Single Sign On Mechanism Implementation Using Java

Mr. Kurhe B. S.
Second Year Master of Engineering, Department of Computer Engineering
Sharadchandra Pawar College of Engineering, Dumbarwadi, Otur, Pune, India
Email- b.kurhe@gmail.com

Prof. Deokate Gajanan S.
Assistant Professor, Department of Computer Engineering
Sharadchandra Pawar College of Engineering, Dumbarwadi, Otur, Pune, India
Email- deokate.gd@gmail.com

Abstract :- Credential of the technology is missing in the recent system. So, to improve the authentication of the distributed networks or multiple applications we are making use of a single sign on mechanism. Previously, users were using n number of usernames and passwords to access different applications on network. This would lead to higher expense for the administrator as each and every account of the organization will be handled with their particular username and password. But, now user needs to remember just single secure credential to access the multiple service providers in a distributed network by using single sign-on mechanism. However, most existing systems which are using this mechanism have some disadvantage regarding security requirements. So through this paper we will discuss about the development of security from earlier stage to present stage. And also discuss about the structure of the mechanism’s working strategy.

Keywords: Single sign-on, Distributed system and Privacy

I. INTRODUCTION
With the widespread use of distributed pc networks, it has become common to permit users to access numerous network services offered by distributed service providers. Consequently, user authentication (also called user identification) plays an important role to verify if a user is legal in distributed computer networks and can so be granted access to the services requested. To avoid phony servers, users sometimes ought to attest service providers. Once mutual authentication, a session key is also negotiated to keep the
confidentiality of the data exchanged between a user and a service provider. In many scenarios, the anonymity of legal users must be secured as well. On the other side, it is usually unpractical by asking one user to maintain distinct pairs of identity and password for different providers, since this could increase the load of both users and service providers as well as the communication overhead of networks. To avoid this problem, the single sign-on (SSO) mechanism has been introduced so that, after obtaining a credential from a trusted authority for a short period, each user’s authentication person can use this single credential to complete authentication on behalf of the user and then access multiple service providers. An SSO scheme should meet at least three basic security requirements, i.e., enforceability, privacy. In this paper we have demonstrated the implementation part of the system.

II. IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it’s constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

SINGLE SIGN-ON:

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 a distributed computer network. The single sign-on (SSO) mechanism has been introduced so that, after obtaining a credential from a trusted authority for a short period, each legal user’s authentication agent can use this single credential to complete authentication on behalf of the user and then access multiple service providers. Intuitively, an SSO scheme should meet at least three basic security requirements, i.e., unforgeability, credential privacy, and soundness. Unforgeability demands that, except the trusted authority, even a collusion of users and service providers are not able to forge a valid credential for a new user. 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 to other service providers. Soundness means that an unregistered user without a credential should not be able to access the services offered by service providers.

CREDENTIAL RECOVERING ATTACK:

In this attack, a malicious service provider who has communicated with a legal user twice can successfully recover the user’s credential. Then, the malicious service provider can impersonate the user to access resources and services provided by other service providers.

IMPERSONATION ATTACK:

In this attack may enable an outside attacker without any valid credential to impersonate a legal user or even a nonexistent user to have free access to the services.

SMART CARD PRODUCING CENTER:

In their scheme, RSA cryptosystems are used to initialize a trusted authority, called an SCPC (smart card producing center), and service providers, denoted as ‘s. The Diffie–Hellman key exchange technique is employed to establish session keys. In the Chang–Lee scheme, each user applies a credential from the trusted authority SCPC, who signs an RSA signature for the user’s hashed identity. After that, uses a kind of knowledge proof to show that he/she is in possession of the valid credential without revealing his/her identity to eavesdroppers. Actually, this is the core idea of user authentication in their scheme and also the reason why their scheme fails to achieve secure authentication as we shall show shortly.

III. DYNAMIC ID BASED ENCRYPTION AND HASHING ALGORITHM

Steps for data authentication
Step1: sender encrypts message using receiver public key
Step2: when receiver receives message from sender, receiver request a private key from key server
Step3: the key server sends an investigating message to sender, for receiver authentication
Step4: after getting the verification message from sender, the key generator provides a private key to receiver for decryption any time.

Steps for node authentication
Step 1: User u generates hash id using \( H(n) = \text{PUB\_KEY/IDENTITY} \)
Step 2: Neighbors node also generates hash id in the same way
Step 3:

{ If (hash_id (user) = hash_id(provider))
Then node is authenticated }
Else{ Node is malicious node }

Modules description

Wireless network setting
This module is developed to node creation and more than 30-50 nodes placed particular distance. Wireless nodes placed intermediate area in a distributed network. Each node knows its location relative to the sink. The maximum dimension of node is set as x=4000 and y=4000. The size of the nodes is set as 35 and the ultimate time simulation is 10 ms. The speed of network nodes set as 10-15 m/s.

Registration phase
In this phase, upon receiving a register request from network nodes, SCPC – smart card producing center provides a fixed length unique id to all network nodes for identification process, SCPC generates an id using RSA signature algorithm which provides a necessary public key and signature to network nodes for its identification purpose. The network nodes communicate with other nodes using this generated id.

Data routing phase
The user and provider establish a communication through multi hop path through the shortest path, the path establishment or path discovery is done through CBR (credit based routing), in the identified path the data is routed from several users through SSO – single sign on mechanism of authenticated user.

Authentication phase
In this phase, RSA-VES is employed to authenticate a user, while a normal signature is used for service provider authentication. Consider an adversary node try to inject false packets to provider in order to confuse the provider about original data packets and also tries to receive the original data packets from user. The RSA –VES algorithm enhanced to authenticate the original packets, the private key is used by the provider to decrypt the original packets.

Malicious node detection and legitimate node identification phase
For the proposed malicious node detection process digital signature dynamic source configuration routing is enhanced, which states KEY SERVER tends to verify the authentication of provider. By this proposed technique any number of users, intermediate nodes and provider can verify each other through dynamic hash function technique. In hash function technique if user wants to verify the provider the USER U generates a hash id through hash function \(H(n) = \text{PUB_KEY/IDENTITY}\), the public key and id of user U generates hash id. In the same way the PROVIDERS generate the hash id, if the user u hash id and provider hash id are same then the nodes are authenticated for data transmission and authentication.

A. User Identification Phase

![User Identification Phase](image)

This method check RSA signature using DH key. To access the resources of service provider, user needs to go through the authentication protocol. A symmetric key encryption scheme which is used to protect the confidentiality of user’s identity. Suppose a service request message from user, service provider generates and return user message which is made up by RSA on Signature Once this signature is validated, it means that user has authenticated service provider successfully. If he receives any message service provider can confirm validity by checking. After that the user generates the key temporarily. Once u close the process the same key does not work automatically your session are stopped [7].

© 2015, IJCSMC All Rights Reserved
This method checks RSA signature using DH key. To access the resources of service provider, user needs to go through the authentication protocol. A symmetric key encryption scheme is used to protect the confidentiality of user’s identity. Suppose a service request message from user, service provider generates and return user message which is made up by RSA on Signature. Once this signature is validated, it means that user has authenticated service provider successfully. If he receives any message service provider can confirm validity by checking. After that the user generates the key temporarily. Once you close the process the same key does not work automatically your session are stopped [7].

A. System Initialization Phase The trusted authority SCP C first selects two large safe primes p and q, and then sets N = pq. After that, SCP C determines its RSA key pair (e, d) such that ed = 1 mod φ(N), where φ(N) = (p − 1)(q − 1). Furthermore, SCP C chooses a generator g ∈ Z∗ n, where n is also a large prime number. Finally, SCP C publishes (e, g, n, N), keeps d as a secret, and erases (p, q) immediately once this phase has completed.

B. Registration Phase In this phase, each user Ui chooses a unique identity IDi with a fixed bit-length, and sends it to SCP C. After that, SCP C will returns Ui the credential Si = (IDi ||h(IDi))d mod N, where || denotes a concatenation of two binary strings and h(·) is a collision-resistant cryptographic one-way hash function. Here, both IDi and Si should be transferred via a secure channel. At the same time, each service provider Pj with identity IDj should maintain its own RSA public parameters (ej , Nj ) and private key dj as does by SCP C.

C. User Identification Phase To access the resources of a service provider Pj , a user Ui needs to go through authentication protocol specified. Here, k and t are random integers chosen by Pj and Ui respectively; n1, n2 and n3 are three random nonces; and E(·) denotes a symmetric key encryption scheme which is used to protect the confidentiality of user Ui’s identity IDi. We highlight this phase as follows. Upon receiving a service request message m1 from a user Ui, service provider Pj generates and returns the user message m2 which mainly includes its RSA signature on (Z, IDj , n1). Once this signature is validated, it means that user Ui has authenticated service provider Pj successfully. Here, Z = g k mod n is the temporal Diffie-Hellman (DH) key exchange material issued by Pj. After that, user Ui correspondingly generates his/her temporal DH key exchange material w = g t mod n and issues a proof x = S h(Kij ||w||n2) i, where Kij = h(IDi ||kij ) is the derived session key and kij = Z t mod n = w k mod
\[ n = g^k t \mod n \] is the raw key obtained by using the DH key exchange technique. • Proof \( x = \text{SID}(K_{ij} || w || n^2) \) is used to convince \( P_j \) that \( U_i \) does hold a valid credential \( S_i \) without revealing the value of \( S_i \). Namely, after receiving message \( m_3 \) service provider \( P_j \) can confirm \( x \)'s validity by checking if \( \text{SID}(K_{ij} || w || n^2) \) \( \mod N = x \) \( \mod N \), where \( \text{SID}_i = (ID_i || h(ID_i)) \). Once this quality holds, it means that user \( U_i \) has been authenticated successfully by service provider \( P_j \). Moreover, note that proof \( x \) is designed in a particular way so that except \( P_j \) and \( U_i \), anyone else cannot verify it as both \( U_i \)'s identity \( ID_i \) and the newly established session key \( K_{ij} \) are used to produce \( x \). This aims to achieve user anonymity as no eavesdropper can learn the values of \( ID_i \) and \( K_{ij} \). • Finally, message \( m_4 \) (i.e. \( h(n^3) \)) is employed to show that \( P_j \) has obtained message \( m_3 \) correctly, which implies the success of mutual authentication and session key establishment.

**Sample Screenshot Of Execution Of Single Sign On System:**

![Sample Screenshot Of Execution Of Single Sign On System](image-url)
IV. CONCLUSION

This paper addresses the limitations of Chang and Lee’s single sign-on (SSO) scheme against two types of attacks and Chang–Lee scheme to achieve soundness and credential privacy. This paper gives the implementations of Single Sign on System using Java.

REFERENCES


[9] Xiangxue Li,et.al.," Anonymity Enhancement on Robust and Efficient Password-Authenticated Key Agreement Using Smart Cards", *IEEE TRANSACTIONS FEBRUARY* 2010.

