format cannot be cracked, that is, the secret information in a system depends on the security of ciphertext to a large extent [7]. This paper adopts the method of ciphertext fragmentation, that is, the system's protection of private cloud data depends on the integrity of private data. The self-destruct technology based on cloud computing environment studied in this project aims to protect expired or archived cloud data. By setting timeout, all privacy data in the cloud is destroyed. Once the data expires, it is automatically destroyed, and even if the hacker gets all the data and passwords stored in the cloud, it is impossible to recover all the private information.

II. SECURITY MODEL BASED ON PRIVACY CLOUD

A. Cloud Data Security

Affected by the cloud environment, the cloud in the cloud is copied, cached and archived, so it is difficult for both users and enterprises to clean it all up. For the cloud computing environment in the cloud computing environment, the cloud computing security Alliance puts forward the data life cycle in the cloud computing environment in the cloud computing environment, and summarizes it into six stages: generation stage, storage stage, use stage, sharing stage, archiving stage and destruction stage. Uncontrolled information is the root cause of data security problems in cloud computing environment [8]. The user has no control over the user's data, but the user must have access to the cloud user's data. According to the data life cycle principle in cloud computing, the privacy leakage problem in cloud computing environment is fundamentally solved.

B. Privacy Cloud Data

Private data in a cloud computing environment refers to cloud data that contains sensitive and confidential information. In the cloud computing environment, users need to carry out a comprehensive "destruction" to prevent it from being attacked, cracked, exploited and analyzed, resulting in huge losses in the economy and reputation of users in the cloud computing environment. A private cloud is a secure cloud service for storing sensitive data in the cloud [9]. The direction of personal cloud development should be: enterprise cloud, public cloud and hybrid cloud. Privacy protection in cloud computing environment is facing serious challenges. Privacy Data in the cloud automatically fails in the secure state and cannot be retrieved, which ensures the privacy security of individuals and enterprises in the cloud computing environment. Data Agent refers to the data in the cloud computing environment managed by the cloud computing provider. However, in the actual application, because it is separated from the actual data, it can not reflect the real information. The security model based on private cloud computing is shown in Figure 1 (image cited in Journal of Network and Computer Applications, Volume 160, 15 June 2020, 102642).

Vanish is a new distributed Hash algorithm based on the threshold secret sharing technology, which combines P2P technology and distributed Hash table technology to build a cloud computing environment. This scheme adopts an encryption algorithm based on P2P network. If you can't get the full key, you can't crack the code. Given that all existing encryption algorithms can segment encrypted files, why not segment encrypted files? To a certain extent, the best way to stop information leakage is to break the integrity of the entire system [10]. In this paper, an improved algorithm based on time limit and unlimited secret sharing is proposed. The process is shown in Figure 2 (image referenced in A secure self-destructing scheme for electronic data).

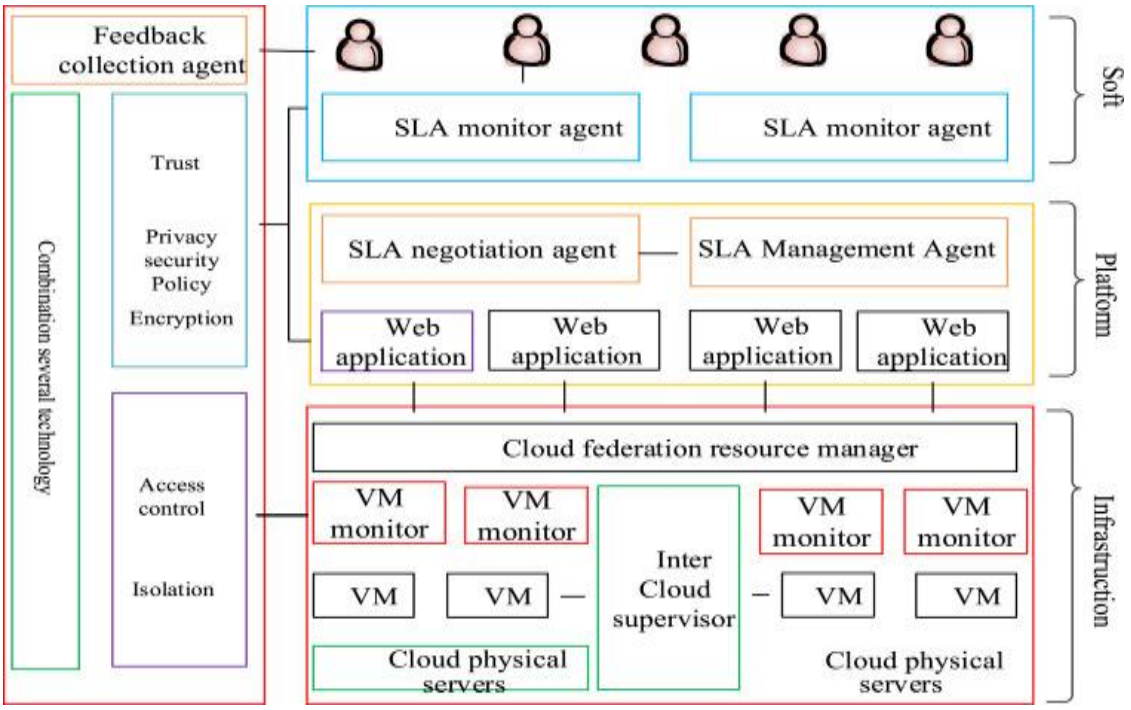

Fig.1 Security model based on privacy cloud

III. TIME-BASED AUTO-DESTRUCTION PROCEDURES

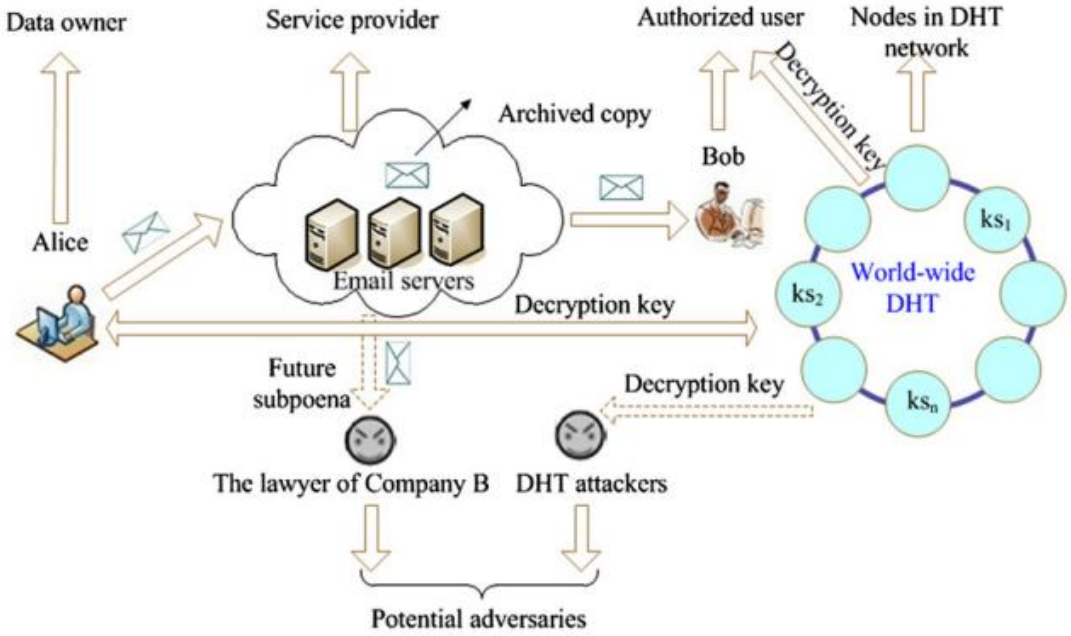

Fig.2 Data self-destruct method based on time period

Step 1 using symmetric cryptography to encrypt the content. Encrypt is a method of encrypting its contents using a random keyword and then decrypting it as a password [11]. In order to improve the security of the system, AES encryption method is adopted in this paper.

Step 2 fragmenting the ciphertext. The function Split divides the ciphertext cipher generated in the first step into data fragments according to the specified fragment length. $p_1, p_2, p_3, \cdots,$ function Disguise the key randomKey as data fragment p_0 by adding random bytes.

Step 3 Switch the order of p_0 and p_i. The system generates i between (0, n) according to the genIndex() function set, and exchanges the contents of p_0 and p_i.

The fourth step encapsulates the ciphertext as SDO. SDO encapsulates the ciphertext data shard, shard identifier, and data expiration timestamp ET. The function Uniquify generates p_i UUID u_i for each shard A to ensure that the ciphertext shard has a unique identifier; Function Encapsulate Encapsulates each UUID u_i , ciphertext fragment p_i , and preset expiration timestamp ETi into an SDOi, which is persisted to storage devices.

Step 5 Generate the data broker. In this article, the data broker is the SDD link. The Link function joins the UUIDA of each SDO in order to form a string, adding "SDD: // " to the beginning of the string, indicating that this is an SDD connection. Through the SDD connection, the user can obtain encrypted data fragments from the server. In this data destruction approach, Sdos in the database are also required to be monitored in real time and cleared when they expire. $u_0, u_1, u_2, \mathrm{L}$, new method based on Delete-On-Delay is proposed, which can effectively solve the storage overhead of multiple Sdos with different expiration dates [12]. The purpose is that the ET of SDOi (i>1) is randomly delayed 1-60 min in SDOi (i≥1).

IV. IMAGE SEARCH ALGORITHM WITH CONFIDENTIALITY

Combining the above three steps organically, a two-value secret SIFT algorithm is presented. The same effect can be achieved by plaintext operation on ciphertext, so homomorphic cryptography has received more and more attention. In the Paillier cipher system, the addition operation of ciphertext is equivalent to the multiplication operation of plaintext [13]. In this paper, we choose the Paillier homomorphic cipher algorithm. At this point, the SIFT characteristics are compressed using the same quantization method as B-SIFT, thus ensuring the same matching effect as B-PPSIFT. Figure 3 shows the operation steps of the B-PPSIFT algorithm (the image is quoted in 81.6 GOPS Object Recognition Processor Based on a Memory-Centric NoC).

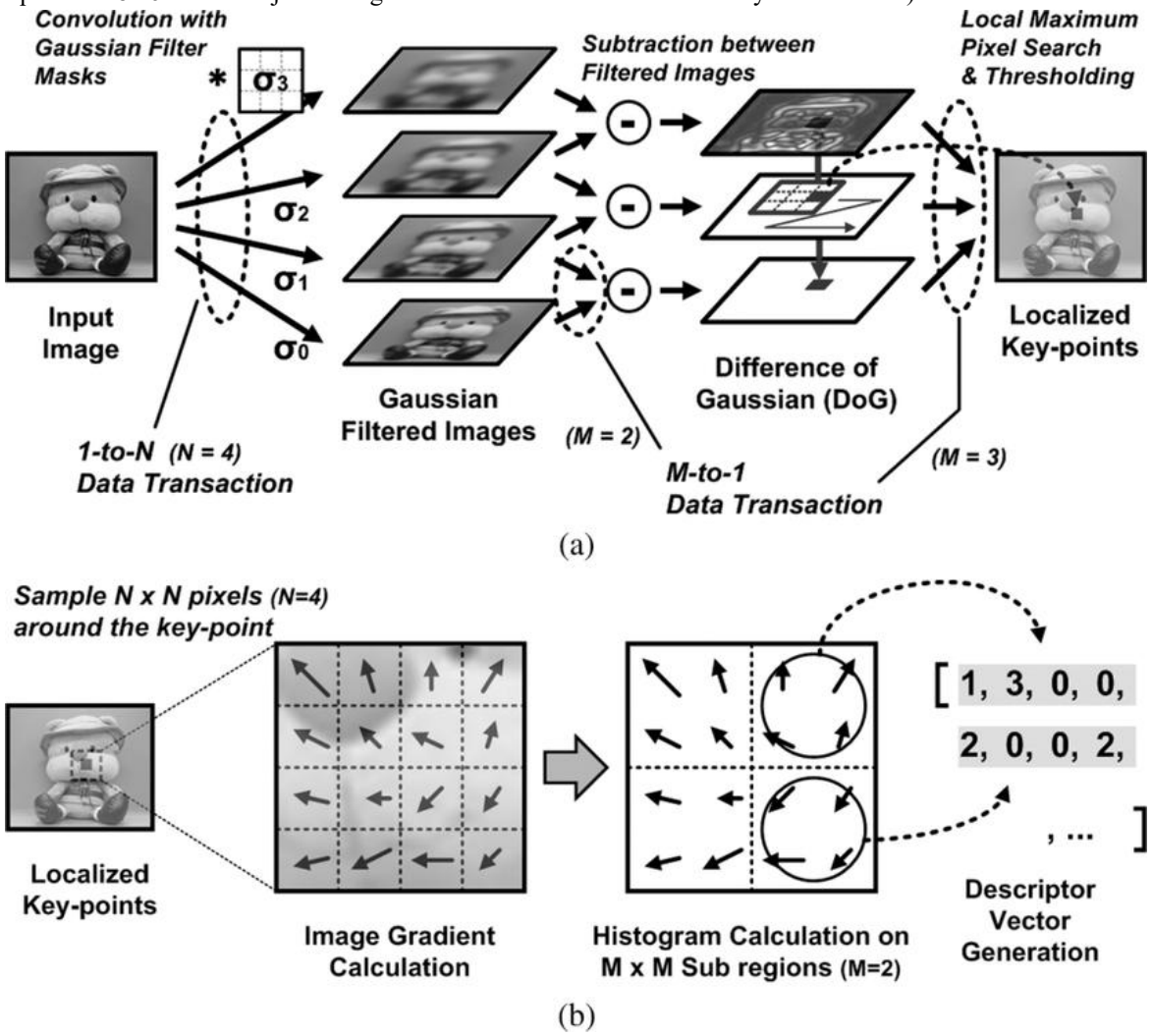

Fig.3 Operation steps of B-PPSIFT algorithm

The flow of this calculation is shown in Figure 4 (image cited in Machines 2022, 10(6), 456). For a given image W, the SIFT feature is directly extracted, denoted as $G(W)$, and quantized to obtain B-SIFT, denoted as $G'(W)$. The image of image W after homomorphic encryption is denoted as W_e, SIFT feature extraction of the encrypted image is denoted as $G(W_e)$, and B-PPSIFT is quantized and generated, denoted as $G'(W_e)$.

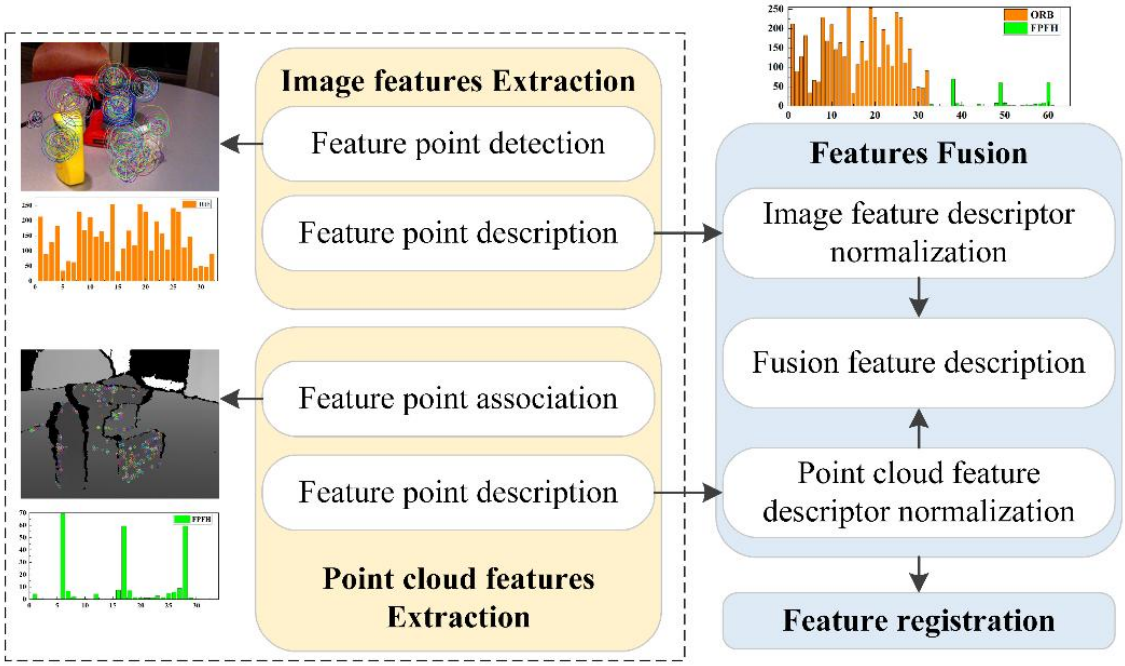

Fig.4 Schematic diagram of calculation process

In B-SIFT, Hamming distance is used to measure image similarity, which maintains almost the same accuracy compared with Euclidean distance used in the original SIFT, as shown in equation (1)

$$sim(G(W^u),G(W^v))=\sqrt{\sum_{j=1}^{m}(g_{i,j}^u-g_{i,j}^v)^2}\Leftrightarrow$$
$$simQ(G'(W^u),G'(W^v))=\sum_{j=1}^{m}(g_{i,j}^u\oplus g_{i,j}^v) \tag{1}$$

Where $simP$ is the Euclidean distance used to calculate the similarity between two feature points; $simQ$ is the Hamming distance to calculate the similarity between two characteristic bit strings; W^u and W^v represent two feature points in image u and image v, respectively. Where $G(W)$ and $G'(W)$ referred to at both ends have equivalent consistency when measured by Euclidean distance and Hamming distance, respectively [14]. Then by replacing W with W_e, equation (2) can be derived from equation (1).

$$simP(G(W_e^u),G(W_e^v))\Leftrightarrow simQ(G'(W_e^u),G'(W_e^v)) \tag{2}$$

First, we need to calculate the difference gauss. It is proved that there is S_e $S_e(u,v,\varphi)=W(S(u,v,\varphi),D_\varphi)$ relation between the extraction of differential Gauss A after homomorphic encryption and the calculation of differential Gauss S directly. D_φ selects the value consistently for the combination, depending on $G()$, not the image size. φ represents the difference $\varphi=\varphi_i-\varphi_j$ between adjacent Gaussian fuzzy images [15]. The user upload the encrypted image W_e to the server, and the differential gauss calculated in the ciphertext field remains equal to the differential Gauss encrypted directly with D_φ on scale φ. When locating extreme points, for λ_1 and λ_2, if $\lambda_1>\lambda_2$, then $W(S(u_1,v_1,\varphi_1),D_{\varphi_1})<W(S(u_2,v_2,\varphi_2),D_{\varphi_2})$.

The similarity measure is R_1 distance, and the formula for calculating R_1 distance in the plaintext domain is $sim_{R_1}(G(W^u), G(W^v)) = \sum_{j=1}^{m} | g_{i,j}^u - g_{i,j}^v |$. It is also proved that the similarity of features can be measured by directly calculating the absolute value of the difference of feature vectors in the ciphertext domain [16]. That is, the Euclidean distance calculated by the original SIFT is equivalent to the distance calculated by R_1 in the ciphertext domain, as shown in equation (3)

$$simP(G(W^u), G(W^v)) \Leftrightarrow sim_{R_1}(G(W_e^u), G(W_e^v)) \tag{3}$$

Euclidean distance is chosen here for the distance measurement method of SIFT feature points in the original image, and Hamming distance is adopted for $G'(W_e)$ based on the transfer principle of equivalence. It is equivalent to calculate Hamming distance in the original image using Euclidean distance and B-PPSIFT

$$simP(G(W^u), G(W^v)) \Leftrightarrow sim_Q(G'(W_e^u), G'(W_e^v)) \tag{4}$$

When the user puts the image through B-PPSIFT, the server will not know which image the user is searching for. On the server, the corresponding B-PPSIFT is saved, at this time, in the process of image retrieval, it is to use a two-value password to match. In the premise of ensuring information security, it can also protect the privacy of users.

V. PROTOTYPE SYSTEM IMPLEMENTATION

Digital image is the most typical application example in cloud computing. Digital images containing personal and corporate confidential information can be regarded as cloud privacy data. Among them, a complete electronic image architecture is established based on SMTP protocol, envelope and content [17]. When sending an image, it passes through multiple image servers and stores a copy of the image at the same time. Although the sender and receiver have removed the image copy from the client, there are still other image copies on the image provider's server. However, because the communication mode of SMTP is not allowed to change, so in order to achieve the purpose of self-destruction, we must establish a content-based self-destruction mechanism on it. In terms of text, the self-destruction of electronic images is essentially the destruction of electronic images that have been aged for a period of time into an unreadable and meaningless format [18]. The self-destruct electronic image system proposed in this paper uses the method mentioned above to process the body part of the image to obtain SDD links, and the transmitter can store the data in the image body, instead of writing the plain text data directly to the image body, and store the plain text of the image in a separate server. Through the SDD connection, the receiver can obtain the corresponding ciphertext and key. If all blocks are not expired, they can be reorganized and decrypted to obtain the plaintext content. When a data slice expires, the other slices and SDD links remain, but the integrity of the data has been damaged, so the image information cannot be captured. The model is divided into two parts: one is the server side and the other is the client side. This client is an extension based on Google Chrome that has two basic features:

(1) Provide the sender with an interface for transmitting the image content to the private cloud via a secure channel and transmitting the image content back to the data agent.

This paper presents an image auto-destruction technique based on SSL. On the web page, the user can select the content of the image with the left mouse button and select the "ToSDDLink" menu item by pressing the right mouse button. Users need to set the expiration date of the image before they can transfer the image content to the private cloud. The SDD link from the private cloud replaces the image part of the web page.

(2) Provide an interface for the receiver to transmit it to the private cloud through a secure channel and transmit it back to the actual data. Once the user has selected a complete SDD link, right-click and select the "ToPlain" menu item. The hidden content appears as a pop-up (Figure 5). If the confidential content expires, the browser plug-in signals to the user that the content is expired (Figure 6).

The server-side architecture is shown in Figure 7 (image cited in Privacy Mediators: Helping IoT Cross the Chasm). The problem of interconversion between plaintext and SDD connection is solved emphatically [19]. The rendering layer is Restful We bService, which is a client-oriented system API. Through data self-destruct technology, image content, ciphertext, data self-destruct, SDD link and other links are transformed. The persistence layer persists the various features of SDO, either NoSQL or a regular relational database.

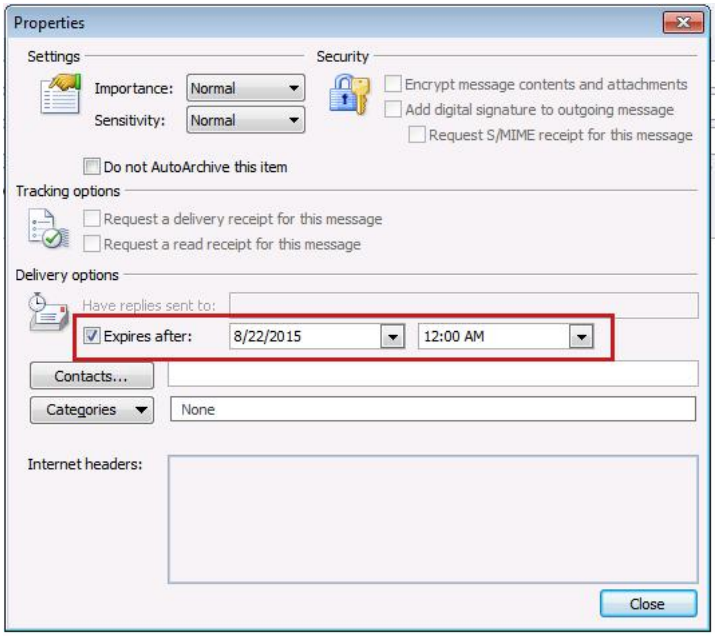

Fig.5 Before the content expires

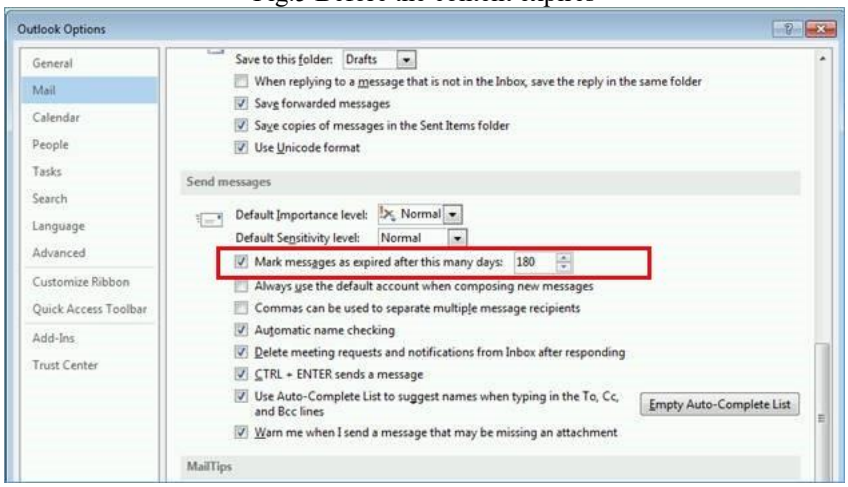

Fig.6 Image content self-destruct

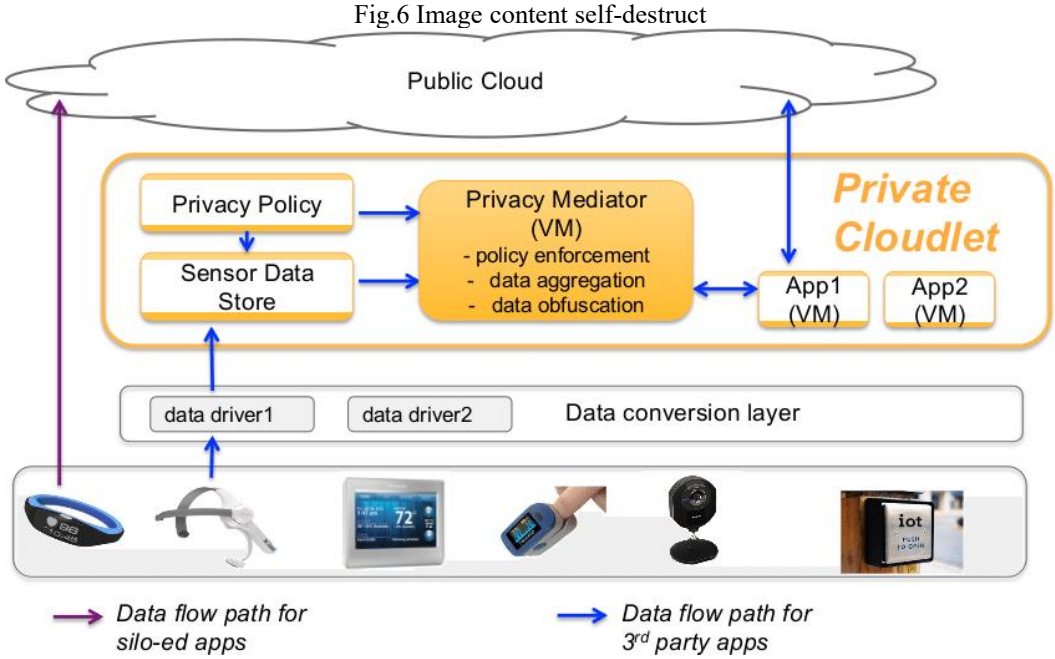

Fig.7 System software design structure diagram

VI. SECURITY AND PERFORMANCE ANALYSIS

In Figures 8 and 9, when a user sends a self-destructing image fragment through the system, the content of the image is essentially a data-mediated SDD link, and through this URI, the browser gets the actual data and provides it to the user. When the image expires, even if the image content stored in the client or image server is leaked, the true content of the original image cannot be reproduced [20]. This method uses a separate memory to save the actual picture. In contrast to traditional image-based data encryption technologies such as Vanish, the solution allows for greater control over the security and control of encrypted files.

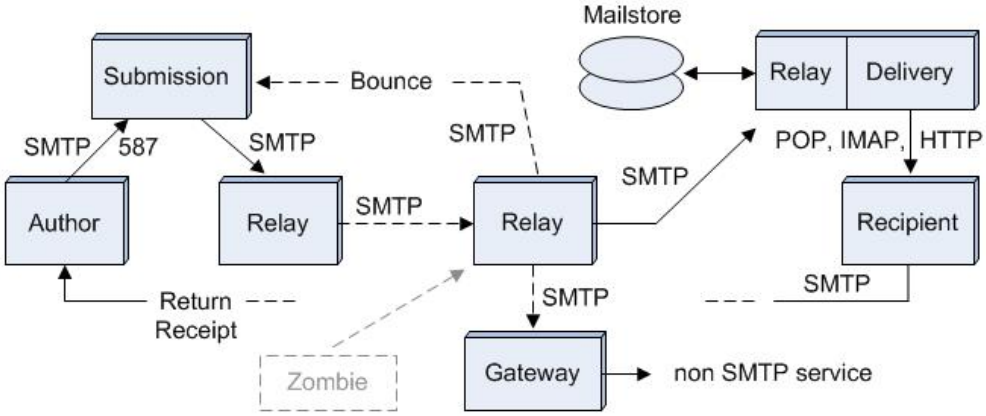

Fig.8 Ordinary electronic image sending and receiving process

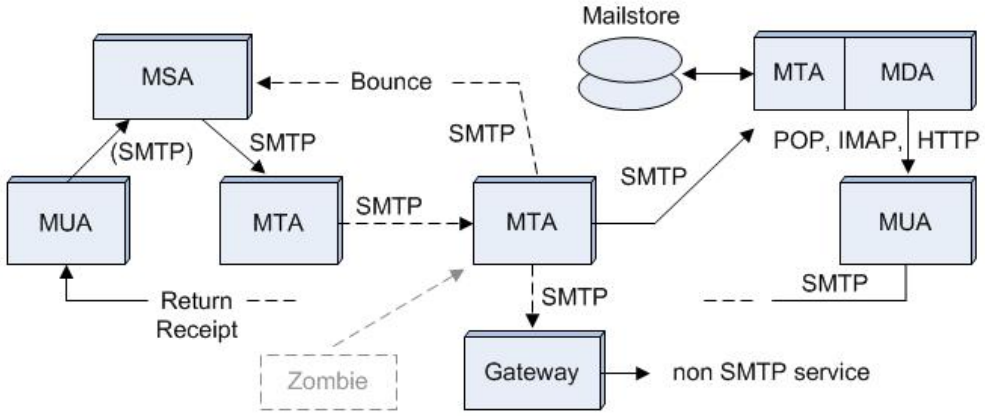

Fig.9 Self-destruct image sending and receiving process

However, the existing SDO models all adopt the Delete-On-Delay mechanism, so there is indistinguishability between different Sdos on different time scales, that is, it is impossible to determine which SDO is the same according to different time markers. The UUID is the unique identifier of the SDO and ensures that data in the database does not collide [21]. The time cost of using brute force decryption is N2, where N is all the records in the database. This paper proposes a network topology based on SSH to realize the communication between users and servers.

From a performance point of view, dividing the data into several paragraphs, whether written or read, will cause a decrease in performance compared to the ordinary situation. In this paper, the data destruction algorithm based on network is used in two cases of less than 1 MB and more than 1 MB, and the effects of reading and writing are compared. The experimental conditions are: CPU 2 dual-core processor 2.66 GHZ, RAM4GB. Using MySQL5.5 database and JDK1.7 Java environment. Scheme design parameters: AES encryption key is 256 bits, slice length is 64 KB. As shown in Figures 10-11, normal is a situation where data of a specific size is written directly to or read from the database in the form of LOBs. SDD is a situation where a database is written or read after the data has been processed using the network-based data self-destruction described here [22]. As you can see, the time cost of reading and writing is proportional to the amount of data required. Under 1 MB, the performance loss of SDD is large, only a few ms to tens of ms. When the data size exceeds 1 MB, because the number of slices grows exponentially, the more database connections and queries are required, and the longer it takes (Figure 12, Figure 13).

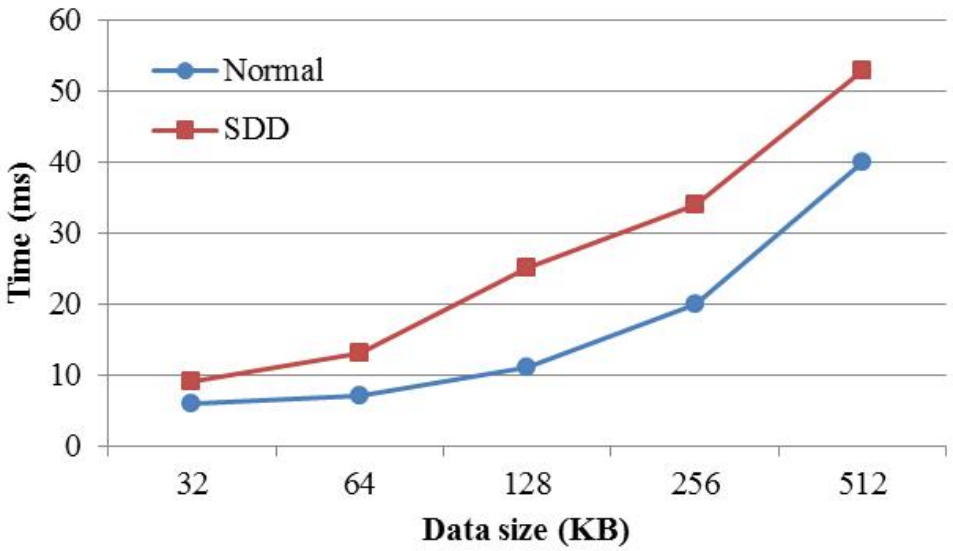

Fig.10 Write data: less than 1MB

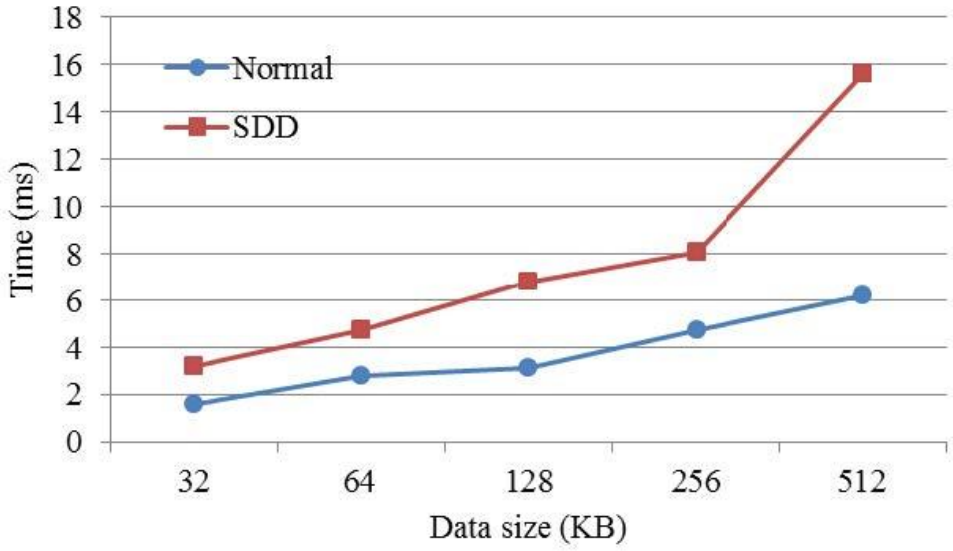

Fig.11 Read data: less than 1MB

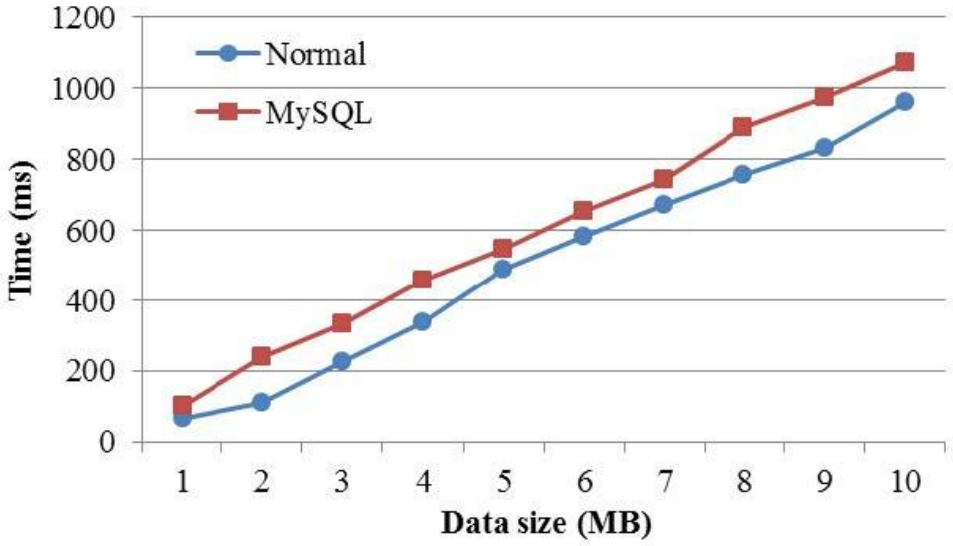

Fig.12 Write data: greater than 1MB

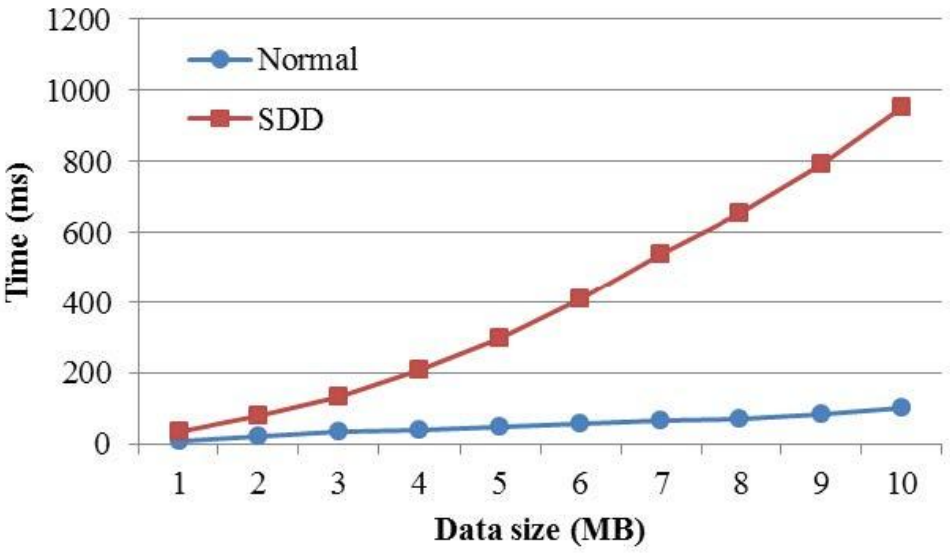

Fig.13 Read data: greater than 1MB

Since the goal of this study is to protect electronic images, and its main content is a web page with more than 100 KB, the protection scheme proposed in this study has no obvious negative impact on the user's experience.

VII. CONCLUSION

The electronic image security guarantee system in the cloud environment proposed in this topic can automatically destroy the image after failure to ensure the confidentiality and privacy of the image. A scheme for enterprise or trust public cloud is proposed. In the future, an authentication feature will be added that only authorized users can access the cloud through this feature, thus enhancing the security of electronic images. Privacy cloud is not limited to digital images, but to build a public cloud service platform for users, so that users can freely create and manage a variety of information containing private information, and share it to a variety of Web applications, and only within a specific time limit, in order to effectively control and protect the privacy of users. Information security is the organic unity of technology and cognition, and as users attach importance to information technology, users will gradually increase their attention to information security.

REFERENCES

[1] Li, J., Zhang, Y., Ning, J., Huang, X., Poh, G. S., & Wang, D. (2020). Attribute based encryption with privacy protection and accountability for CloudIoT. IEEE Transactions on Cloud Computing, 10(2), 762-773.

[2] Calzada, I. (2022). Citizens' data privacy in China: The state of the art of the Personal Information Protection Law (PIPL). Smart Cities, 5(3), 1129-1150.

[3] Sun, Z., Wang, Y., Cai, Z., Liu, T., Tong, X., & Jiang, N. (2021). A two-stage privacy protection mechanism based on blockchain in mobile crowdsourcing. International Journal of Intelligent Systems, 36(5), 2058-2080.

[4] Liu, B., Ding, M., Shaham, S., Rahayu, W., Farokhi, F., & Lin, Z. (2021). When machine learning meets privacy: A survey and outlook. ACM Computing Surveys (CSUR), 54(2), 1-36.

[5] Wang, Y., Tian, Y., Yin, X., & Hei, X. (2020). A trusted recommendation scheme for privacy protection based on federated learning. CCF Transactions on Networking, 3(3), 218-228.

[6] Wang, Y., Su, Z., Zhang, N., Xing, R., Liu, D., Luan, T. H., & Shen, X. (2022). A survey on metaverse: Fundamentals, security, and privacy. IEEE Communications Surveys & Tutorials, 25(1), 319-352.

[7] Yang, Y., Bai, F., Yu, Z., Shen, T., Liu, Y., & Gong, B. (2024). An Anonymous and Supervisory Cross-Chain Privacy Protection Protocol for Zero-Trust IoT Application. ACM Transactions on Sensor Networks, 20(2), 1-20.

[8] Li, Q., Wen, Z., Wu, Z., Hu, S., Wang, N., Li, Y., ... & He, B. (2021). A survey on federated learning systems: Vision, hype and reality for data privacy and protection. IEEE Transactions on Knowledge and Data Engineering, 35(4), 3347-3366.

[9] López Martínez, A., Gil Pérez, M., & Ruiz-Martínez, A. (2023). A comprehensive review of the state-of-the-art on security and privacy issues in healthcare. ACM Computing Surveys, 55(12), 1-38.

[10] Zhang, P., Wang, Y., Kumar, N., Jiang, C., & Shi, G. (2021). A security-and privacy-preserving approach based on data disturbance for collaborative edge computing in social IoT systems. IEEE Transactions on Computational Social Systems, 9(1), 97-108.

[11] Ying, Z., Cao, S., Liu, X., Ma, Z., Ma, J., & Deng, R. H. (2022). PrivacySignal: Privacy-preserving traffic signal control for intelligent transportation system. IEEE Transactions on Intelligent Transportation Systems, 23(9), 16290-16303.

[12] Peng, L., Feng, W., Yan, Z., Li, Y., Zhou, X., & Shimizu, S. (2021). Privacy preservation in permissionless blockchain: A survey. Digital Communications and Networks, 7(3), 295-307.
[13] Sun, Y., Liu, J., Wang, J., Cao, Y., & Kato, N. (2020). When machine learning meets privacy in 6G: A survey. IEEE Communications Surveys & Tutorials, 22(4), 2694-2724.
[14] Yang, W., Wang, S., HuHu, J., & Karie, N. M. (2022). Multimedia security and privacy protection in the internet of things: research developments and challenges. International Journal of Multimedia Intelligence and Security, 4(1), 20-46.
[15] Zhao, L., Wu, D., & Zhou, L. (2023). Data Utilization Versus Privacy Protection in Semantic Communications. IEEE Wireless Communications, 30(3), 44-50.
[16] Ma, C., Li, J., Ding, M., Yang, H. H., Shu, F., Quek, T. Q., & Poor, H. V. (2020). On safeguarding privacy and security in the framework of federated learning. IEEE network, 34(4), 242-248.
[17] Wang, M., Zhu, T., Zhang, T., Zhang, J., Yu, S., & Zhou, W. (2020). Security and privacy in 6G networks: new areas and new challenges. Digital Communications and Networks, 6(3), 281-291.
[18] Cui, L., Qu, Y., Xie, G., Zeng, D., Li, R., Shen, S., & Yu, S. (2021). Security and privacy-enhanced federated learning for anomaly detection in IoT infrastructures. IEEE Transactions on Industrial Informatics, 18(5), 3492-3500.
[19] Bonomi, L., Huang, Y., & Ohno-Machado, L. (2020). Privacy challenges and research opportunities for genomic data sharing. Nature genetics, 52(7), 646-654.
[20] Kumar, S., Srivastava, P. K., Pal, A. K., Mishra, V. P., Singhal, P., Srivastava, G. K., & Mamodiya, U. (2022). Protecting location privacy in cloud services. Journal of Discrete Mathematical Sciences and Cryptography, 25(4), 1053-1062.
[21] Fabrègue, B. F., & Bogoni, A. (2023). Privacy and security concerns in the smart city. Smart Cities, 6(1), 586-613.
[22] Rani, P., Verma, S., Yadav, S. P., Rai, B. K., Naruka, M. S., & Kumar, D. (2022). Simulation of the lightweight blockchain technique based on privacy and security for healthcare data for the cloud system. International Journal of E-Health and Medical Communications (IJEHMC), 13(4), 1-15.