Research on the Security Data Transmission based on Linux

ShuLiang Li¹ Jie zhang² Youzhu Liu

¹ College of Mathematics and Computer Science, Xinyu University Xinyu, China
² College of Mathematics and Computer Science, Xinyu University Xinyu, China
*Department of Information Engineering

Abstract: This paper develops system software in application layer based on Linux operating system. Different transmission modes are adopted in different parts to improve the transmission efficiency in the network. The data capture encryption method is used for the file data, and the storage transmission method is used for the image data. It designs an identity authentication protocol and key exchange protocol based on digital certificate and digital sign technology, to avoid the sophistication and pilfer and improve the reliability of the identity authentication and the security of the key exchange. The common algorithms of symmetric encryption, asymmetric encryption and information digest are studied by using OpenSSL standard library function to realize data encryption module of the system.

1 Introduction
With the rapid development of Internet technology and the improvement of information infrastructure, computer network has become a very important and fast means for people to acquire and exchange information. However, due to the openness of network system, the sharing of information resources, the sharing of communication channels and the diversity of connection forms, information processing, storage, transmission and use have serious problems. Vulnerability can easily be interfered, abused, omitted and lost, and even makes important data transmitted through the network vulnerable to theft, forgery, tampering and destruction by illegal elements. In this paper, the embedded system is used for network transmission, and the embedded system is used as the security node server of the network. The protection of network transmission data is realized on the embedded system.

2 Overall Design of System Software
The information security transmission platform adopts embedded Linux system, and on this basis, the functional modules of the system are developed. Embedded Linux system has the characteristics of small size and stable performance, which is very suitable for embedded development. The system software is modularized according to its functions. The main function modules include network transmission module, identity authentication module, data encryption module and digital image watermarking module. The system software design structure diagram is shown in Figure 1.
3 Design of Network Transmission Module

Network transmission module is the basic module of the whole platform network transmission, responsible for the network transmission of platform data. The embedded system is used as the node server in the network, and the corresponding security operations are carried out on it. The design of different transmission modules for file data and digital image data improves the efficiency of network transmission.

3.1 Design of File Data Transmission Submodule

Packet capture technology is used for file transmission. The segment-to-segment communication mode is adopted between the two clients, that is, when the client and the client send file data, the node server is transparent to the client, and the client sends the data directly to the other client. The node server SN A captures the entire IP packet sent by the client. After the node server captures the IP packet, the whole IP packet is encrypted and encapsulated into a new TCP message, which is sent to the node server on the opposite side by TCP connection. The end-node server SN B extracts the encrypted IP data packet, decrypts it, and sends the original IP data packet to the destination client CN B2 by using the original socket method. The flow of file data transmission is shown in Figure 2.
3.2 Design of image data transmission sub-module
For the network transmission of image data, the storage and forwarding mode is adopted. First, the client sends the image data to the node server, embeds the digital watermarking on the node server, and embeds the time label. Then the whole picture is sent to the target node server, and the watermarking information is extracted on the target node server to verify whether the image has been tampered with, and extract the time label. The target server sends the image data and extracted watermarking information to the target client.

Node servers may simultaneously handle multiple client-side transmission tasks, so node servers work in TCP concurrent mode. The parent thread is responsible for listening for the connection port. When there is a connection request, the parent thread derives the corresponding sub-thread to receive image data, and then the sub-thread completes the embedding of image watermarking.

The whole transmission line is programmed by Socket, and the image data is transmitted reliably by TCP. Images are deleted immediately after embedding and extracting watermarks on embedded node servers to improve the security of the system.

4 Design of Identity Authentication Module
The identity authentication module of information security transmission platform adopts the technology of combining digital certificate and digital signature, and adopts the form of simple PKI, which can effectively solve the problem of identity authentication of network users.

After the communication parties establish TCP connection, the system identity authentication begins. Each embedded node server has a digital certificate issued by CA and a CA root certificate. The two node servers in communication exchange their own digital certificates first, and then the CA root certificate verifies the validity of each other's digital certificates. If the authentication fails, the identity of the other party is illegal. Then the node server digests the message digest of the authentication information together with the timestamp and sends it to the other party, which verifies the validity of the signature. If the signature is verified, the identity of the communicating party is proved to be legitimate. The authentication process of the system identity authentication module is shown in Figure 3.

Combining Hash function with public key algorithm can not only provide data integrity, but also ensure data authenticity. Integrity guarantees that the transmitted data is not tampered with, while authenticity guarantees that the Hash generated by the identified legitimate users is not counterfeited by others. Therefore, the problem of identity authentication for network users can be effectively solved.

Identity authentication module mainly includes certificate verification sub-module, digital signature and verification sub-module.

5 Design of Data Encryption Module
The data encryption module mainly implements the function of encrypting and decrypting data. It can realize the commonly used symmetric and asymmetric encryption algorithms, provide the protection of data security, and realize the commonly used digest algorithm, provide the protection of data integrity. Data encryption module mainly includes symmetrical encryption sub-module, asymmetrical encryption sub-module, digest sub-module and key anti-quotient sub-module. Module uses OpenSSL standard library function to implement various encryption algorithms.

5.1 Design of Symmetric Encryption Submodule
Symmetric encryption sub-module can implement common symmetric encryption algorithms, such as RC4, DES, 3-DES. The library functions of OpenSSL are encapsulated into three main functions: CryptInit initializes the encryption structure according to the name of the encryption algorithm, CryptUpdate encrypts the data, and CryptFinal processes the end data. Whether the data is encrypted or decrypted is controlled by the parameter type of the function CryptInit. If type = 1, the encrypted structure is initialized to the encrypted structure. If type = 0, the encrypted structure is initialized to the decrypted structure.

The encryption process first initializes the encryption structure, then repeatedly calls the function CryptUpdate to process a continuous block of data, and finally calls the CryptFinal function to process a piece of data. The encryption process is shown in Figure 4.

5.2 Design of Asymmetric Encryption Submodule
The asymmetric encryption sub-module mainly implements RSA algorithm. RSA key pairs are obtained from digital certificates. The module includes two parts: key initialization and encryption.

5.3 Design of Digital Summary Submodule
The digital digest sub-module mainly protects the integrity of the transmitted data, and can implement the main digital digest algorithms, such as MD5, SHA and SHA1. Segmentation method is used to digest data, which can effectively solve the problem of digitally digest data. OpenSSL library function is used to realize digest function. OpenSSL library function is encapsulated into three main functions: function HashInit initializes digest structure according to digest algorithm name, function WINHash digests data, function HashFinal calculates and outputs digest result of data.

![Fig4 Flow chart of symmetric encryption sub-module](attachment:image)

The process of digest is to initialize digest structure according to the name of digest algorithm, then call function WINHash repeatedly to process data blocks, and finally call HashFinal function to output digest structure. The operation of digit digest is shown in Fig. 5.

In the digit summary part, the method of segment Hash is used, that is, a file is read into memory by segments for Hash operation, the result of segment Hash is saved in the Hash structure, and the Hash result is output when all the file operations are settled, thus avoiding the memory overflow caused by Hash reading all the larger files into memory at one time.

5.4 Design of Key Agreement Submodule
The main function of key agreement sub-module is to negotiate session key and image watermarking key used in data transmission between server nodes. The length of the key is determined by the originator node server according to the encryption algorithm. The asymmetric encryption algorithm is used to protect the session key, and the public key of the server of the other node is used to protect the session key. Only the other party can decrypt the session key with its own private key, so as to effectively protect the security of the transmission of the session key in the network.

The process of session key agreement is that node server A generates a random number RA, reads the public key of B from the certificate of node server B, encrypts it with the public key of B and sends it to node server B, and notifies user B of the purpose of the key negotiated (encryption or digital watermarking). B generates a random number RB which is encrypted with the public key of A and returned to A. Both sides decrypt the key sent by the other side using their private key. Data, session key RAIIRB is obtained by using pre-negotiated key combination rules.

6 Client Software Design
The client software of the platform runs on the user's ordinary computer and is developed with Visual C++. The client software mainly includes three modules: user interface module, data management module and network transmission module.

7 Conclusion
This paper studies and designs the software structure and design idea of the secure transmission platform.
The embedded Linux operating system is studied and implemented on the development platform; the network topology and transmission mechanism of the system are introduced; the identity authentication and key exchange protection of the platform are introduced; the data encryption algorithm of the platform is introduced and the data encryption is realized by using OpenSSL cryptographic algorithm library. Finally, the design of client software of the platform is introduced.

Reference
[1] H.Chien, J.JAN, An efficient and practical solution to remote authentication, Computers and Security[J], 2002.21(4):372-375
[2] K.Wang, S.Chen, Weaknesses and improvements of an efficient password base remote user authentication scheme[J], IEEE Transactions on Consumer Electronic, 2004, 50: 204-206
[3] Wang Fei, Liu Jianqun, Liu Lushan, Li Shiyong. Touch screen driver design based on uClinux [J]. Microcomputer information, 2007, 3-2: 81-83
[4] Al-Riyami SS,Paterson KG Certificateless public key cryptography.[J]. Advances in Cryptography-Asiacry-pt’03 . 2003
[5] Arthur W,Challener D,Goldman K. A Key Management[D]. Practical Guide to TPM 2.0 . 2015