IOSR Journal of Computer Engineering (IOSR-JCE)
ISSN: 2278-0661, ISBN: 2278-8727, PP: 58-61
www.iosrjournals.org

FTP Security using face recognition & Dynamic password

Mr.A.T.Sonale[1], Mr.S.S.Matsagar[2]
[1](*Department of CST, DOT / Shivaji University,India)*
[2](*Department of CSE, SMCET/JNT University, India)*

ABSTRACT : *File Transfer Protocol (FTP) is widely used protocol for transferring files over a network. However, there exists some secure vulnerability in the protocol. For example, both passwords and files are transmitted in plaintext. Although some new FTPs such as FTPS have been proposed and applied to overcome these vulnerabilities, there are many drawbacks such as lack of flexibility in use, failing to meet specific security requirements, etc. Given these facts, the FTP and its requirements are studied deeply and a new secure FTP system is designed in this paper. In the new system, a dynamic password mechanism is combined with face recognition technology to achieve mutual authentication, key distribution and secure information transmission. The security level selection mechanism is adopted to meet individual security requirements. The resource access control mechanism is used to keep the server from unauthorized access attacks. Analysis shows that compared with existing FTP systems, the new system makes not only data transmission securer but also system in use easier, more flexible and efficient.*

Keywords- *Dynamic password, Interrupted Resume, Hash value, XOR function, Face recognition value*

1. INTRODUCTION

File Transfer Protocol is the simplest way to exchange files over the Internet. With FTP you can access large repositories of documents, software applications and multimedia files stored on thousands of public FTP computers and download them to your personal computer. FTP is the protocol used on the Internet for uploading & downloading files. Other protocols are capable of transferring files as well, including HTTP (Hypertext Transfer Protocol) and SMTP (Simple Mail Transfer Protocol). HTTP is used to transfer web pages from a server to a user's browser and SMTP transfers electronic mail across the Internet. Unlike HTTP and SMTP, FTP is designed specifically for transferring files of all kinds – large and small, binary and text directly from one system to another. However, there exists some secure vulnerability in the protocol. For example both passwords and files are transmitted in plaintext. Although some new FTPs such as FTPS have been proposed and applied to overcome these vulnerabilities, there are many drawbacks such as lack of flexibility in use, failing to meet specific security requirements, etc. Given these facts, the FTP and its requirements are studied deeply and a new secure FTP system has been proposed. In the new system, face recognition to achieve secure authentication, key distribution and secure information transmission. The security level selection mechanism is adopted to meet individual security requirements. The resource access control mechanism is used to keep the server from unauthorized access attacks. With such features, a new FTP system makes not only data transmission securer but also system in use easier, more flexible and efficient.

2. RELATED STUDY

The original FTP specification is an inherently insecure method of transferring files because there is no method specified for transferring data in an encrypted fashion. This means that under most network configurations, user names, passwords, FTP commands and transferred files can be captured by anyone on the same network using a packet sniffer. The general solution is to use either FTP over SSH which brings SSH encryption into FTP system, or FTPS (FTP over SSL) which brings SSL encryption into FTP system. Although they can overcome the fatal weakness of data transmission in plaintext, there are still some disadvantages, such as data connections security, system cost and meeting specific security requirements etc. FTP over SSH uses multiple TCP connections, which is particularly difficult to tunnel over SSH. For SSH clients, attempting to set up a tunnel as the control connection will only protect that connection; when data are transferred, the FTP

software at either end will set up new TCP connections (data connections) which will bypass the SSH connection, thus saving no confidentiality, integrity protection and etc. FTPS is an extension to FTP that adds the support for Transport Layer Security (TLS) and the Secure Sockets Layer (SSL) cryptographic protocols. However, it relies on TLS and SSL security protocol at the transport layer, which can not meet specific security requirement, such as flexible security level selection and resource access control and etc. And secure authentication and information transmission implemented by a credible third-party, certificate and public key cryptography will add more complexity and burden to the system. New proposed secure FTP system, which adopted the public key cryptography mechanisms to solve the secure certification and transmission. But asymmetric key algorithms are generally much more computationally intensive than symmetric key algorithms, which are typically hundreds to thousands of times slower than symmetric key algorithms in practice; it takes a more expensive computational cost to achieve their solutions.

So there is need of Secure FTP system which minimizes the computation cost by using symmetric key cryptography. The proposed system neither adopts authentication techniques such as PKI or Kerberos and etc, nor adopts public key cryptography to encrypt messages. Instead we propose a new secure FTP system based on combination of dynamic password, Face recognition as well as the hash function and symmetric key algorithms to achieve its high security and efficiency.

III. ARCHITECTURE OF SYSTEM

The system will utilize separate control and data connections between a client and a server during the session. At first the client makes a connection to the server called the control connection, which remains open in duration of the session. Whether the system runs in an active or passive mode will determine how a second connection is established. This connection, called the data connection, is opened as required to transfer file data. It is dynamically created as needed by control connection before each data transmission. After the file data transmission is finished, the data control is closed. Similarly, after the session is finished, the control connection is closed.

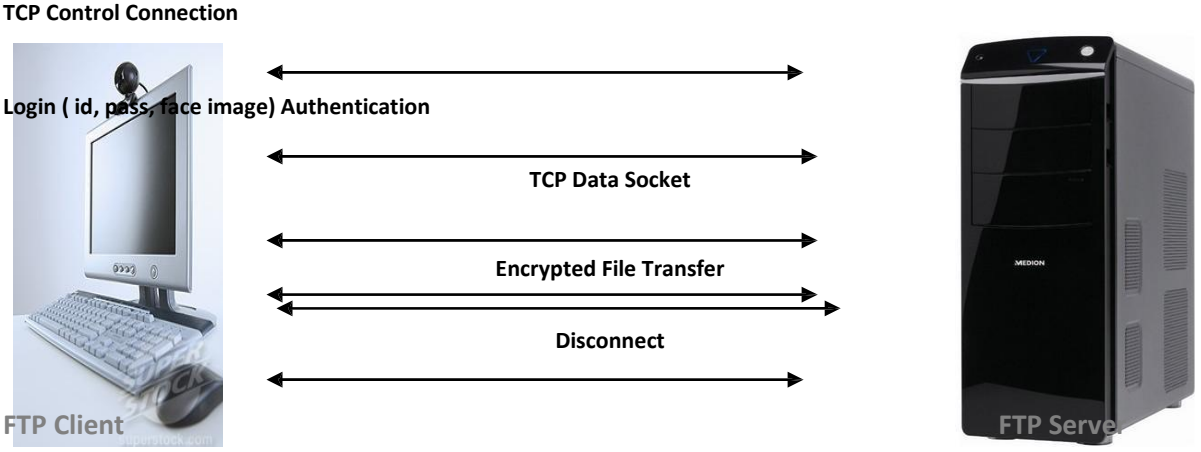

Fig: Secure FTP System

4. WORKING PRINCIPLE

The Secure FTP System works in following three Processes:

4.1 Registration Process
4.2 Authentication Process
4.3 Secure Transfer Process

4.1 Registration Process:

4.1.1 The Registration process is manual process. User must present physically at server system.
4.1.2 The System asks user for personal information and requests for user ID.
4.1.3 If the ID is available then Face Image has been retrieved by using Face recognition device and store into the database.
4.1.4 System also asks for password to the user, after submitting password a hash value of that password is saved in database instead of original password value.

4.2 Authentication Process:

4.2.1 When a registered user logs into the system, he needs to enter ID, password and its Face Image by using Face reader device.
4.2.2 The client makes connection to the server. Server generates random number Ns & sent it to the client.
4.2.3 After receiving Ns from Server, Client computes F (Ns) = (Ns O H (Pw)). Where H (Pw) is hash of password & F is an Ex-OR Function. Client generates random number Nc & computes Hc = H (Nc). At last ID, F (Ns),Hc & Face recognition value is encrypted with Secret key which is integrated in both the client & server software.
4.2.4 The Server then decrypts message & retrieves ID, Face recognition value & H(Pw) from F (Ns).Server then checks whether user ID, H(Pw) and Face Image matches with stored templates. If match is found then user can make use of FTP service.
4.2.5 If the match is not found then authentication fails.

4.3 **Secure Transfer Process:**

4.3.1 After successful authentication user can interact with server in very secure manner.
4.3.2 Every message sent from client to server or vice versa goes into encryption procedure.
4.3.3 Encryption: The message M to be sent, Its hash value H(M) is used to calculate Z = (M || H (M)). This Z is encrypted with shared key Ks and an encrypted message Eks (Z) is sent to the receiver.
4.3.4 Decryption: At receiving end the Eks (Z) is decrypted by using shared key Ks to retrieve Z. Z is then decompresses to retrieve to get M & H(M). Finally hash value of M is taken and compares with H(M). If both are same then M is received or it will be rejected.

5 RESULTS

The system uses face recognition, Dynamic password, Resource access level, Security level selection mechanism to create most secure & efficient FTP System. This fact can be analyzed based upon following views.

5.1 Preventing Replay Attacks:
During each authentication session, the server and clients will respectively generate a random number (called nonce) Ns and Nc, the nonce must be different and cannot be reused. Moreover, the system sends the function value of the nonce (e.g. f(Ns)=(Ns O H(PW)) which is encrypted (e.g. Eks(ID || f(Ns || He
|| Kc)) instead of nonce itself to the other end. Therefore, it is almost impossible for any attacker to capture this encrypted value and replay it on another session, because each session has a different nonce to compute different f(Ns) or f(Nc) So this authentication scheme can prevent replay attacks that may come from C to S or S to C.

5.2 Preventing Impersonation Attacks**:**
System makes use of face recognition value of user & shared key Ks is embedded in server & client softwares. So that it becomes difficult for attackers to personate authorized users, nor can they personate the server to cheat authorized users. This realizes a mutual authentication.

5.3 Ensuring Security of Password Storage and Data Transmission

The system stores password in a form of hash function instead of a plaintext, which stops the attackers from getting the password itself. Moreover, during each session, all the information including authentication information and file data is encrypted by symmetric key algorithms. So the use of hash function ensures information integrity and prevents information from being pried or modified.

5.4 Preventing Unauthorized Access

FTP server of the system provides resource access control scheme to meet users' different access control requirements, such as resource denied access, permitted access, IP limited access and etc. When a user tries to access some files unauthorized to him, the system will stop his unauthorized access operation, thus protecting files in the server.

5.5 Lower Computation Cost**:**

The FTP system Authentication process only computes hash once. Client and server need one hash, one XOR, one symmetric-key encryption and one symmetric-key decryption, one nonce generation and storage computing respectively. In contrast to this scheme has twice fewer hash computing. Adopting symmetric cryptography instead of asymmetric cryptography also decreases the computing cost. In addition, the generated nonce is only used in hash, XOR and authentication computing process, so it need not be stored in other processes. This character reduces server cost and improves sever operation efficiency greatly.

5.6 Higher Safety Strength of Identity Authentication**:**

System uses face recognition value of a user which is unique, also make use of Dynamic password resulting into the strong authentication process. During single session client & server authenticate each other by generating nonce & replying back with the nonce. In addition to this login data is encrypted before sending over network. So that system preserves higher safety strength of Identity Authentication.

6. CONCLUSION

This paper shows drawbacks of existing FTP systems & also proposes a new Secure FTP system.
The new System makes use of Face recognition, Dynamic password, Resource access levels, and security levels & as shown in Result Analysis phase provides most securer & efficient solution as compared to existing FTP system.

REFERENCES

[1] RFC-959 J. Postel, J. Reynolds, l SI, "File Transfer Protocol (FTP)," Oct 1985. Available: http://www.ietforg/rfc/

[2] RFC 4217: P. Ford-Hutchinson, IBM UK Ltd, "Securing FTP with TLS," Oct 2005. Available: http://www.ietforg/rfc/

[3] RFC 4251: T.Ylonen, T. and C. Lonvick, Ed. Cisco Systems, Inc, "The Secure Shell (SSH) Protocol Architecture," Jan 2006. Available: http://www.ietforg/rfc/

[4] Y Ma, H. T. Liu, B. Y Cai, "Design and implementation of a secure FTP system," Applications and Software; 2007.

[5] Liu Xia, Feng Chao-sheng, "A FTP Mutual Authentication Scheme Based on Dynamic Password", Communications Technology; 2010.

[6] Liu Xia, Feng Chao-sheng, Yuan Ding, Wang Can, "A Design of Secure FTP System", Communications, circuits and systems; 2010.

[7] Liu Cuilin, Shen Yongjun, Zang Guidong, Wen Fang, "E-mail System based on Dynamic password and Face Recognition"; 2010