International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol. 3, Issue. 3, May - June 2013 pp-1811-1814 ISSN: 2249-6645

Security Enhancement of Single Sign on Mechanism for Distributed Computer Networks

Jean Jacob[1], Mary John[2]
[1](Information Technology, Rajagiri School of Engineering & Technology, India)
[2](Department of Information Technology, Rajagiri School of Engineering and Technology, Rajagiri Valley, Cochin, India

ABSTRACT: *Single sign-on mechanisms allow users to sign on only once and have their identities automatically verified by each application or service they want to access afterwards. There are few practical and secure single sign-on models, even though it is of great importance to current distributed application environments. Most of current application architectures require the user to memorize and utilize a different set of credentials (eg, username/password or tokens) for each application he/she wants to access. However, this approach is inefficient and insecure with the exponential growth in the number of applications and services a user has to access both inside corporative environments and at the Internet. Single sign-on (SSO) is a new authentication mechanism that enables a legal user with a single credential to be authenticated by multiple service providers in distributed computer networks. Recently, Chang and Lee proposed a new SSO scheme and claimed its security by providing well-organized security arguments. In this paper, however, it is shown that their scheme is actually insecure as it fails to meet security during communication. This paper shows the Chang & Lee scheme and it aims to enhance security using AES encryption and decryption. Implementation is done using socket programming in Java.*

KEYWORDS: *Authentication, Attacks, Decryption, Encryption, Single Sign on*

I. INTRODUCTION

Identification of user is an important access control mechanism for client–server networking architectures. The goal of a single sign on platform is to eliminate individual sign on procedures by centralizing user authentication and identity management at a central identity provider. In a single sign-on solution, the user should seamlessly authenticated to his multiple user accounts (across different systems) once he proves his identity to the identity provider. Nevertheless, in many current solutions, the user is required to repeat sign on for each service using the same set of credentials, which are validated at the identity provider by each service.

User authentication [3], [4] plays a crucial role in distributed computer networks to verify the legacy of a user and then can be granted to access the services requested. To prevent bogus servers, users usually need to authenticate service providers. After mutual authentication, a session key may be negotiated to keep the confidentiality of data exchanged between a user and a provider [4], [5], [6]. In many scenarios, the anonymity of legal users should be protected as well [4], [7], [6]. These protocols offer varying degrees of efficiency. This paper aims to ensure more security to the existing Chang Lee SSO scheme. It also aims to add additional security during data transfer between user and provider. It also proposes further research into more efficient enhancements to the current work. The main objective of this paper is to enhance security for single sign-on solutions and eliminate the need for users to repeatedly prove their identities to different applications and hold different credentials for each application.

II. RELATED WORKS

In 2000, Lee and Chang [4] proposed a user identification and key distribution scheme to maintain user anonymity in distributed computer networks. Later, Wu and Hsu [8] pointed out that Lee-Chang scheme is insecure against both impersonation attack and identity disclosure attack. Meanwhile, Yang et al. [9] identified a weakness in Wu-Hsu scheme and proposed an improvement. In 2006, however, Mangipudi and Katti [10] pointed out that Yang et al.'s scheme suffers from DoS (Deniable of Service) attack and presented a new scheme. In 2009, Hsu and Chuang [11] showed that both Yang et al. and Mangipudi-Katti schemes were insecure under identity disclosure attack, and proposed an RSA-based user identification scheme to overcome the drawbacks.

On the other hand, it is usually not practical by asking one user to maintain different pairs of identity and passwords for different service providers, since this could increase the workload of both users and service providers as well as the communication overhead of networks. To tackle this problem, single sign-on (SSO) mechanism [12] has been introduced so that after obtaining a credential from a trusted authority, each legal user can use this single credential to authenticate itself and then access multiple service providers. Intuitively, an SSO scheme should meet at least two basic security requirements, i.e., soundness and credential privacy. Soundness means that an unregistered user without a credential should not be able to access the services offered by service providers. Credential privacy guarantees that colluded dishonest service providers should not be able to fully recover a user's credential and then impersonate the user to log in other service providers. Formal security definitions of SSO schemes were given in [13]. Chang and Lee made a careful study of SSO mechanism. Firstly, they argued that Hsu-Chuang user identification scheme, actually an SSO scheme, has two weaknesses: (a) An outsider can forge a valid credential by mounting a credential forging attack since Hsu-Chang scheme employed naive RSA signature without any hash function to issue a credential for any random identity selected by a user ; and (b) Hsu-Chuang scheme requires clock synchronization since timestamp is used in their scheme. Then, Chang and Lee presented an interesting RSA based SSO scheme, which is highly efficient in computation and communication (So it is suitable for mobile devices), and does not rely on clock synchronization by using nonce instead of timestamp. Finally, they presented well-organized security

analysis to show that their SSO scheme supports secure mutual authentication, session key agreement, and user anonymity. In [13], Han et al. proposed a generic SSO construction which relies on broadcast encryption plus zero knowledge (ZK) proof showing that the prover knows the corresponding private key of a given public key. So, implicitly each user is assumed to have been issued a public key in a public key infrastructure (PKI). In the setting of RSA cryptosystem, such a ZK proof is very inefficient due to the complexity of interactive communications between the prover (a user) and the verifier (a service provider). Therefore, compared with Han et al.'s generic scheme, Chang-Lee scheme has several attracting features: less underlying primitives without using broadcast encryption, high efficiency without resort to ZK proof, and no requirement of PKI for users.

Table I: Notations used in the algorithm

Notations	Descriptions
SCPC	A trusted authority
U_i, P_j	The user and the service provider, respectively
ID_X	The identity of the entity X
S_X	The secret token of the entity X
e_X	The public key of the entity X
d_X	The private key of entity X
$E_K(M)$	A symmetric encryption of plaintext M using a key K
$D_K(C)$	A symmetric decryption of ciphertext C using a key K
$h(\cdot)$	The one-way hash function
$\|\|$	The concatenation operator

III. PROPOSED SCHEME

The notations used in the algorithm are explained in Table I. The scheme consists of three phases:

A. *System Initialization Phase*

SCPC does the following

1. selects large two primes p, q and computes p*q.
2. determines the key pair (e,d) such that e *d ≡ 1mod φ(N),where φ(N)=(p − 1)*(q − 1).
3. chooses a generator g over the finite field Z∗n, where n is a large odd prime number.
4. SCPC protects the secrecy of d and publishes (e,g,n,N).

B. *Registration Phase*

1. Each user U_i registers a unique identity $ID_i$with a fixed bit length.

2. Obtain a secret token $S_i=(ID_i||h(ID_i))^d$ mod N, from the SCPC through a secure channel where h(·) is a cryptographic one-way hash function.

C. User Identification Phase

U_i submits the request with a random nonce n1, m1 to P_j . On receiving m1, P_j chooses a random number k and then generates a random nonce n2. P_j calculates $Z = g^k$ mod n, u = h(Z||ID_j ||n1), and the signature $v = (u||h(u))^{dj}$ mod N_j .Next, P_j sends the message m2 = {Z,v,n2} back to U_i.. After receiving m2 from P_j, U_i computes u =h (Z||ID_j ||n1) and performs the next step. U_i verifies the signature v by checking the equivalency of v^e_j mod N?=(u||h(u))mod N_j . Otherwise, U_i informs P_j that someone has tampered with Z and aborts the protocol. Otherwise, U_i chooses a random number t to be his short-term private key and computes $w = g^t$ mod n. U_i calculates the parameter k_{ij} as $k_{ij} = Z^t$ mod n. U_i generates a random nonce n3 and calculates three parameters K_{ij}, x and y in accordance with the following equations: K_{ij} = h(ID_j ||k i_j), the session key, x = $S_i^{h(Kij\ ||w||n2)}$ mod N, y = E_{Kij} (IDi||n3||n2), where E(·) is a symmetric crypto system such as DES or AES. U_i sends m3 = {w,x,y} to P_j After receiving m3, P_j computes k_{ij} as $k_{ij} = w^k$ mod n. P_j can obtain the session key K_{ij} by computing K_{ij} =h(ID_j || k_{ij}). P_j uses K_{ij} to decrypt cipher text y and retrieves IDi, n3, and n2. If n2 is valid, Pj computes SIDi = (IDi||h(IDi)). P_j verifies the validity of the identity IDi by checking $SID_i^{h\ (Kij\ ||w||n2)}$ mod N ? = x^e mod N. If the equation holds, P_j trusts that U_i is a legal user. P_j computes V = h(n3) and sends m4 = {V } to Ui. After receiving m4 from P_j, U_i computes V ||= h(n3) and confirms that V? = V '. When both the equations are same, U_i trusts that P_j is an authorized service provider and P_j has really calculated the common session key K_{ij}.

C. ***Encryption and Decryption Phase:*** Encryption and Decryption between user and provider is ensured using AES algorithm which is more secure than DES and there are currently no known non-brute-force attacks against AES. Data which is send from each provider to user is encrypted and send to the user, then the user decrypts it and the original data is retrieved. All these encryption and decryption are done using the more secure Advanced Encryption Algorithm (AES). The implementation is done using socket programming in Java and it uses server programs and client programs. To run in different machines, programming is based on IP address of the systems. Using the multithreading features of Java, all the providers can be run in parallel. The overall checking of authentication of user and provider are explained in fig.1.

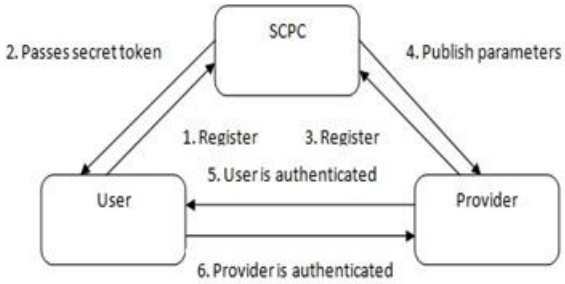

Figure 1: Checking authentication of user and provider

III.1. ADVANTAGES OF SSO

- Users need only one password for access to all applications and systems.
- Users can access the corporate network at the start of their workday.
- Users have immediately have access to all necessary password-protected applications.
- Users don't need to remember multiple passwords.
- Users don’t have to write down their passwords.
- Users don't have to guess passwords, which potentially expose applications to unauthorized users.

For example, Google Accounts allows a user to sign on to different services provided by Google using the same username/password pair. Another famous example is RSA SecurID [14], which a two factor authentication solution based on a OTP token and classical username/password credentials, allowing a user to sign on to several SecurID enabled services using the same token. However, a recent attack to EMC facilities exposed the overall fragility of this heuristic system. Even though their security was unaffected by current attacks, both solutions still require the user to repeatedly perform the sign on procedure. In most of current transparent single sign-on architectures, the user receives some kind of "authentication ticket" after he successfully signs on to the identity provider. When the user desires to sign on, he sends this ticket to the intended service provider or application, which then verifies it's validity by direct communication with the identity provider. This approach has several drawbacks, such as complex management and the requirement of secure online communication between applications and identity providers, which increases network traffic and processing loads.

IV. METHODOLOGY

In the existing system, different security schemes are proposed by many researchers. In the proposed system, various Client-Server programs are written to implement the project using socket programming in Java. This work uses the multithreading features of Java to run in parallel for different providers. Chang-Lee algorithm is used for user identification phase. But, it is using a less secure DES algorithm. This paper user a more secure AES algorithm to enhance the security features. So, this scheme is more secure than Chang-Lee scheme.

V. CONCLUSION

This paper proposes a secure single sign-on mechanism based on one-way hash functions and random nonces to solve the weaknesses described above and to decrease the overhead of the system. Encryption and Decryption of data sent between user and provider can improve security of communication. Encryption and Decryption process can be done using a more secure algorithm, ie, AES Encryption. AES is strong enough to be certified for use by the US govt. for top secret information. AES is federal information processing standard and there are currently no known non-brute-force attacks against AES. Thus AES is given priority than other standards when security is taken into consideration. By using this sso scheme, users need only one password for secure access to all applications and systems and would lock out the hackers entering into the system. But there are some vulnerability problems and there should be a good password, one that is very hard to crack.

This paper proposes further research into more efficient enhancements for security of single sign on for distributed computer networks. For third-party sites, credential generation and synced, cloud-based storage can be provided. Auto login, Smart cards, Biometrics are other methods to enhance security for single sign on mechanism for distributed computer networks.

REFERENCES

[1]. Weaver and M. W. Condtry, "*Distributing Internet services to the network's edge*", *IEEE Trans. Ind. Electron.*, 50(3): 404-411, Jun. 2003.
[2]. L. Barolli and F. Xhafa, "*JXTA-OVERLAY: A P2P platform for distributed, collaborative and ubiquitous computing*", *IEEE Trans. Ind.Electron*., *58(6):* 2163-2172, Oct. 2010.
[3]. L. Lamport, "*Password authentication with insecure communication"*,*Commun. ACM, 24(11*): 770-772, Nov. 1981.
[4]. Chin-Chen Chang, "*A secure single mechanism for distributed computer networks*," *IEEE Trans. On Industrial Electronics* ,*vol. 59, no. 1,* Jan 2012.
[5]. W. B. Lee and C. C. Chang, "*User identification and key distribution maintaining anonymity for distributed computer networks*," Computer Systems Science and Engineering, *15(4*): 113-116, 2000.
[6]. W. Juang, S. Chen, and H. Liaw, *Robust and efficient password authenticated key agreement using smart cards*, *IEEE Trans. Ind. Electron*.,*15(6):* 2551-2556, Jun. 2008.
[7]. X. Li, W. Qiu, D. Zheng, K. Chen, and J. Li, "*Anonymity enhancement on robust and efficient password-authenticated key agreement using smart cards*," *IEEE Trans. Ind. Electron., 57(2):* 793-800, Feb. 2010.
[8]. T.-S. Wu and C.-L. Hsu, "*Efficient user identification scheme with key distribution preserving anonymity for distributed computer networks,"* Computers and Security, *23(2*): 120-125, 2004.
[9]. Y. Yang, S. Wang, F. Bao, J. Wang, and R. H. Deng, *"New efficient user identification and key distribution scheme providing enhanced security,"* Computers and Security, *23(8):* 697-704, 2004.
[10]. K. V. Mangipudi and R. S. Katti, "*A secure identification and key agreement protocol with user anonymity (sika),"* Computers and Security, *25(6):* 420-425, 2006.
[11]. C.-L. Hsu and Y.-H. Chuang, *"A novel user identification scheme with key distribution preserving user anonymity for distributed computer networks*," Inf. Sci., *179(4):* 422-429, 2009.
[12]. Data Encryption Standard, NIST Std. FIPS PUB 46-2, 1988.
[13]. Advanced Encryption Standard, NIST Std. FIPS PUB 197, 2001.
[14]. W. Stallings, *Cryptography and Network Security*, 4th ed. Upper Saddle River, NJ: Pearson, Nov. 2005, pp. 334–340.