Single Sign-on Mechanism for Secure Web Service Access through ISSO

Ramamurthi Deeptha, and Rajeswari Mukesh

***Abstract*—Single sign-on (SSO) is an emerging and more secure authentication mechanism that enables an authorized user with a single username/password to be authenticated by many service providers in a distributed network system. The existing technique used SSO scheme and it has achieved security by applying well-organized security parameters and its improved scheme introduced Verifiable Encryption of Signatures (RSA-VES). But the improvement of both the techniques with respect to security is not fully accomplished. We identified two attacks in existing SSO techniques. The first attack permits a malicious service provider to successfully communicate with a legal user more than one time and to recover the authenticated username/password and then to impersonate the service consumer to grant access to web resources and web services provided by other SP (Service Provider). Another attack is that a third party without any security credential may be able to access network services easily by impersonating some legal user or a fictional user. In our proposed work we introduced Improved Single sign-on (ISSO) scheme, which prevents Credential recovery attack, Impersonation attack and Data injection attack. We used the modified version of JMeter open source tool for generating the test report of the particular web apps. We implemented three web applications which provide financial solutions to customers. These three web applications used SOAP based request and response mapping for efficient handling of communication protocols. The testing result stated that the ISSO scheme fights against the attacks that were present in current SSO scheme.**

***Index Terms*—ISSO, Web Services, SOAP, Data Security, Secure Data Transfer, Josso, Distributed Network**

I. INTRODUCTION

Single sign-on (SSO) scheme is a one of access control technique of several associated, but self-determining software techniques [1]. With this property a user logs in to the system and is granted all access to service without any prompt to sign in again in each of them. Equally, Single Sign-off is the method for single action of signing out of all the user session and terminates access to multiple software resources. User authentication plays a crucial role in distributed computer networks to verify if a user is legal and can therefore be granted access to the services requested.

Features of using SSO are as follows:

- Narrowing password exhaustion from different user name/password combinations
- Decreasing time spent by re-typing passwords for the same credentials
- Reducing development and maintenance costs due to lower number of infrastructure help desk calls about passwords

SSO shares combined authentication service [2], which all web applications and system services use for authentication determinations and merging this with techniques to verify that users do not have to dynamically enter their credentials into more than once [3]. As Single Sign-on offers access to resources once the user is primarily authenticated it produces the negative fact in case of the credential details being accessible to other individuals. As a result, single sign-on needs an improved focus on the protection of user authorizations, and should preferably be joined with robust authentication methods like smart cards and OTP (One Time Password) tokens [4]. To tackle this situation we proposed multiple web service authentications with JOSSO for efficient authentication mechanism [5]. It provides security certificate to client machines for identifying whether the user is legitimate or not.

SSO also creates the secure authentication systems into extremely critical structure; a loss of their accessibility can lead to denial of service to all security systems under the SSO [6]. Single Sign-on can be unpredictable for systems to which access must be granted at all periods, such as security systems.

Every authorized user authentication provider agent will use this single credential to finish authentication on behalf of its user and then grants access to multiple SP. Instinctively, an SSO system must satisfy at least three basic security requirements, i.e., password privacy, enforceability, and secure authentication of web service consumer [7]. These requirements if not handled properly gives increase to these corresponding attacks like hijack attack, credential recovery attack, impersonation attack and SQL query injection attack. Enforceability guarantees the trusted authority to access services; even knowledge of SP and users is not possible to breach a valid certificate for a new user [8][9]. Privacy assures that schemed unfair SP must not be able to completely rollback a user's credential keys and intrude the user to log in to some other SP [10]. The term soundness states that an unauthorized user without a username/password cannot be able to access the web services offered by SP [11].

Certificate Authority (CA): Certificate Authority is the person or organization that digitally signs a report with its private key [12]. In these applications, the public key digital certificate contains a report, including the user public key and

Manuscript received February 5, 2015; revised April 12, 2015.

Authors are with Hindustan University, Chennai-603103, Tamilnadu, India (E-mail: r.deeptha@gmail.com, rajeswarim@hindustanuniv.ac.in).

digital signature of the report [13]. The difference between the digital certificate and the existing public key certificate is that in a general digital certificate, the public information does not contain any user's public key [14].

SOAP Messages: SOAP can form the layer of a web services protocol, it provides a basic message passing framework upon which web services are built [15]. This XML-based protocol contains SOAP envelope which defines the message and information about how to process the messages; it is a set of encoding rules and procedure for expressing occurrences of application-oriented data and a convention for expressive procedure calls and reactions [16]. SOAP has certain characteristics like extensibility (Web Service-routing are among the extensions under expansion), impartiality (it can be used over any transport protocol such as HTTP, SMTP, TCP, or Java Messaging Service), and independent access (SOAP allows message communication to any programming model) [17].

SOAP messages can be sent to a web application that has web services enabled systems, such as big database, with the constraints required for searching a record. The web service will return an XML-formatted document with the resultant data, e.g., location information, data, features. Through the records being dispatched in a XML format given by web service consortium, it can then be integrated into a third-party web service for their presentation [18].

In upcoming chapters we discussed deeply about our proposed scheme and advantages over the existing scheme. Section II A describes the existing Chang-Lee scheme [19]. The RSA-VES scheme is discussed in section II B. These state the overview of the existing encryption and security mechanism that were used in earlier work. In section II C, architecture of the existing Java based Single Sign-On tool is discussed along with the advantages and disadvantages of this tool. In section III, the proposed work is discussed. This section focuses more about the proposed implementation and gave an importance of this scheme and disadvantages of existing Chang-Lee scheme. We have implemented the improved JMeter tool for testing our web application [20]. The proposed tool contains the prevention mechanism which denied those intrusions and gave protection against vulnerabilities.

II. EXISTING TECHNIQUE : CHANG-LEE SCHEME

A. Chang-Lee Scheme

Chang and Lee's single sign-on scheme is a remote user authentication scheme [1], supporting session key establishment and user secrecy. In this scheme, RSA cryptosystems are used to mark a trusted authority provider, named an authorization unit, and SP. The Diffie-Hellman key interchange technique is engaged to establish session keys. In the Chang–Lee scheme, all users must apply a credential from the trusted authority, which authorize an RSA digital signature for the user's credential identity. After that, it uses an identity proof to claim that particular user is in proprietorship of the valid credential without compromising user's identity to intruder, this is essential idea of user authentication in their formation and also the purpose of why existing scheme fails to accomplish more secure certification. On the other side, each preserves its own RSA key pair for doing server side authentication. But Chang-Lee scheme fails to prove that how security standards are implemented. This scheme focused only on data encryption and decryption.

Chang–Lee scheme is not a secure Single Sign-on scheme because it has two probable effective and real impersonation attacks. The first attack, the "credential recovery attack" negotiations the certificate privacy in the Chang–Lee scheme as a malicious SP is able to recover the username/password of an authorized customer [21]. The other attack is an impersonation attack deprived of credentials and it determines how an external attacker may be able to easily make use of properties and services obtainable by service providers, since the attacker can effectively impersonate a legitimate user short of holding a valid credential and thus disturb the prerequisite of reliability for an SSO scheme [22]. In real, these events may put both consumers and service providers at high threat.

Creating user account For SSO in Josso:
In josso-users.xml

```
<user>
<name>java</name>
<properties>
<property>
<name>user.name</name>
<value>Bob</value>
</property>
<property>
<name>user.lastName</name>
<value>William</value>
</property>
<property>
<name>user.registrationDate</name>
<value>11/02/2014</value>
</property>
<property>
<name>email</name>
<value>bob.a@sampledomain.com</value>
</property>
</properties>
<roles>Admin</roles>
</user>
```

Figure 1. JOSSO User Account Creation

B. RSA-VES Scheme

To overcome the drawback in the Chang-Lee *et al* scheme, this scheme propose an improvement by employing an RSA Verifiable Encryption Signatures (RSA-VES) [23], an efficient primitive introduced for realizing fair exchange of RSA signatures. This scheme comprises of a trusted party and user. The VES scheme is that the trusted party who has a key pair of signature signs a given data and then encrypts the signature into the trusted party's public key. This scheme gives the service provider the freedom to secure signature [24]. The remaining procedures are the same as in the Chang-Lee *et al* scheme. The

problem in RSA-VES is authentication soundness [25]. Our proposed approach will overcome these problems by applying ISSO technique.

C. *Existing Java Single Sign-on (JOSSO) Tool:*

JOSSO is an Open Source and Internet based Single Sign-on

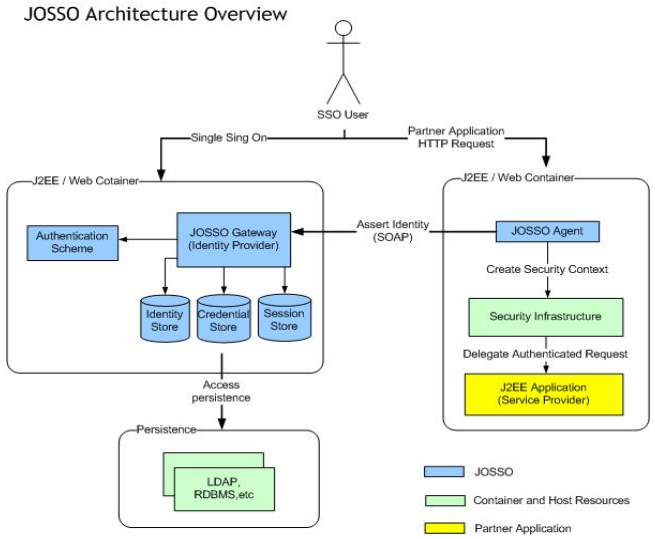

Figure 2. JOSSO - SSO Architecture

mechanism for quick and standards-based (XML) Internet-scale Single Sign-on application development tool [5]; this permits secure Internet access gateway to the Web applications or services of consumers, providers, and business associates and web services. Fig.2 shows the JOSSO-SSO architecture which has many features like J2EE, spring framework and transparent cross-platform and cross-group Single Sign-on scheme [26]. This architecture uses the persistence interface which is dealt with the database transactions and this is the thin client database server. This interface accepts inputs that are only coming from the web container and then it will be validated with the web container request and performs the database transactions. It also provides XML support for seamless Internet SSO capability and Apache Http 2.x server which enables transparent SSO with Ruby and Grails framework, JSP, Python, C#, Hadoop etc., It overcomes the security attacks of SSO by providing complete control of user's identity and the authentication of user is centralized [27]. Whenever the user signs in to the web application, then the session would be created and the users' cookies and session information would be maintained into the local file system. These stored cookies and session information are easily readable by any sniffer tools. So the JOSSO tool does not efficiently fight against the attacks which were discussed earlier. Hence a robust tool is needed to fight against those kinds of security threats. So we implemented our modified tool and discussed in the upcoming chapters.

III. PROPOSED SCHEME

In our proposed scheme we used customized open source library files for single sign-on authentication with Apache Tomcat server. The ISSO installation has two tasks, namely ISSO container configuration and ISSO agent configuration.

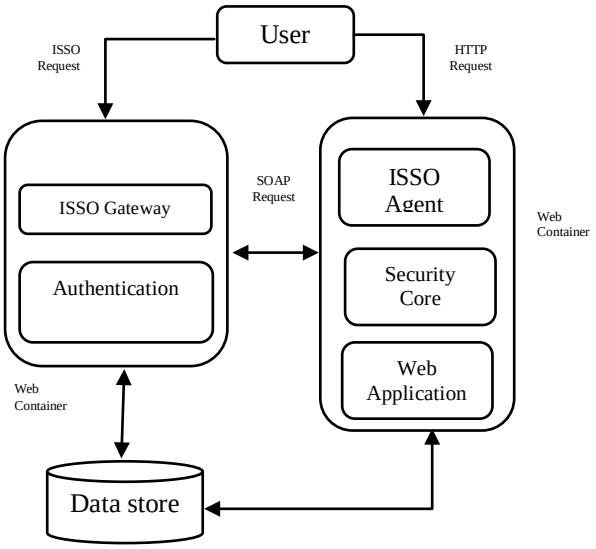

Figure 3. Proposed Architecture

Fig.3 shows the proposed architecture and the core blocks of the system are ISSO Agent, ISSO Gateway, Authentication block, Security block and Data store.

The information transaction between user and the ISSO web container are SOAP based requests. The ISSO agent every time sends requests to the ISSO gateway for consuming the web service. These requests are validated by the Authentication module and security module which performs the basic level authentication and encryption of the user's data. Finally the encrypted information is stored in users system for Single sign-on authentication. The data store collects the user request for processing the web service request and compares the user access level and responds to the client.

Our proposed work contains the following important steps:

Step1: Installing ISSO Gateway, this is for identity and credential data transfer to the unsecure medium.

Step2: Installing ISSO agent, this agent is responsible for executing web container and provides security infrastructure.

Step3: SOAP request and response creation for authenticating agent to gateway in order to provide authentication scheme.

Step4: Creating user account in ISSO configuration file, this is done through by editing *josso-users.xml* in root directory of JOSSO.

Step5: Determining password for created user, for this we need to edit *josso-credentials.xml*.

Step6: Creating *keystore* file for user information, this is an important step and this is done through the *keytool* command in windows, Figs 4 & 5 shows the *keystore* file creation and SSL socket creation.

Step7: Enabling 8443 port number for *https* protocol activation, server configuration manager in tomcat web container performs

this process.

Setup SSL and Container Configuration

```
>keytool -genkey -alias tomcat -keyalg
RSA

Enter keystore password:  changeit
What is your first name?
[Unknown:xxxx

What is the name of your organization?
[Unknown:cs144

What is the name of your organization?
[Unknown:HU

What is the name of your City?
[Unknown:CHENNAI

What is the name of your State?
[Unknown]:TN

What is the two-letter country code?
[Unknown]:IN

Is CN=localhost, OU=cs144, O=UCLA,
L=Los Angeles, ST=California, C=US
correct?
[no]:yes
```

Figure 4. Secure Socket Creation

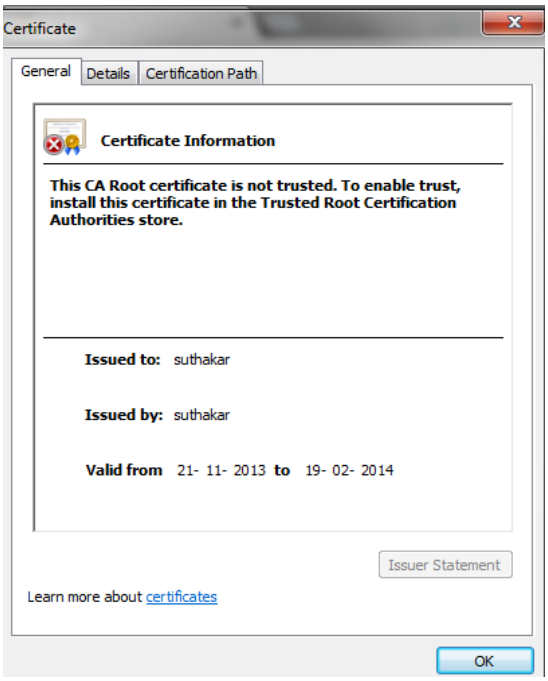

Figure 5. Digital Certificate for User

Fig.3 shows the Improved SSO based SSO architecture scheme. This entire architecture can be divided into following steps:

1. Gateway is acting as user authentication to the login page

2. Agent is active on the web as a user authentication system in conjunction with other web-based gateway.

a) Improved SSO Gateway:- This module will collect the information from the user request message and parse the required information by the XML parser. This module is available inside the web container. So this inherits the security features from the web container as well. The web container provides to host the web service and receives the request and response from the user. This gateway provides the improved XML responses that are received by the consumer service.

b) Authentication:- Authentication module provides the security implementations, which is an important part of the proposed technique. This uses the Secure Socket Layer(SSL) and secure http connection. This prevents the attacks against the http auto responder software (ex. Fiddler). This uses the efficient filtering scheme with data leak detection and malicious information collected from the unauthorized users. It will fight against the attack that comprises of the cookie attacks. Cookies will store the session and user information into the temporary internet files. So it leads to the attacker who can get credentials from different users. In this improved SSO scheme, it can defend against the cookie attack and prevent against the data loss. This module encrypts the data of the user by implementing the encryption methodologies like the RSA scheme. This is done by the windows based encryption tool namely *keytool* command which provides strong prevention against all types of attacks.

c) *ISSO Agent*:- This is the core module of our improved system. This will help in getting the request and response message from the user and extracts the session information and all the available information of the particular session user that are comprised into the session. This will interact with the user and web container whenever the user sends the request and response message. This will help to convert the session information into the API response format. The communication between the agent and the web container of the gateway will be interconnected with the user session information. Web service requests are rendered and populated in this module only. This web service will comprise the session management, logging, tracking of the user session and monitoring. If user is positively authenticated they are allocated an effective session. This session covers a quantity of characteristics and things that describe the user's individuality and some time-dependent activities.

d) Security Core:- This module will help to prevent unauthorized access and prevent against the malicious users in SOAP request/response handler and transferring data between the web containers.

Steps for implementing the ISSO

a) Client issues the first request in the session information (no user/password)
b) Server reads this and responses like, 'I don't know who you are, you must bring me some identity information for authentication' (technically it sends some header)
c) When Client Browser gets such a response, it issues a connection to the server (it includes apache server

implementation) and obtains some identification information from the user.

d) Browser issues a request to the web server which comprises of a ticket to the web server.

e) Web server reads and analyzes this ticket, talks by itself to the server and ensures that the ticket is validated and corrected.

f) Web Server allows the client to connect; the username is accessible from the client request, and it can store it on HttpSession, if service provider wants.

Pseudo Code for ISSO implementation

a) Initializing ISSO module:

```
status = sso_init(properties);
if (status != SUCCESS) {
printf("ISSO_init failed.\n");
return 1;
 }
```

Properties – parameters for initializing the ISSO module, this contains username, password, client session id, cookie information.

Status – Boolean value, either true or false.

b) Handling incoming request from different providers:

```
if (isso_token_handle != NULL) {
const char *isso_token_id = get_sso_token_id(isso_handle);
boolean_t is_valid =
am_isso_is_valid_token(isso_token_handle);
print ("session state is %s.\n",
is_valid == TRUE ? "valid":"invalid");
}
```

Where,

isso_token_handle – incoming request from user

isso_token_id- session id

isso_handle- user id information

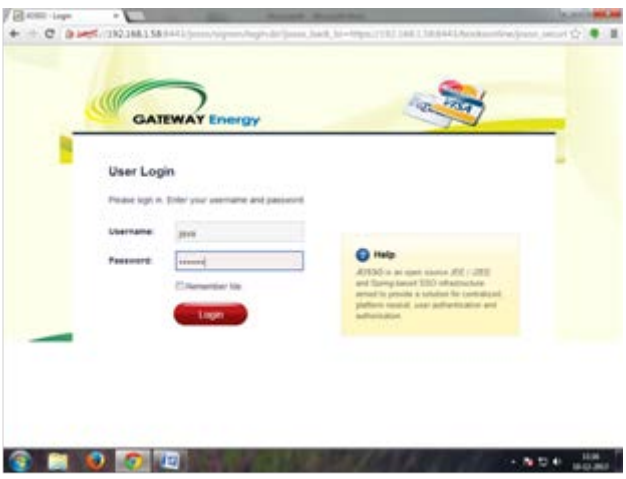

Figure 6. SSO Sample Login

is_valid- true if authenticated user. Else it set to malicious and denies the access requested.

We created sample user account by modifying the *josso-users.xml* located in the tomcat server. This process is done through *keytool* command available in Windows operating system.

Digital Signature Creation: Digital signatures are a type of asymmetric cryptography. For data sent through a non-secure channel, an accurately implemented digital signature gives the receiver the reason to believe that the message was sent by the sender. Digital signatures certificate are alike to hand written signatures in numerous salutations, but accurately employed digital signatures system are more problematic to forge than the handwritten type data. The digital signature security system is cryptography based and need be executed appropriately to be active. Digital signature system can also distribute non repudiation and meaning that the signer cannot positively claim that they didn't sign a private data; however their private key remains hidden; moreover, several non-repudiation methods compromise a time stamp on behalf of the digital signature certificate, with the purpose of even if one private key is exposed then the signature is valid. Examples contain electronic mail system, contracts, agreements, or a message sent via some other cryptographic protocol.

Fig.5 clearly shows our newly created digital certificate. This is the license given by the trusted authority. While executing *keytool* command in system, it will create user confidential information into *keystore.* Fig.6 showed the sample output for proposed implementation. This *keystore* contains private key, user account information of the particular system and other user identification details.

IV. Discussion

TABLE I
COMPARISON BETWEEN THE ATTACKS IN EXISTING AND PROPOSED TECHNIQUES.

Attack Type	RSA-VES Scheme	ISSO Scheme
Impersonation Attack	Yes	No
SQL Injection	No	No
Credential Recovery	Yes	No
Zero Day Attack	No	No

In Table.1, we have compared the ISSO scheme with existing implementation and consolidated the attack detection rate in both of the implementations. The impersonation attack is possible in RSA-VES scheme and Chang-Lee scheme. In ISSO scheme we implemented the character based POST message request for avoiding this type of intrusion. The credential recovery attack is also possible in the existing schemes, because they used the simple encryption technique which can easily be broken by the attacker.

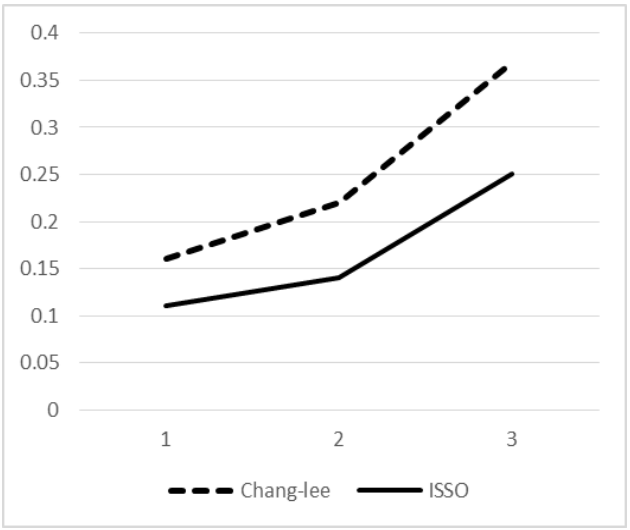

Figure 7. Time (in msec) Consumption for authentication in Chang-lee and ISSO (proposed) schemes.

The proposed implementation uses the XML configuration and security protocol for preventing against these threats. Fig.7 clearly shows the time consumption rate for authentication and detects the different attacks which were mentioned in the earlier chapters. The new mechanism takes a maximum of 0.25 msec for testing the request and response message sent by the client or service provider.

V. SECURITY ANALYSIS

ISSO users can use this service from LAN, WLAN, Mobile Devices, kiosk, etc. as in our proposed technique we had developed standard UI design for all the devices. Fig.8 shows the prevention rate of the ISSO and the Chang-Lee schemes.

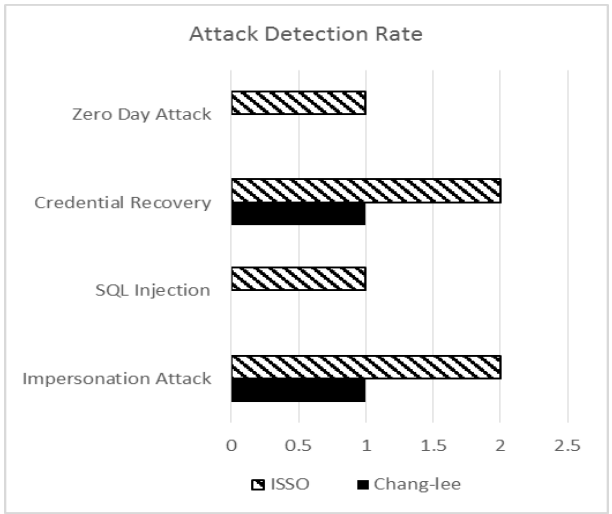

Figure 8. Security comparison between ISSO and Chang-Lee schemes

We applied SSL and communication port number 8443. This port number is for https protocol authentication. Encryption algorithm provides confidentiality in user's credentials. Comparing with existing technique they focused on secure communication for desktop computers; this approach fights against all type of intrusions and denial of service attack.

VI. IMPLEMENTATION

Fig.6 shows our implementation of ISSO based web application that was developed in J2EE. This prototypic implementation used Apache Tomcat 6.0, JDK1.7, SOAP web service and operating system is Windows 7 Ultimate. Fig.6 shows the sample login page created by JOSSO. We applied various parameters on this model. Fig.5 shows our new digital signature certificate issued by *keytool* command. This is a temporary license given by system. This will authenticate SSL (Secure Socket Layer) socket and HTTPS protocol for secure authentication mechanism. These techniques fight against the attacks mentioned in Chang-Lee scheme and RSA-VES scheme.

VII. CONCLUSION AND FUTURE WORK

In our implementation, we demonstrated improved version of Chang and Lee's Single Sign-on (SSO) scheme and RSA-VES scheme. Most existing single sign-on schemes suffer from various security issues and are vulnerable to different attacks. In this proposed scheme, we formalized authentic key interchange single sign-on scheme using ISSO. Specially, we properly define secure authentication for both users and service sources. This scheme will protect against a malicious Service Provider which can impersonate a legal user in order to enjoy the resources and services from other web service provider. Further research will focus on simplifying the digital signature creation and container configuration difficulty. Another enhancement is on the report generating tool. Here we can use the user interface for collecting the log information from the user's session information. From the session logs and server logging dataset, we can mine further the log information and from that log we could take decision for the testing report. This report page may include the way of communication between the web service provider, gateway and customer.

REFERENCES

[1] Guilin Wang, Jiangshan Yu,"Security Analysis Of A Single Sign-On Mechanism For Distributed Computer Networks," IEEE Trans on Industrial Informatics, Vol. 9, No. 1, pp. 294-302 February 2013.

[2] W. Lee and C. Chang, "User Identification And Key Distribution Maintaining Anonymity For Distributed Computer Networks," Comp.Sys. Sci. Eng., vol. 15, no. 4, pp. 113–116, year 2000.

[3] W. Juang, S. Chen, "Robust And Efficient Password Authenticated Key Agreement Using Smart Cards," : IEEE Transaction on Electronics., volume-15, number.6, pp.2551–2556, June-2008.

[4] X. Li,W. Qiu, D. Zheng, Chen, "Anonymity Enhancement On Robust And Efficient Password-Authenticated Key Agreement Using Smart Cards" : Transaction on . Industry Electronics - IEEE., vol. 57, no. 2, pp. 793–800, Feb. 2010.

[5] The Open Group, "Security Forumon Single Sign-On". http:// www. opengroup.org .

[6] SSO Security http:// www.opengroup.org/security/l2-sso.htm.

[7] http://www.josso.org/confluence/display/JOSSO1/Setup+JOSSO+Agent+-+Apache+2.2.

[8] http://docs.oracle.com/javase/6/docs/technotes/tools/windows/keytool. html , "Windows Key Tool".

[9] L. Harn and J. Ren, "Generalized digital certificate for user authentication and key establishment for secure communications," IEEE Trans. Wireless Commun., vol. 10, no. 7, pp. 2372–2379, Jul. 2011.

[10] C.-C. Chang and C.-Y. Lee, "A secure single sign-on mechanism fordistributed computer networks," IEEE Trans. Ind. Electron., vol. 59, no. 1, pp. 629–637, Jan. 2012.

[11] U. Feige, A. Fiat, and A. Shamir, "Zero-knowledge proofs of identity,"

J. Crytography, vol. 1, no. 2, pp. 77–94, 1988. [21] G. Ateniese, "Verifiable encryption of digital signatures and applications," ACM Trans. Inf. Syst. Secur., vol. 7, no. 1, pp. 1–20, 2004.

[12] H. Delfs and H. Knebl, Introduction to Cryptography: Principles and Applications, 2nd ed. Berlin, Germany: Springer, 2006.

[13] G. Tenenbaum, Introduction to Analytic and Probabilistic Number Theory. Cambridge, U.K.: Cambridge Univ., 1995, p. 41.

[14] E. W. Weisstein, "Relatively Prime," MathWorld-A Wolfram Web Resource [Online]. Available: http://mathworld.wolfram.com/RelativelyPrime.html

[15] Public Key Cryptography Standards, PKCS #1 v2.1, RSA Cryptography Standard, Draft 2, PKCS, 2001 [Online]. Available: http://www.rsasecurity.com/rsalabs/pkcs/

[16] D. Boneh, "Twenty years of attacks on the RSA cryptosystem," Notices Amer. Math. Soc., vol. 46, no. 2, pp. 203–213, 1999.

[17] Wikipedia, RSA (algorithm) [Online]. Available: http://en.wikipedia.org/wiki/RSA_(algorithm)

[18] Y. Xu, R. Song, L. Korba, L.Wang, W. Shen, and S. Y. T. Lang, "Distributed device networks with security constraints," IEEE Trans. Ind. Inf., vol. 1, no. 4, pp. 217–225, Nov. 2005.

[19] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," ACM Trans. Comput. Syst., vol. 8, no. 1, pp. 18–36, 1990.

[20] M. Bellare and P. Rogaway, "Entity authentication and key distribution," in Proc. of CRYPTO', 1993, pp. 232–249.

[21] C. Boyd and W. Mao, "On a limitation of BAN Logic," in Proc. Of EUROCRYPT, 1994, pp. 240–247.

[22] N. Asokan, V. Shoup, and M. Waidner, "Optimistic fair exchange of digital signatures," IEEE J. Sel. Areas Commun., vol. 18, no. 4, pp. 591–606, Apr. 2000.

[23] J. Camenisch and M. Michels, "Confirmer signature schemes secure against adaptive adversaries," in Proc. EUROCRYPT, 2000, pp. 243–258.

[24] G. Ateniese, J. Camenisch, M. Joye, and G. Tsudik, "A practical and provably secure coalition-resistant group signature scheme," in Proc. CRYPTO, 2000, pp. 255–270.

[25] D. Chaum and T. P. Pedersen, "Wallet databases with observers," in Proc. of CRYPTO, 1993, pp. 89–105.

[26] G. Wang, J. Yu, and Q. Xie, Security analysis of a single sign-on mechanism for distributed computer networks Cryptology ePrint Archive, Rep. 102, Feb. 2012 [Online]. Available: http://eprint.iacr.org/2012/107

[27] J. Yu, G.Wang, and Y.Mu, "Provably secure single sign-on scheme in distributed systems and networks," in Proc. 11th IEEE TrustCom, Jun. 2012, pp. 271–278.

Ramamurthi Deeptha received the Bachelor of Technology and Master of Technology degrees in Information Technology from University of Madras and Sathyabama University, Chennai, Tamil Nadu, India, in 2004 and 2009, respectively. She was awarded with a college Gold medal and secured University fourth rank in UG and PG degrees respectively.

She is currently pursuing research in Hindustan University and her area is Security testing of SSO of Web Services. She has more than 7 years of teaching experience and her areas of interest include Web Services, Software Testing, Network Security and UML. She has got 6 publications in reputed conferences and journals.

She was awarded Dale Carnegie certificate for promoting teaching-learning process sponsored by Wipro Technologies. Also she has completed Mission 10X and Advanced Mission 10X programs organized by Wipro Technologies. Apart from this, she is a certified internal auditor and an IBM Web Sphere Studio Application Developer.

Rajeswari Mukesh is working as Professor in the Department of CSE of Hindustan University, Chennai. She has 20 years of experience in teaching and research. Her area of specialization is Network Security. She has got more than 25 publications in the reputed conferences and journals.