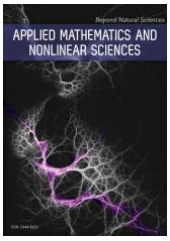
sciendo Applied Mathematics and Nonlinear Sciences, 9(1) (2024) 1-17



Applied Mathematics and Nonlinear Sciences

https://www.sciendo.com

Research on Computer Network Security Vulnerabilities and Encryption Technology in Cloud Computing Environment

Peng Peng[1,†]

1. Admission Office, Guilin Institute of Information Technology, Guilin, Guangxi, 541214, China.

Submission Info

Communicated by Z. Sabir
Received December 18, 2023
Accepted December 23, 2023
Available online January 31, 2024

Abstract

Inadequate network security defense measures threaten the information and property security of the state and the public, and how to safeguard network security is of vital practical significance. This paper proposes a dynamic security threat assessment model and a robust optimal control strategy to improve the efficiency of detecting network vulnerabilities and the accuracy of detecting network threats in a cloud computing environment. It also protects the privacy of the user's identity through a multi-factor continuous authentication method, encrypts and protects the user's data using a homomorphic encryption algorithm, and strengthens the ability of computers to resist intrusion. Three case studies are conducted to verify the effectiveness of the proposed technical approach: dynamic assessment of security threats, control policy, authentication, and network encryption. The results show that in the network vulnerability control policy and authentication case study, when adding the control policy $T_k = 40$, the percentage of network normal nodes, malicious nodes and restorer nodes are 0.98, 0.02, and 0.009, respectively, and the encrypted plaintext ASCII value is distributed in [0,60] in an unordered manner, and the network is in a very desirable security state. Real-time accurate assessments of network security state can be provided by cloud computing-based network security vulnerability and encryption technology.

Keywords: Dynamic assessment model; Optimal control policy; Continuous authentication; Homomorphic encryption; Network security vulnerability.
AMS 2010 codes: 05C82

†Corresponding author.
Email address: 13788333111@163.com

ISSN 2444-8656
https://doi.org/10.2478/amns-2024-0171

1 Introduction

A vulnerability scanner, which is an automated remote program that detects security weaknesses, is typically used in traditional network system security vulnerability detection. Through the use of vulnerability scanners, system administrators are able to timely discover the Web server-related TCP port allocation, services provided, Web services software version and related services software security vulnerabilities in the Internet to provide technical support for the computer network system security defense in a targeted manner, in order to repair the vulnerabilities in a timely manner, to build a security protection barrier [1-3]. Passive and active strategies are commonly employed in vulnerability scanning technology to detect vulnerabilities. Passive strategy is based on the host, and active strategy is based on the network, with the help of the network remote target host to establish a connection, send a timely request, analyze the return information, and then accurately determine the existence of vulnerabilities in the target host with or without vulnerabilities to determine where the vulnerabilities are located [4-6].

Protection of computer network security, in addition to paying attention to the detection of vulnerabilities, encryption and protection of information and networks, in the protection of personal privacy and property is also very critical and important [7-8]. Today's current stage of information technology and computer networks has been able to develop rapidly, and at the same time, it has had a certain impact on contemporary society and, even to a certain extent, has changed the whole world. However, in the whole development process, people should pay attention to the sensitivity and specificity of data and information issues, and at the same time, strengthen the information encryption and protection of the importance of information, and can effectively promote the current computer network communication security [9-12].

To ensure network security and prevent network hacking or other network supply, effective vulnerability detection scanning is crucial in the entire network data information security management system. Chen, Z. and other scholars aim at better-detecting network penetration attacks and propose an attack detection method based on the ant colony classification rule mining algorithm of group intelligence theory, which is tested in simulated experiments and corroborated that the method can effectively identify network vulnerabilities and network penetration attacks, and the recognition rate accuracy is high [13]. Zhang, J. scholars took the intrusion detection system (IDS) as the research object, established the intrusion detection model based on data mining and the traditional IDS method for comparison experiments, and the results of the research show that the intrusion system based on data mining is better in the network vulnerability intrusion detection and network security protection realizations [14]. Amin, A. and other scholars constructed a static and static system based on an ant colony classification rule mining algorithm to better detect network penetration attacks. ...and other scholars constructed a hybrid static and dynamic analysis method to detect vulnerabilities in Android applications, evaluated the model by applying it to a variety of applications, and found that the model can detect information leakage, unfamiliar network requests, and other privacy-compromising vulnerabilities [15]. Jia, H scholars evaluated the performance of mainstream word embedding techniques for detecting vulnerabilities, proposed a supervised framework based on language model (ELMo) for optimization, and verified through examples that the optimized framework demonstrated excellent performance in downstream detection tasks, which could have facilitated the process of vulnerability detection [16]. Nuno Antunes and other scholars designed a general web service vulnerability testing tool design Nuno Antunes and other scholars design a generalized web service vulnerability testing tool design method, which is evaluated by simulated numerical measurements; the method is used in a wide range of scenarios, with higher detection coverage and accuracy and better performance compared to commercial tools in the market [17]. Li, R. Q. scholars conceptualize a cross-domain information sharing key security detection method based on PKG trust gateway. By employing simulated computational analysis methods, it

was discovered that this security detection technique could effectively enhance the accuracy of cross-region information-sharing key detection and increase detection efficiency [18]. Dankwa, S. et al., scholars segmented the edges of a new training dataset with a 2-pixel spacing on the CAPTCHA image and then proposed a text-based CAPTCHA-corrupted depth-divisible convolutional neural network. This model has a more simplified structure and greater cracking accuracy when compared to other cracking techniques [19]. Scholars such as Alsabeh, A. introduced a classification method that helps to categorize and analyze the articles related to P4 development. In addition, STRIDE analysis is also used to check the vulnerabilities (congestion control, load balancing, in-network caching, etc.) related to P4-based applications and give reasonable patching suggestions [20].

The application of related technology and cryptography for transfer or replacement, that is, data encryption technology, combined with related technology, can be the corresponding text information for encryption key processing, converted to the corresponding worthless cipher text, so as to avoid the corresponding text information is easy to read and leakage. Li, J scholars introduced data encryption methods AEC and ECC algorithms, which are the most advanced algorithms, based on symmetric key encryption algorithms and public key encryption algorithms, a data encryption technology program was constructed, which realizes the operation of fast and secure transmission of confidential data [21]. Ma, Z. and other scholars designed a data privacy protection scheme for cross-border blockchain networks and confirmed the security and practicality of this protection scheme through simulation experiments [22]. Alshamrani, S and other scholars envisioned a new lightweight encryption technique called DNA-GA (Deoxyribonucleic Acid-Genetic Algorithm) to enhance the resource-constrained IoT devices' data security sensing capabilities; the scheme was demonstrated to be excellent, more secure and more efficient than the existing methods through simulation experiments using different sensing data [23]. A, J. W et al. scholars conceptualized a wireless broadband stream encryption scheme based on quantization logic mapping. The scheme was evaluated in simulation and showed high security and high resource utilization [24]. A, F. H. et al. explored an attribute revocation scheme based on ciphertext attribute encryption, which delivers the complex encryption and decryption process to a fog server with more efficient overall computational power compared to state-of-the-art techniques [25]. Goel, A. et al. discussed an authentication-based solution to ensure and identify secure access to data; simulation experiments were conducted using MATLAB to corroborate the reliability of the LEOBAT technique [26]. Madni, H. A. et al. scholars proposed a (FHE) method for encrypting the model parameters before sharing. Extensive experiments were conducted to test and analyze the proposed method, which outperformed other existing methods [27].

In this paper, we first establish a dynamic assessment model of network security threats in a cloud computing environment using dynamic heterogeneous redundancy and propose a robust optimal control strategy to improve the detection efficiency and accuracy of the assessment model. Secondly, a lightweight continuous authentication protocol for computer networks is proposed to determine whether a user's identity is legitimate or not through static authentication and continuous authentication in order to protect the privacy of the user's identity and save time and resources. Then, homomorphic encryption algorithms are utilized to strengthen privacy protection during network data matching, resist plaintext attacks, and improve computer network security. Finally, using the gigabit backbone network of University A as an experimental platform, the effectiveness and feasibility of the computer network security vulnerability and encryption technology proposed in this paper are verified through the analysis of dynamic assessment of network security threats, the case study of network vulnerability control strategy and authentication, and network encryption test analysis.

2 Methodology

2.1 Network Security Vulnerabilities and Encryption Techniques Based on Cloud Computing

In recent years, cyber-attacks have frequently benefited from the vulnerabilities of critical information infrastructure weaknesses, and the black industry chain formed around information on network security vulnerabilities has caused information technology risks to be detached from the controllable level and has fallen deep into the dilemma of imbalance between freedom and security. Based on the current situation of network data leakage, strengthening the assessment of network security threats, continuous authentication and information encryption plays an important role in reducing network governance risks and ensuring network information security.

2.1.1 Security Threat Dynamic Assessment Models

Firstly, the control logic and forwarding devices in the cloud computing environment are abstracted into meta-channels for information transmission and processing and then based on the abstract architecture of the key network elements of the cloud computing environment constructed by Dynamic Heterogeneous Redundancy (DHR), an abstract model of network data forwarding and information processing in the cloud computing environment constructed by DHR is established. For the set of input requests X denoted as $x=\{0,1\}$ and the set of output responses Y denoted as $y=\{0,1\}$, the DHR-based dynamic assessment model of network security threats in cloud computing environment is shown in Fig. 1. Where the coding sequence is denoted as:

$$x(t)=\left(x_1(t),x_2(t),\cdots,x_N(t)\right) \tag{1}$$

The DHR channel is represented as:

$$c(t)=\left[c_1(t),c_2(t),\cdots,c_N(t)\right] \tag{2}$$

The output response is:

$$y(x(t)\mid F(t))=\left(y_1\left(x_1(t)\mid F_{e_1}(t)\right),y_2\left(x_2(t)\mid F_{e_x}(t)\right),\cdots,y_N\left(x_N(t)\mid F_{e_x}(t)\right)\right) \tag{3}$$

The ruling translates to:

$$R(y(x(t)\mid F(t)))=x_i(t) \tag{4}$$

Where input X is encoded as $\left\{x_{t,1},\cdots x_{i,N}\right\}$, the encoded sequence is transmitted and processed through different meta-channels alone, and the output of the whole meta-channel is decoded at the receiver to obtain output Y. Because of the existence of the feedback control mechanism, the previous state of the meta-channel affects the transfer probability of the current meta-channel for the secure transmission and processing of data. Each heterogeneous meta-channel is designed under the DHR construction, which makes the random or non-random attack perturbation uncertain under the DHR dynamic selection and feedback demystification conditions.

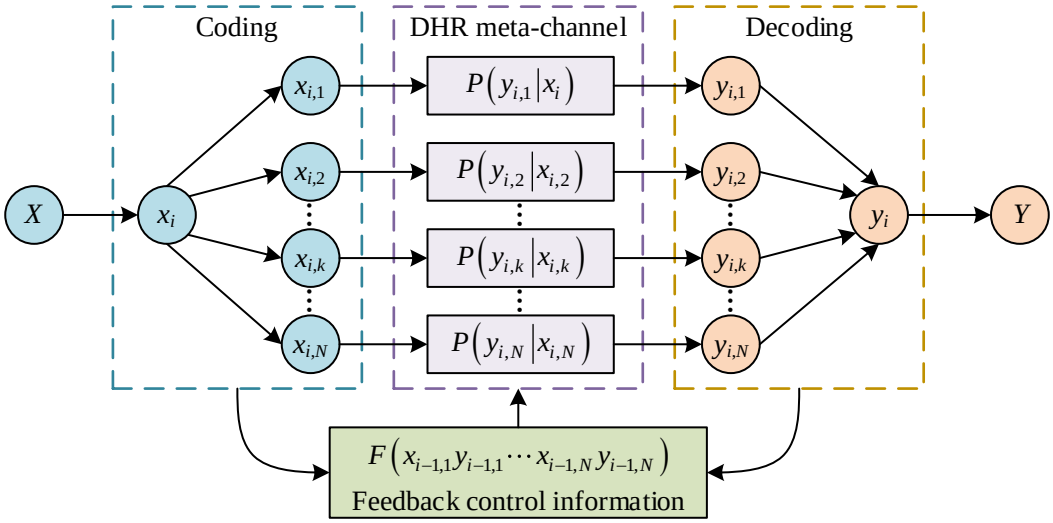


Figure 1. Cloud computing environment network security threat dynamic assessment model

2.1.2 Robust optimal control strategy

In order to improve the detection efficiency of the security threat dynamic assessment model given in the previous section for detecting network vulnerabilities and the detection accuracy of network threats, this subsection proposes a robust optimal control policy.

In the following theorem, the main task, i.e., designing the robustly optimal consistency policy. For simplicity, the following notation is introduced first:

$$\Lambda_k = (1+\varepsilon_1)R_k + (1+\varepsilon_3)\lambda_{\max}\left\{\overline{P}_{k+1}\right\}B_k^T B_k \tag{5}$$

$$\overline{u}^*(k) = \lambda_{\max}\left\{\overline{P}_{k+1}\right\}\Lambda_k^{-1}B_k^T A_k x(k) \tag{6}$$

$$\Omega_k = \lambda_{\max}\left\{\overline{P}_{k+1}\right\}\Lambda_k^{-1}B_k^T A_k = \left[\Omega_k^{(1)T}\Omega_k^{(2)T}\cdots\Omega_k^{(N)T}\right]^T \tag{7}$$

$$V_k = \left[-\overline{H}\otimes C_k\right] = \left[v_k^{(1)T}v_k^{(2)T}\cdots v_k^{(N)T}\right] \tag{8}$$

If there exists a set $\left\{K_k^*\right\}_{0\le k\le K}\pi$ and a set of matrices $\left\{\overline{P}_k\right\}_{0\le k\le K}$ satisfying the following Riccati recursive equations such that the upper bound $\overline{J}_K(u(k))$ of the cost function satisfies Eq. $\overline{J}_K\left(\overline{u}^*(k)\right) \le \overline{J}_K(\overline{u}(k))$, then there are:

$$\begin{aligned}\overline{P}_k =& \left(-\overline{H}\otimes K_k^* C_k + \lambda_{\max}\left\{\overline{P}_{k+1}\right\}\Lambda_k^{-1}B_k^T A_k\right)^T \Lambda_k \\ &\times\left(-\overline{H}\otimes K_k^* C_k + \lambda_{\max}\left\{\overline{P}_{k+1}\right\}A_k^{-1}B_k^T A_k\right) \\ &+H^T Q_k H - \lambda_{\max}^2\left\{\overline{P}_{k+1}\right\}\left(A_k^T B_k A_k^{-1}B_k^T A_k\right) \\ &+\left(1+\varepsilon_2\right)\lambda_{\max}\left\{\overline{P}_{k+1}\right\}A_k^T A_k \\ &+\left(\left(1+\frac{1}{\varepsilon_1}\right)\lambda_{\max}\left\{R_k\right\}+\left(1+\frac{1}{\varepsilon_2}+\frac{1}{\varepsilon_3}\right)\right. \\ &\left.\times\lambda_{\max}\left\{\overline{P}_{k+1}\right\}\lambda_{\max}\left\{B_k^T B_k\right\}\right)\overline{H}_C^2\otimes\delta I C_k^T C_k\Bigg)\end{aligned} \tag{9}$$

And there is:

$$\overline{P}_K = H^T Q_K H, \overline{Q}_K = M^T Q_K M \tag{10}$$

$$\overline{P}_k > 0, \Lambda_k > 0 \tag{11}$$

In addition, consistent control gain K_k^* is:

$$K_k^* = -\left[\Omega_k^{(1)}\Omega_k^{(2)}\cdots\Omega_k^{(N)}\right]\left[v_k^{(1)}v_k^{(2)}\cdots v_k^{(N)}\right]^{\dagger} \tag{12}$$

For a given control strategy of K_k^*, the cost function has an upper bound with the expression:

$$\overline{J}_K^* = x_0^T\overline{P}_0 x_0 \tag{13}$$

Prove that by taking the second order derivative of $\overline{u}(k)$ with respect to $\overline{J}_K$, we obtain $3\,\Lambda_k > 0$, which shows that $\overline{J}_K$ is a convex function with respect to $\overline{u}(k)$. The upper bound on the cost function can be rewritten as follows by matching complete squares:

$$\begin{aligned}\overline{J}_K =& \sum_{k=0}^{K-1}\overline{x}^T(k)Q_k\overline{x}(k)+\left(\overline{u}(k)+\overline{u}'(k)\right)^T\Lambda_k(\overline{u}(k) \\ &+\overline{u}^*(k)\Big)+x^T(k)\left(\left(1+\frac{1}{\varepsilon_1}\right)\lambda_{\max}\left\{R_k\right\}\overline{H}_C^2\otimes\delta I C_k^T C_k\right) \\ &+\left(1+\varepsilon_2\right)\lambda_{\max}\left\{\overline{P}_{k+1}\right\}A_k^T A_k \\ &+\left(1+\frac{1}{\varepsilon_2}+\frac{1}{\varepsilon_3}\right)\lambda_{\max}\left\{\overline{P}_{k+1}\right\}\lambda_{\max}\left\{B_k^T B_k\right\}\overline{H}_C^2\otimes\delta I C_k^T C_k\Bigg) \\ &-\lambda_{\max^2}\left\{\overline{P}_{k+1}\right\}\left(A_k^T B_k\Lambda_k^{-1}B_k^T A_k\right)x(k) \\ &-\overline{x}^T(k)P_k\overline{x}(k)\Big\}+\overline{x}^T(K)Q_K\overline{x}(K) \\ &-\overline{x}^T(K)P_K\overline{x}(K)+\overline{x}^T(0)P_0\overline{x}(0)\end{aligned} \tag{14}$$

Let $\bar{u}(k)=\bar{u}^{*}(k)$, then there is:

$$K_k^{*}=\arg\min norm\left(\left(I_N\otimes K_k\right)V_k+\Omega_k\right) \tag{15}$$

The consistent control gain can be rewritten as:

$$\begin{aligned}K_k^{*}=\arg\min norm\Big(&K_k\left[v_k^{(1)}v_k^{(2)}\cdots v_k^{(N)}\right]\Big)\\&+\left[\Omega_k^{(1)}\Omega_k^{(2)}\cdots\Omega_k^{(N)}\right]\Big)\end{aligned} \tag{16}$$

Available:

$$K_k^{*}=-\left[\Omega_k^{(1)}\Omega_k^{(2)}\cdots\Omega_k^{(N)}\right]\left[v_k^{(1)}v_k^{(2)}\cdots v_k^{(N)}\right]^{\dagger} \tag{17}$$

This can be obtained by bringing K_k^{*} into equation (14):

$$\begin{aligned}\bar{P}_k=&\left(-\bar{H}\otimes K_k^{*}C_k+\lambda_{\max}\left\{\bar{P}_{k+1}\right\}\Lambda_k^{-1}B_k^{T}A_k\right)^{T}\\&\times\Lambda_k\left(-\bar{H}\otimes K_k^{*}C_k+\lambda_{\max}\left\{\bar{P}_{k+1}\right\}\Lambda_k^{-1}B_k^{T}A_k\right)\\&+H^{T}Q_kH-\lambda_{\max}^{2}\left\{\bar{P}_{k+1}\right\}\left(A_k^{T}B_k\Lambda_k^{-1}B_k^{T}A_k\right)\\&+\left(1+\varepsilon_2\right)\lambda_{\max}\left\{\bar{P}_{k+1}\right\}A_k^{T}A_k\\&+\left(\left(1+\frac{1}{\varepsilon_1}\right)\lambda_{\max}\left\{R_k\right\}\right.\\&+\left(1+\frac{1}{\varepsilon_2}+\frac{1}{\varepsilon_3}\right)\lambda_{\max}\left\{\bar{P}_{k+1}\right\}\\&\left.\times\lambda_{\max}\left\{B_k^{T}B_k\right\}\bar{H}_C^{2}\otimes\delta IC_k^{T}C_k\right)\end{aligned} \tag{18}$$

Further, if there is $\bar{P}_k$ satisfying Eq. (9), it is easy to obtain:

$$\bar{J}_K\left(u^{*}(k)\right)=x^{T}(0)\bar{P}_0x(0) \tag{19}$$

Proof of graduation.

According to Eq. (13), the unitary cost function is obtained as $\bar{J}_K^{*}=x^{T}(0)\bar{P}_0x(0)=\bar{x}^{T}(0)P_0\bar{x}(0)$. In fact, this result can be extended to any starting time τ, i.e.:

$$\bar{x}^{T}(K)\bar{Q}_k\bar{x}(K)+\sum_{k=1}^{K-1}\left\{\bar{x}^{T}(k)\bar{Q}_k\bar{x}(k)+\vec{u}^{T}(k)R_k\vec{u}(k)\right\}\le\bar{x}_\tau^{T}P_\tau\bar{x}_\tau \tag{20}$$

Eq. $\bar{u}^{*}(k)$ can be obtained from Eq. (15). Thus, it is seen that $\bar{x}^{T}(\tau)P_\tau\bar{x}(\tau)$ i.e., the upper bound of the cost function is:

$$\bar{z}^T(K)Q_K\bar{z}(K)+\sum_{k=r}^{K-1}\left\{\bar{z}^T(k)Q_k\bar{z}(k)+\vec{u}^T(k)R_k\vec{u}(k)\right\} \tag{21}$$

2.2 Multi-factor continuous authentication

The lightweight continuous authentication protocol for computer networks proposed in this section, which consists of two static authentication phases and a continuous authentication phase, can realize mutual authentication between personal servers and network nodes while protecting the privacy of user identity and saving time and resources. The specific process of continuous authentication is shown in Fig. 2. The continuous authentication system is utilized to gather the user's behavioral feature vectors, and these vectors are used to verify their identity. The continuous authentication system can be divided into the registration and authentication phases. The registration phase involves collecting and storing user behavioral feature vectors and using the collected vectors to train the authentication model for the user. The authentication phase is to upload the collected user behavioral feature vectors into the user identity model generated in the registration phase and to determine whether the user identity is legitimate or not by calculating the user authentication score.

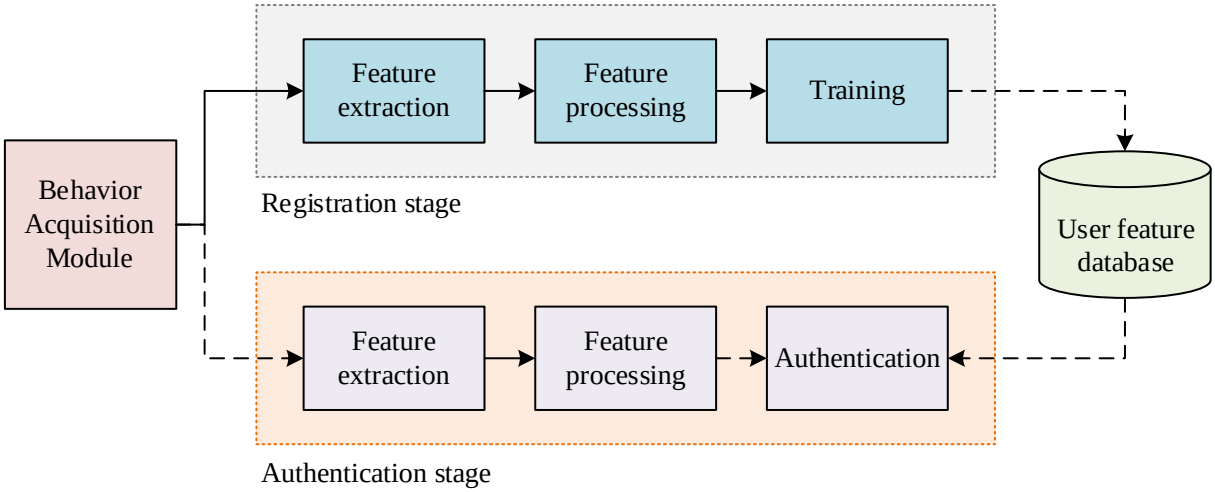


Figure 2. Continuous authentication process

1) Registration phase

This phase is executed when a new sensor node SN wishes to register with the personal server PS over a secure channel. The registration phase proceeds as follows:

(1) SN enters its identity ID_{SN}, private key K_{SN}, computes $V_{SN}=h\left(ID_{SN}\|K_{SN}\right)$, selects a random number r_1 and computes $C_{SN}=h\left(ID_{SN}\|K_{SN}\right)$, and then sends the request message $\{ID_{SN},K_{SN}\}$ over a secure channel to PS.

(2) After receiving the registration request, PS computes $Pub_{PS}=x\cdot G$, $A_{SN}=h\left(h(ID_{SN}\|x)\|V_{SN}\right)$, $B_{SN}=h\left(ID_{SN}\|x\right)\oplus V_{SN}$, where Pub_{PS} is the public parameter of PS and x is the private key of PS 8. Then PS stores $\{ID_{SN},B_{SN}\}$ in the database and sends $\{Pub_{PS},B_{SN}\}$ to SN.

(3) Finally, SN stores tuple $\{Pub_{PS}, B_{SN}, C_{SN}, r_1\}$.

2) Static authentication phase

In this phase, sensor nodes and personal servers authenticate each other over a public channel, and only legitimate personal servers can access the data stored in the sensor nodes. The process of static authentication is as follows:

(1) SN first enters its identity ID_{SN}, bio-key $BKSSN$, private key K_{SN}, which is combined with a random number r_1 stored in the registration phase to verify that Cs is equal to $h(ID_{SN} \| K_{SN} \| r_1)$. If C_{SN} is invalid, the communication is terminated. Otherwise, SN selects a random number N_1, the current timestamp T_1, computes $D_1 = N_1 \cdot G, E_1 = N_1 \cdot Pub_{PS}$, $G_{SN} = ID_{SN} \oplus h(E_1)$, $Z_{SN} = h(E_1) \oplus h(BK_{SN})$, $V_{SN} = h(ID_{SN} \| K_{SN})$, $h(ID_{SN} \| x) = B_{SN} \oplus V_{SN}$, $A_{SN} = h(h(ID_{SN} \| x) \| V_{SN})$, and $W_1 = h(A_{SN} \| E_1 \| ID_{SN} \| T_1)$. and then sends a login message $\{D_1, G_{SN}, Z_{SN}, W_1, T_1\}$ over the public channel to PS.

(2) Upon receiving a message from SN, PS first checks if $|T_c - T_1|$ is less than the time interval ΔT, and if this validation holds, PS computes $E_1 = x \cdot D_1, ID_{sv} = G_{sv} \oplus h(E_1)$, then uses ID_{sv} to retrieve the corresponding $W_1^* = h(A_{SN} \| E_1 \| ID_{SN} \| T_1)$ from the database and checks if W_1^* is equal to W_1. If the equation does not hold, the session is terminated immediately. Instead, PS selects a random number N_2, the current timestamp T_2, computes $M_1 = N_2 \cdot G, O_1 = N_2 \cdot D_1$, $SKs = h(O_1 \| ID_{SN} \| E_1 \| h(BK_{SN}))$, $W_2 = h(M_1 \| SK_S \| E_1 \| T_2)$, and sends $\{M_1, W_2, T_2\}$ it over the public channel to SN.

(3) After receiving the message from PS, SN checks if $|T_c - T_2|$ is less than ΔT. If the verification holds, SN computes $O_1 = N_1 \cdot M_1$, $SKs = h(O_1 \| ID_{SN} \| E_1 \| h(BK_{SN}))$, $W_2^* = h(M_1 \| SK_S \| E_1 \| T_2)$ and checks $h(ID_{SN} \| K_{SN} \| r_1) = C_{SN}$ and checks if W_2^* is equal to W_2. If W_2^* is not equal to W_2, the session terminates immediately. Instead, the current timestamp T_3, SN calculates $W_3 = h(O_1 \| SKs \| T_3)$, $P_{SN} = SKs \oplus h(BK_{SN})$, SN stores P_{SN} and sends $\{W_3, T_3\}$ over the public channel to PS.

(4) Upon receiving a message from SN, PS computes $W_3^* = h(O_1 \| SK_s \| T_3)$ and compares W_3^* with W_3. If they are equal, PS computes $Q_{SN} = SKs \oplus h(x)$ and stores $\{ID_{SN}, Q_{SN}\}$ in its database.

3) Continuous authentication phase

Compared to static authentication, continuous authentication is a more straightforward and lightweight authentication method, and it is utilized for multiple sensory data transmissions within

two adjacent static authentication intervals. The sensor nodes are continuously monitored and authenticated by the continuous authentication mechanism in this phase. The process of continuous authentication is as follows:

(1) SN enters its identity $IDsv$, $Biokey\ BK_{SN}$ private key K_{SN} and checks if $h\left(ID_{SN}\|K_{SN}\|r1\right)=C_{SN}$ holds. If the equation holds, compute $SKs=P_{SN}\oplus h(BK_{SN})$, choose timestamp T_4 and random number N_3 and compute $J_1=N_3\cdot G, K_1=N_3\cdot Pub_{PS}$, $Y_{SN}=ID_{SN}\oplus h\left(K_1\right)$, $V_{SN}=h\left(ID_{SN}\|K_{SN}\right)$, $h\left(ID_{SN}\|x\right)=B_{SN}\oplus V_{SN}$, $A_{SN}=h\left(h\left(ID_{SN}\|x\right)\|V_{SN}\right)$, $W_4=h\left(Y_{SN}\|SKs\|A_{SN}\|T_4\right)\oplus h(B_{SN})$. then SN sends message $\left\{J_1,Y_{SN},W_4,T_4\right\}$ to PS over the public channel.

(2) After receiving the message from SN, PS verifies the validity of T_4 and checks if $\left|T_c-T_4\right|$ is less than ΔT. If this verification holds, PS computes $K_1=x\cdot J_1, ID_{SN}=Y_{SN}\oplus h\left(K_1\right), PS$ to retrieve B_{SN},Q_{SN} from the $\left\{ID_{SN},B_{SN},Q_{SN}\right\}$ database, using the identity ID_{SN}, and computes $SKs=Q_{SN}\oplus h(x), V_{SN}=h\left(ID_{SN}\|x\right)\oplus B_{SN}$, $A_{SN}=h\left(h\left(ID_{SN}\|x\right)\|V_{SN}\right)$, and $h(BK_{SN})^*=W_4\oplus h\left(Y_{SN}\|SK_S\|A_{SN}\|T_4\right)$. and then verifies if $h(BK_{SN})^*$ is equal to $h\left(BK_{SN}\right)$. if they are equal, SN is successfully authenticated.

2.3 Homomorphic encryption algorithms

In recent algebra, let $\langle G,*\rangle,\langle H,\circ\rangle$ be a system of two algebras, $f:G\to H$ is a map from G to H, and f is said to be a homomorphic map from G to H if, for $\forall a,b\in G$, there are both $f(a*b)=f(a)\circ f(b)$. In cryptography, an encryption scheme is essentially a certain mapping from the plaintext space to the ciphertext space. If the mapping is a homomorphic mapping, the encryption scheme is said to be a homomorphic encryption scheme. Specifically, let P be the plaintext space, C be the ciphertext space, Enc be the encryption algorithm, and Dec be the decryption algorithm. Let some operation in the ciphertext space be * and some operation in the plaintext space be 0. Given two plaintexts $m_1,m_2\in P$ and two ciphertexts $c_1,c_2\in C, c_1=Enc\left(m_1\right), c_2=Enc\left(m_2\right)$, the homomorphic encryption scheme satisfies:

$$Dec\left(c_1*c_2\right)=Dec\left(Enc\left(m_1\circ m_2\right)\right) \tag{22}$$

Based on the above definition, it can be seen that homomorphic encryption makes it possible to perform operations on data without decrypting it. The security of secure multi-party computation is made possible by this feature of homomorphic encryption. Hommorphic encryption has significant research and application promotion value, particularly for privacy protection in distributed networks.

In Eq. (22), the encryption scheme is said to be an additive homomorphic encryption scheme if it is an additive operation, i.e., $Dec\left(c_1*c_2\right)=Dec\left(Enc\left(m_1+m_2\right)\right)$. If it is a multiplicative operation, i.e., $Dec\left(c_1*c_2\right)=Dec\left(Enc\left(m_1\times m_2\right)\right)$, the encryption scheme is said to be a multiplicative homomorphic encryption scheme. When a scheme is only capable of satisfying either

additivehomomorphism or multiplicativehomomorphism, it is referred to as a semi-homomorphic encryption scheme. If a scheme satisfies both additive homomorphism and multiplicative homomorphism, it is known as a fully homomorphic encryption scheme. Typical semi-homomorphic encryption algorithms are RSA algorithm, ElGamal algorithm, Pailliar algorithm and so on. Among them, RSA algorithm and ElGamal algorithm have multiplicative homomorphism i.e. $E(m_1)E(m_2)=E(m_1m_2)$. Pailliar algorithm has additive homomorphism i.e. $E(m_1)E(m_2)=E(m_1+m_2)$.

In this paper, Pailliar's algorithm is applied to realize privacy preservation during the matching process, which is briefly described. Let p,q be a large prime number, $n=pq,\lambda=lcm(p-1,q-1)$, defined as:

$$L(u)=\frac{u-1}{n} \tag{23}$$

Suppose g satisfies $\gcd\left(\left(g^{\lambda} \bmod n^2\right),n\right)=1$. To encrypt message $m<n$, choose a random number $r<n$, then the ciphertext is $E(m)=g^m r^n \bmod n^2$, due to:

$$\begin{aligned} E(m_1)E(m_2) &= \left(g^{m_1}r_1^n\right)\left(g^{m_2}r_2^n\right) \\ &= g^{m_1+m_2}\left(r_1r_2\right)^n \\ &= E\left(m_1+m_2 \bmod n\right) \end{aligned} \tag{24}$$

So Pailliar encryption algorithm has additive homomorphism. According to the encryption process described above, with Pailliar encryption, a random number needs to be chosen for each encryption so that even the ciphertext produced by encrypting the same plaintext twice is different. The Pascal algorithm has a greater resistance to chosen plaintext attacks than the RSA algorithm.

3 Results and analysis

Computer networks in a cloud computing environment have both advantages and disadvantages; The network operation process should always pay attention to the maintenance of the system as well as the prevention of security vulnerabilities. This chapter verifies the feasibility and scientificity of the computer network security vulnerability and encryption techniques proposed in this paper from three aspects: dynamic assessment of security threats, control policy and authentication, and network encryption.

3.1 Analysis of dynamic assessment of cybersecurity threats

In order to verify the effectiveness of the security threat dynamic assessment model proposed above, a network attack source threat assessment experiment was carried out in a real computer network environment, utilizing the gigabit backbone network of university A deploying the IDS intrusion detection system as the experimental platform, and taking more than 1352000 consecutive alarm messages as the experimental data. These alarms contain 109 types of alarm events from 12,486 source IP addresses and are sent to 11,814 target IP addresses. The number of alarms sorts all the attack sources, and the top 95 are taken out as the threat assessment objects of the experiment. The number of alarms sorts the results of the experiment after being evaluated by the security threat

dynamic assessment model, and the results of the threat behaviors are evaluated in Figure 3. The number of alarms showed a downward trend, the threat value obtained from the dynamic assessment of security threats and the trend towards roughly the same, but there are individual anomalies, such as the threat value of the attack source in position 48 is 0.716334, the number of alarms is 10453, relatively speaking, the assessment of the existence of a large difference.

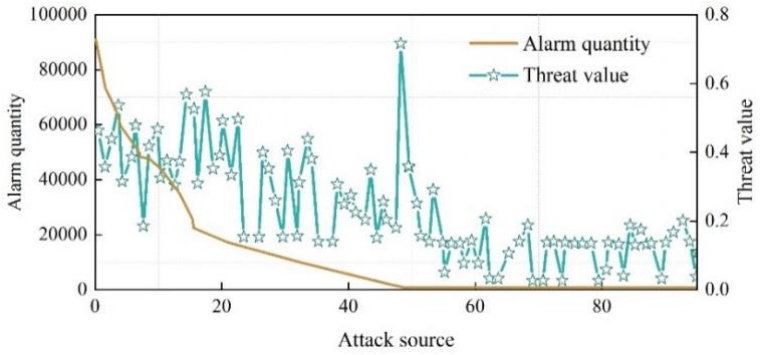


Figure 3. Assessment results of threat behavior

The threat values corresponding to the attack sources are sorted. In order to reflect the difference between the traditional sorting of the number of alarms and the sorting of the evaluation results, the sorting position of the traditional number of alarms is made to differ from the sorting position of the evaluation values. The offset of the sorting position is obtained as shown in Fig. 4. The sorting position is mainly shifted in the interval of -34.273~51.833, and the average sorting offset is 15.537. The experimental results show that, compared with the traditional sorting directly according to the number of alarms, the threat value obtained by the model assessment can more effectively respond to the threat capability of the attack source in its monitoring environment, thus verifying the feasibility of the security threat dynamic assessment model proposed in this paper, which can be used for maintaining the computer network security and the prevention of security vulnerabilities.

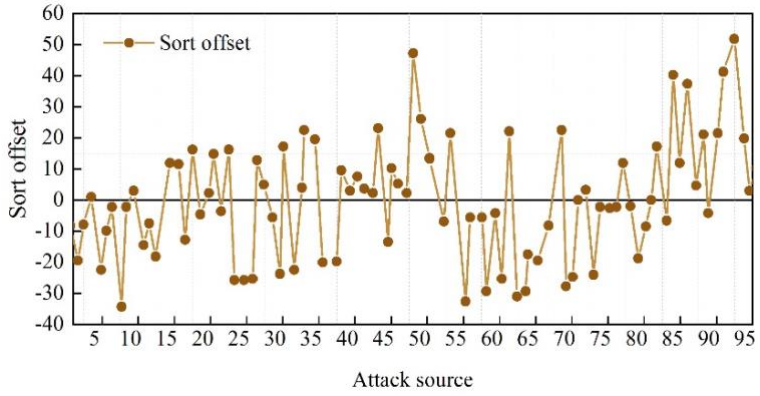


Figure 4. Compare the traditional method to sort the position deviation

3.2 Network Vulnerability Control Policy and Authentication Case Study

In order to verify the effectiveness and scientific validity of the control strategies and authentication techniques mentioned above, this section builds a basic experimental environment with the help of the gigabit backbone network of university A. 800 nodes have been set up to simulate a centerless cloud computing system, with forwarding packets as the main business interaction. In particular, in order to simulate the dynamics of cloud computing more realistically, a continuous time input Λ and node deactivation probability ε are set up, respectively, where input Λ indicates the existence

of a certain number of new nodes joining the network and ε indicates that the node loses its activity and leaves the network with a certain probability.

Comparative experiments are set up for three different scenarios, i.e., no control policy $T_k = 0$ and two control policies with different frequencies $T_k = 40$ and $T_k = 80$. The control policy is to identify and reject the malicious nodes by performing a network-wide authentication with an accuracy of 0.90, where $T_k = 40$ denotes that the network-wide authentication is performed at every 40-second interval. Fig. 5-Fig. 7 show the percentage of susceptible nodes, percentage of malicious nodes, and percentage of restorer nodes under different control strategies, respectively. In Fig. 5, when the control frequency $T_k = 80$, the percentage of normal nodes is increased, and at each control, the percentage of normal nodes is instantaneously increased due to the fact that the malicious nodes are eliminated with an accuracy of 0.90 when performing authentication. When the control policy $T_k = 40$, the percentage of normal nodes can be up to 0.98 or more. Combining the conclusions of Fig. 5, Fig. 6 and Fig. 7, the network will be in a very desirable security state when the control policy $T_k = 40$. In particular, in Fig. 6, the percentage of malicious nodes decreases from 0.338 to 0.009. However, executing the control every 40 seconds not only consumes the necessary execution cost but also interferes with the normal operation of the network functions. For example, under the control policy of $T_k = 40$, 20 times of authentication needs to be executed in the 0-1000s time period, which is obviously an excessive defense. The experimental case study verifies the feasibility of the optimal control policy proposed in this paper, which can control the cost and improve the network operation efficiency according to the needs of network security.

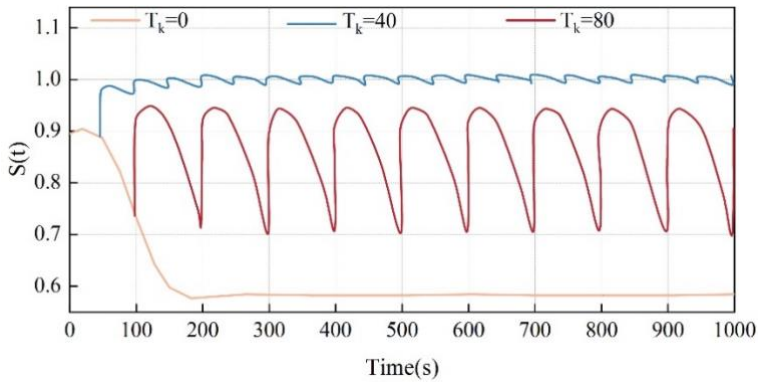


Figure 5. Indicates the proportion of susceptible nodes in different control policies

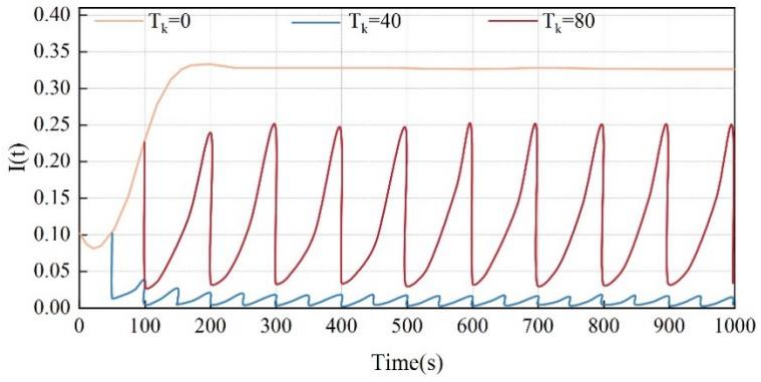


Figure 6. Indicates the proportion of malicious nodes in different control policies

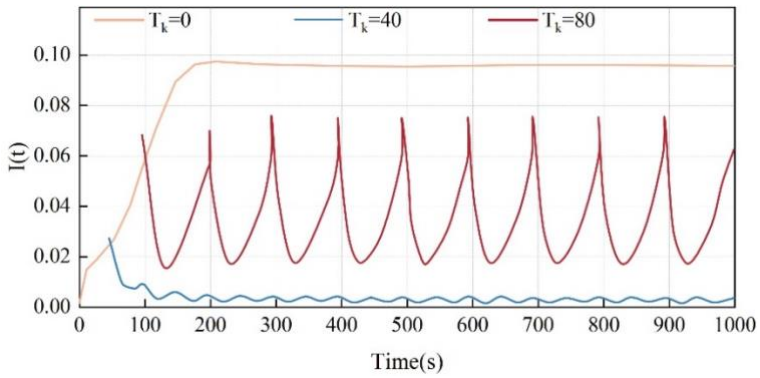
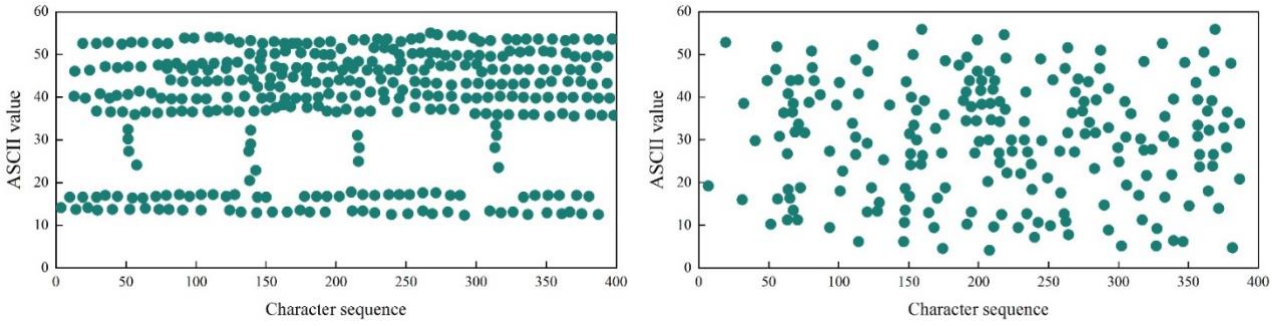


Figure 7. Indicates the number of recoverable nodes in different control policies

3.3 Network encryption analysis

To verify the effectiveness of the encryption algorithm proposed in this paper, the encryption of the computer network is empirically measured by MATLAB in this section. Setting the network security authentication service scenario, a 200m×200m monitoring area is planned in the simulation environment, 250 network sensor nodes are randomly generated, the node raw energy is 120J, and the network topology is randomly deployed. Fig. 8 shows the distribution of ASCII code values of character sequences before and after firewall data encryption under the method of this paper. Fig. 8(a) shows the distribution of plaintext ASCII values before encryption. The character sequence before encryption has obvious regularity and statistical properties, and the ASCII values are mainly concentrated in the interval of [10,20] and [35,55]. Figure 8(b) shows the distribution of plaintext ASCII values after encryption. After encryption, the character sequence is disturbed, and the ASCII values are disorderly distributed in the interval of [0,60], showing chaotic and random patterns, and the law of the initial information is covered up, which effectively resists the malicious infringement such as plaintext attack, key attack, etc., and improves the confidentiality of data stored in the firewall.



(a) Distribution of plaintext ASCII values before encryption

(b) Distribution of plaintext ASCII values after encryption

Figure 8. Distribution of plaintext ASCII values before and after data storage encryption

Simulation experiments are conducted to test the proposed method's performance in achieving secure multi-channel transmission of computer network data. The length of the network data sample is given as 1300, the time interval of data transmission is 0.35s, the bandwidth is 50kHz, and the parameters of the multi-channel transmission are shown in Table 1; there are 8 channels, and the noise ratio, the key threshold, and the confidence level are in the range of 35.712-39.645Db, 0.234-1.174, and 12-15.

Table 1. Sample parameter Settings

Argument	Signal-to-noise ratio /dB	Key threshold	Confidence degree
Channel 1	38.189	0.541	12
Channel 2	38.645	1.124	13
Channel 3	36.114	1.174	14
Channel 4	36.031	0.815	15
Channel 5	37.891	0.234	14
Channel 6	36.285	0.419	13
Channel 7	35.712	0.277	12
Channel 8	36.894	0.817	13

According to the parameter settings in Table 1, the channel dynamic characteristic parameter K=512 is selected, and the computer network data time domain waveform is obtained as shown in Fig. 9, and the amplitude of the data time domain waveform is below 300dB in the sampling time of 0-250s. Taking the data in Fig. 9 as a test object, the data is securely encrypted for transmission, and the data encryption output results are shown in Fig. 10. As there are more sampling points, the amplitude of the data time domain waveform of the communication network changes from -4KPps to 3.99KPps. It can be seen that the steganographic ability of computer network data encryption using the method proposed in this paper is better, and the amplitude can be kept more stable under the multi-channel transmission of computer network data.

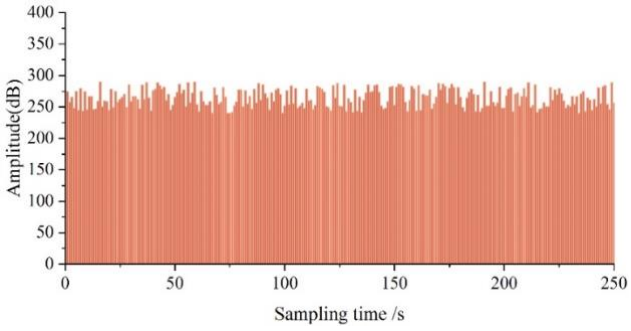


Figure 9. Computer network data time domain waveform

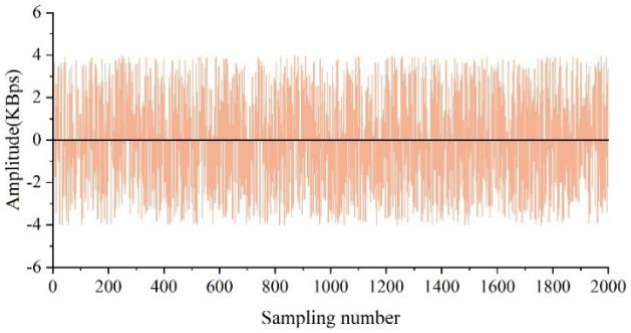


Figure 10. Data encryption output

4 Conclusion

The security vulnerability problem has largely jeopardized the security of information system data and has seriously damaged the rights and interests of users. This paper proposes network security vulnerability and encryption technology based on cloud computing and empirically analyzes from three perspectives: dynamic assessment of security threats, control strategy and authentication, and network encryption, and draws the following conclusions:

1) In the network attack source threat assessment experiment, the traditional sorting position of the number of alarms and the sorting position of the assessment value make a difference to get an average sorting offset of 15.537, which is a large difference, indicating that the threat value obtained from the assessment of the model proposed in this paper can more effectively respond to the threat capability of the attack source in its monitoring environment.

2) In the case study of network vulnerability control policy and authentication, the percentage of susceptible nodes, malicious nodes, and restorer nodes without control policy are 0.58, 0.33, and 0.98, respectively, and the network is in an insecure state. After adding the control policy $T_k = 40$, the proportion of normal nodes reaches 0.98, and the proportion of malicious nodes and malignant nodes tends to 0. The optimal control policy proposed in this paper is effective because the network security state is very ideal.

3) From the analysis of network encryption measurement, the network encryption after the plaintext ASCII value disorderly distribution in the [0,60] interval, in chaotic, random form, and the multi-channel secure transmission, the time domain waveform amplitude is [-4,3.99], the initial information of the stealth is better, to improve the confidentiality of the firewall storage data.

4) The network security vulnerability and encryption technology proposed in this paper can prevent the information system from being attacked or controlled, important information being stolen, user data being tampered with, etc., to reduce the security risk caused by network security vulnerability on the information system, and to provide technical support for the construction of a safe and reliable computer network environment.

References

[1] Zhou, W., Zhang, H., & Li, Q. M. (2017). A network risk assessment method based on attack-defense graph model. Journal of Computers (Taiwan), 28(2), 105-118.

[2] Onawola, H. J., Aliyu, G., Badamasi, B., & Longe, O. B. (2021). A conceptual model for mitigating security vulnerabilities in iot-based smart grid electric energy distribution systems. International Journal of Engineering Research in Africa, 55, 122-131.

[3] Biswas, & Kumar, A. (2017). Source authentication techniques for network-on-chip router configuration packets. Acm Journal on Emerging Technologies in Computing Systems, 13(2), 1-31.

[4] Weiwei, W. U., Su, H. U., Lin, D., & Gang, W. U. (2022). Reliable resource allocation with rf fingerprinting authentication in secure iot networks. Science China Information Sciences, 65(7), 1-16.

[5] Gu, H., Zhang, J., Liu, T., Hu, M., Zhou, J., & Wei, T., et al. (2020). Diava: a traffic-based framework for detection of sql injection attacks and vulnerability analysis of leaked data. IEEE Transactions on Reliability(1), 69.

[6] Lee, S., Kim, S., Choi, K., & Shon, T. (2017). Game theory-based security vulnerability quantification for social internet of things. Future Generation Computer Systems, 82(MAY), 752-760.

[7] Singh, U. K., & Joshi, C. (2018). Scalable approach towards discovery of unknown vulnerabilities. International Journal of Network Security, 20(5).

[8] Lim, M. (2020). Avoiding the most common vulnerability-management pitfalls. Network Security, 2020(7), 12-14.

[9] Tian, Y., & Lu, Z. (2017). Novel permutation-diffusion image encryption algorithm with chaotic dynamic s-box and dna sequence operation. AIP Advances, 7(8), 085008.

[10] Qi, H. (2017). Model of computer network topology optimization based on pattern recognition technology. International Journal of Technology, Management.

[11] Kuilin, C., Xi, F., Yingchun, F., Liang, L., & Xiaoke, T. (2020). Design and implementation of system-on-chip for peripheral component interconnect express encryption card based on multiple algorithms. Circuit World, ahead-of-print(ahead-of-print).

[12] Gao, J. (2017). A support vector machine model for computer network security technology. Boletin Tecnico/Technical Bulletin, 55(12), 564-568.

[13] Chen, Z., Zuo, X., Dong, N., & Hou, B. (2019). Application of network security penetration technology in power internet of things security vulnerability detection. Transactions on Emerging Telecommunications Technologies(2).

[14] Zhang, J. (2019). Detection of network protection security vulnerability intrusion based on data mining. International Journal of Network Security, 21(6), 979-984.

[15] Amin, A., Eldessouki, A., Magdy, M. T., Abdeen, N., & Hegazy, I. (2019). Androshield: automated android applications vulnerability detection, a hybrid static and dynamic analysis approach. Information (Switzerland), 10(10).

[16] Jia, H. (2021). A context-aware neural embedding for function-level vulnerability detection. Algorithms, 14.

[17] Nuno Antunes, & Marco Vieira. (2017). Designing vulnerability testing tools for web services: approach, components, and tools. International Journal of Information Security.

[18] Li, R. Q. (2022). Research on key security detection method of cross domain information sharing based on pkg trust gateway. Journal of Interconnection Networks, 22(Supp01).

[19] Dankwa, S., & Yang, L. (2021). An efficient and accurate depth-wise separable convolutional neural network for cybersecurity vulnerability assessment based on captcha breaking. Electronics, 10(4), 480.

[20] Alsabeh, A., Khoury, J., Kfoury, E., Crichigno, J., & Bou-Harb, E. (2022). A survey on security applications of p4 programmable switches and a stride-based vulnerability assessment. Computer Networks, 207, 108800-.

[21] Li, J. (2017). Research on the application of data encryption technology in network security transmission. Revista De La Facultad De Ingenieria, 32(5), 595-604.2

[22] Ma, Z., Wang, J., Gai, K., Duan, P., Zhang, Y., & Luo, S. (2023). Fully homomorphic encryption-based privacy-preserving scheme for cross edge blockchain network. Journal of systems architecture.

[23] Alshamrani, S. S. B. A. F. (2021). Iot data security with dna-genetic algorithm using blockchain technology. International Journal of Computer Applications in Technology, 65(2).

[24] A, J. W., A, K. H., A, S. F., A, Y. Z., A, H. T., & B, G. J., et al. (2020). A logistic mapping-based encryption scheme for wireless body area networks. Future Generation Computer Systems, 110, 57-67.

[25] A, F. H., B, M. W. A., A, S. T., B, G. A., & C, Z. H. A. (2021). A revocable and outsourced multi-authority attribute-based encryption scheme in fog computing. Computer Networks.

[26] Goel, A., Sharma, D. K., & Gupta, K. D. (2021). Leobat: lightweight encryption and otp based authentication technique for securing iot networks. Expert Systems.

[27] Madni, H. A., Umer, R. M., & Foresti, G. L. (2023). Swarm-fhe: fully homomorphic encryption-based swarm learning for malicious clients. International Journal of Neural Systems, 33(08).