International Journal of Engineering Research in Computer Science and Engineering (IJERCSE) Vol 2, Issue 4, April 2015

Single Sign-On Mechanism Using RSA-VES

[1]Reeba Alexander,[2]R.Medona Selin
[1][2]Department of Computer Science and Engineering, Vins Christian College of Engineering,chunkankadai Anna University,Chennai

Abstract **Single Sign-On (SSO) mechanism is one of the latest authentication mechanisms in distributed computer network. This mechanism enables a valid user with single token to access services of multiple service providers in a network. Previously many SSO schemes are implemented. This paper proves that previous schemes are insecure as it fails to meet token privacy and soundness of authentication.Basically; here two impersonation attacks are present. The first attack allows the dishonest service provider who had communicated with valid user twice can easily recover user's token and impersonate user to access services of other service providers. In second attack, an external without any token can be able to enjoy the services freely by acting as valid user or nonexistent user. This paper proposes verifiable encryption of RSA signatures to overcome the flaws of previous SSO scheme.**

Keywords - **Encryption, network security, signature, single sign-on, token.**

I. INTRODUCTION

The mass usage of distributed system allows users to access various services given by multiple service providers [1].Authentication plays a key role in distributed computer systems [2] to verify if the user is valid one or not. If valid can grant access to services requested. To avoid malicious servers, authentication of service provider is needed. Once mutually authenticated, a session key is generated to keep the confidentiality of the data exchanged between user and service provider [3].In many cases, anonymity of valid users must be protected [4].It is a big challenge to design efficient and secure authentication protocol in complex computer network environments [5].In year 2000, [3] a scheme of user identification and key distribution was proposed to maintain user anonymity. Later this scheme was prove insecure against impersonation attack and identity disclosure attacks by second scheme [6]. In meanwhile third scheme [7] was implemented for the improvement of second scheme. But, in year 2006,fourth scheme[8] was implemented to prove that third scheme suffers from Denial of Service(DOS) .Later in year 2009 a new scheme[9] based on RSA was implemented to show that both third and fourth scheme was insecure under identity disclosure attack.It is usually difficult by asking one user to maintain different pairs of id and password [12] for multiple service providers, as it could provide burden to user and service provider [10] as well as increase the overhead of the networks. To avoid this problem, the single sign on mechanism [13] is introduced. Single sign-on (SSO) is a method of access control that enables a user to log in once and gain access to the resources of multiple service providers without being prompted to log in again. Single sign-off is the opposite mechanism whereby a single action of signing out terminates access to multiple service providers. An SSO scheme should meet three basic security requirements, i.e., unforgeability, *tokenprivacy, and soundness.*Unforgeability defines that, except the trusted party even a collusion of users and service providers are not able to forge a valid token for a new user. Token privacy [11] defines that colluded malicious service providers should not be able to fully recover the user's token and then act as user to access services of another service provider. Soundness defines that unregistered user without a token should not able to access services provided by service provider. A careful study of SSO scheme was made by a scheme [16].This scheme point out two weakness of another scheme.1) an external can forge a valid token by mounting token forging attack as the scheme employed RSA signature without using any hash function to issue a token for any random identity selected by user and 2) the scheme requires clock synchronization as it uses time stamp. The scheme was based on RSA which doesn't depend on clock synchornization.It rely on nonce instead of timestamp. The scheme has high efficiency in computation and communication. The scheme supports secure mutual authentication, session key agreement [15] and user anonymity. Another scheme [14], which has a generic SSO construction which depends on encryption and ZK proof. Compared to this scheme above scheme has many attracting features such as less underlying primitives, high efficiency and not require a PKI for users. But this SSO scheme was not secure.

In this paper, the above scheme [16] is insecure by presenting two impersonation attacks, i.e., token recovering attack and impersonation attack without token. In the first

attack, a dishonest service provider who has communicated with valid user twice can able to recover the token. With this token dishonest service provider can act as a user and access the resources and services of other service providers. The second attack enables an external without any valid token to act as a valid user or nonexistent user to have free access to the services. These two attacks prove that the scheme fails to meet token privacy and soundness, which are the basic requirements of SSO scheme and other authentication protocol. Lastly to avoid these two attacks, an improved SSO scheme is proposed to enhance the previous scheme. At the end,RSA-based verifiable encryption of signatures is discussed to introduce the fair exchange of signature. Later part of this paper is organised as follows. Section II gives summary of previous scheme. After that, in Section III two attacks are discussed. Then improved SSO scheme using VES is given in Section IV and at last conclusion is given in Section V.

II. SUMMARY OF THE PREVIOUS SCHEME

The previous scheme is the authentication scheme for user which supports session key establishment and user anonymity. In this scheme RSA cryptosystem is used to initiate the trusted party and service providers. The Diffie-Hellman key exchange technique is used to establish the session keys. In this scheme, each user uses a token from trusted party who signs RSA signatures for the user's hashed identity. After that user uses a proof to use valid token without revealing identity to the attackers. This is the key concept of this scheme.

On the other side, each service provider maintains their own RSA key pair for the authentication of server. This scheme comprises of three phases: initialization, registration and user identification. The detail of this scheme is as given below.

A. Initialization Phase

The trusted authority selects two large safe primes p and q and then computes N=p.q. Trusted party also determines the key pair (e, d) such that $e \cdot d \equiv 1 \bmod \varphi(N)$, where $\varphi(N) = (p-1)(q-1)$. Next, trusted party chooses a generator g over the finite field Z_n^*, where n is a large odd prime number. Finally, trusted party protects the secrecy of d and publishes (e, g, n, N).

B.Registration Phase

In this phase each user chooses a unique identity with fixed bit –length and sends to trusted party. After receiving this the trusted party provide token to the user. The token and identity are passed through a secure channel.

At the same time, each service provider maintains its own RSA public and private parameters as the trusted party.

C.Users Identification Phase

To access the resources of the service provider, user needs to go through the authentication protocol.The random integers a nonces and symmetric key are used to protect the confidentiality of user identity. The highlight of this phase is as follows.

- On receiving a service request from user service provider generates and returns user message which is made up by its RSA signature. Once this signature is validated, it defines that user has authenticated service provider successfully. In this Diffie-Hellman (DH) key exchange material is issued by service provider.
- After that, user correspondingly generates the Diffie-Hellman key exchange material and also a proof.
- Proof is used to convince service provider that user hold valid token without revealing the token.
- Finally, message is employed to show that service provider has get correct message, it implies the success of mutual authentication and session key establishment.

III. ATTACKS AGAINST THE PREVIOUS SCHEME

As seen from the previous section, it seems that the above scheme achieves secure mutual authentication, as authentication of server is done by using traditional RSA signature issued by service provider. Without valid token it is difficult for an attacker to act as a valid user when pass through the user authentication procedure. The scheme is actually insecure SSO scheme as there are two impersonation attacks. The first attack is the token recovering attack which compromises the token privacy in the scheme as a dishonest service provider is able to recover the token of the valid user.

The second attack is an impersonation attack without token. In this attack an external attacker may be able to freely access the resources and services offered by service providers, as the attackers can easily act as a valid user without having a valid token and thus violate the soundness of SSO scheme. These attacks may leads user and service provider at high risk.

The description of both these attacks is as follows.

A. Token Recovering Attack

The schemes satisfy the requirement of token privacy as receiving token proof doesn't allow service provider to recover the token of the user. The difficulty of calculating value is the reason of RSA cryptosystem to be secure in other words it is not that much easy for an attacker to derive the RSA private key from public key. This is because here another RSA public/private key pair is used. It is difficult to recover token from proof as this is equivalent to decrypt the

RSA cipher text w.r.t to public key. There is a drawback in producing proof as here the same token is encrypted multiple times under different public keys w.r.t same RSA modulus N.

When dishonest service provider has use the scheme with the same user twice, service provider will able to easily recover the token at high probability by using another algorithm. Finally the attack lead to serious consequence as after recovering the valid token of a valid user dishonest service provider can act as user and can access all the resources and services offered by other service providers.

B.Impersonation Attack without Token

The soundness of the SSO scheme which is to satisfy security requirements. To get the valid proof is difficult.So, the attacker is not able to log in to any service provider if it doesn't know the trusted party RSA private key or user's token.

Again, however, such a plausible discussion simply explains the rationale of the SSO scheme but cannot guarantee its security w.r.t. the soundness. The attack is explained as follows:

1. To act as a valid user for accessing the services of service provider an attacker first send request message as valid user normally does.
2. On receiving the reply from the service provider the attacker checks the service provider signature and selects a random integer and computes some value. Before the next step the attacker checks the divisibility of the value. If not, then attacker has to choose another random integer or start a new session to fulfill the condition.
3. As value is divisible so calculate the proof.
4. At last, attacker can act as user to pass through authentication by sending message to the service provider.

There are a number of things worth noting in regard to the above impersonation attack without token. First, the attack will succeed at a rate of about 1/e for one random number in a new session. The reason is that value holds with a probability of about 1/e, since and the output of hash function h can be treated as random numbers. Consequently, if e=3 the above attack can succeed once by trying about three values of t on average. Even if e is as large as 65537, trying 65537 times to get a successful impersonation may not be difficult for attacker as it may explore a machine, which can be much more powerful than a mobile device, to do the computations needed for each try, i.e., two modular exponentiations and two hash evaluations. Moreover, even when timeout is introduced into the scheme it may be not a real obstacle for attacker as it can initialize new sessions (w.r.t. the same or different identities). Second, in the above attack we assume that e is a small integer and attacker may know the value of one valid user's identity. This is reasonable as explained below. On the one hand, in the initialization phase of the scheme only specifies that the trusted party needs to set its RSA key pair(e,d) but does not give any limitation on the length of public exponent .So,e could be a small integer with binary length less than the output length of hash function h ,

Moreover, this is to happen because: 1) to speed up the RSA signature verification, some security standards, academic papers and popular web sites suggest that e can be set as 3 or 65537; and 2) as the scheme is claimed to be efficient even for mobile devices in distributed networks, using small exponent e can provide further computational advantage for these devices as they usually have limited resources for computation and storage. In addition, the security analysis given in [16] neither excludes the case of small e nor relies on the concrete procedure of setting trusted party RSA key pair (e, d). On the other hand, in the SSO scheme users identities are not as important as the token, though the identities are transferred in cipher text to provide user anonymity. So, user's identities could be known by an attacker due to reasons, such as user's negligence. At least service providers know user's identities. Moreover, even if user's identities are well protected so that attacker cannot impersonate registered user as above, then attacker can freely forge an identity. This is possible because in this scheme, each user selects identity by following only one requirement: each identity is a string with fixed bit-length. Therefore, an external attacker can use an arbitrary such string as an identity to mount the above attack, as the service providers are not provided any additional mechanism to check whether identity has been registered with trusted party. This also implies that if e is a small integer, attacker can even act as a nonexistent user to make use of the resources and services offered by service providers.

Finally, it must be emphasized that impersonation attacks without valid token seriously violate the security of SSO schemes as it allows attackers to be successfully authenticated

without first obtaining a valid token from the trusted party after registration. In other words, it means that in an SSO scheme suffering these attacks there are alternatives which enable passing through authentication without token.

The security of the SSO scheme has been analyzed in three different ways: 1) BAN logic was used to show the correctness of the schemes; 2) informal security arguments were given to demonstrate that the scheme can resist some attacks, including impersonation attacks; and 3) a formal security proof was given to prove that their scheme is a secure authenticated key exchange (AKE) protocol. However, these security analyses and proofs still do not guarantee the full security of the scheme and there are a number of reasons for this. First, as early as the 1990s, it was known that although BAN logic had been shown useful to identify some attacks, it could approve protocols which are actually unsound in practice because of some technical

weaknesses in the logic. Finally, the formal proof about AKE only focuses on the session key security, i.e., an attacker with all reasonable resources is not able to know the session key established between the two parties under the computational DH (CDH) assumption, and not the security of mutual authentication. one fundamental requirement of a secure AKE protocol is that there be a secure mutual authentication.

IV. PROPOSED IMPROVEMENT

To overcome the drawbacks in the previous scheme [16], the scheme propose an improvement by employing an RSA-based verifiable encryption of signatures [18] (RSA-VES), which is an efficient primitive introduced in [20] for realizing fair exchange of RSA signatures. VES has three persons: a trusted party and two users, say A and B. The key idea of VES is that A who has a key pair of signature scheme signs a given message and encrypts the resulting signature under the trusted party's public key, and uses a noninteractive zero-knowledge (NZK) proofs[17] to convince B that she has signed the message and the trusted party can recover the signature from the cipher text. After validating the proof, B can send his signature for the same message to A. For the purpose of fair exchange, A should send her signature in plaintext back to B after accepting B's signature. If she refuses to do so, however, B can get her signature from the trusted party by providing A's encrypted [19] signature and his own signature, so that the trusted party can recover A's signature and sends it to B, meanwhile, forwards B's signature to A. Thus, fair exchange is achieved.

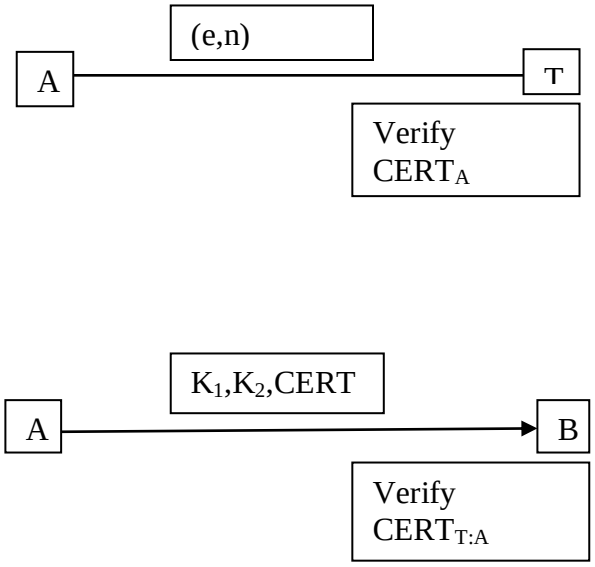

Fig 1. Verifiable Encryption of RSA Signature

The basic idea of the improved scheme can be highlighted as follows. User 's token is the trusted party's RSA signature on the square of the hashed user identity. This scheme comprise of basically three phase. The phases are the initialization phase, registration phase and authentication phase.

A.Initialization Phase

Trusted party select two large safe primes p and q to set N=pq where p=2p'+1 and q=2q'+1 and p', q' are two prime numbers. Trusted party sets its RSA public/private key pair (e, d) ed=1mod2p'q' where e is prime number. Trusted party picks generator g of Q_N, selects Elgamal decryption key u and compute public key $y=g^u$ mod n.For completing Diffie-Hellman key exchange trusted party chooses generator $\overline{g} \in Z_N$.The n is another large prime number. Trusted party chooses cryptographic hash function. At last trusted party publishes(e,N,h(.),€,g,y, $\overline{g}$,n) and keeps (d,u) secret.

B.Registration Phase

On register request the trusted party gives user U_i a fixed length unique id ID_i and issues credential $S_i=h(ID_i)^{2d}$ mod N.

Each service provider P_j with identity ID_j maintains a pair of signing/verifying keys for secure signature scheme. $\sigma_j(SK_j,msg)$ denotes the signature σ_j on message msg signed by P_j using signing key SK_j.Ver (PK_j, msg, σ_j) denotes verifying of signature σ_j with public key PK_j which outputs '0' or '1' to indicate if the signature is valid or invalid.

C.Authentication Phase

The authentication is for both user and service providers.

Service Provider Authentication:

U_i sends a service request with nonce n_1 (random number) to service provider P_j.On receiving (Req, n1), P_j calculates its session key material $Z=g^k$ mod n where $k \in Z_n$ is a random number.P_j sets u=Z concatenates service provider ID concatenates nonce value n1 and issues a signature $v=\sigma_j$ (SK_j, u) and sends m_2= (Z, v, n2) to user where n_2 is nonce selected by P_j.After receiving m_2= (Z, v, n2), user sets u= Z concatenates service provider ID concatenates nonce value n_1.User terminates the conversation if Ver (PK_j, u, v) =0.Otherwise user accepts service provider P_j because sign v is valid.In this case user U_i selects a random number $t \in Z_n$ to compute $w=g^t$ mod n,$k_{ij}=Z^t$ mod n and issues session key K_{ij}=h (ID_j concatenate k_{ij}).

User Authentication:

User first encrypts the credential S_i under public key y.User encrypts identity ID_i, new nonce n_3,P_j's nonce n_2 using session key K_{ij} to get cipher text CT.CT=$_{EKij}$(ID_i concatenates n_3 concatenates n_2).Sends $m_{3=}$ (w, x, CT) to service provider P_j.To verify user U_i,P_j calculates $k_{ij}=w^k$mod n session key $K_{ij}=h(ID_j$ concatenates $k_{ij})$.Then uses K_{ij} to decrypt CT and recover (ID_j,n_3,n_2).If output is negative,P_j aborts the conservation.Otherwise,P_j accepts user U_i and believes that they shared the same session key K_{ij} by sending user U_i $m_4=(V)$ where $V=h(n_3)$.After user U_i receives V,he checks if $V=h(n_3)$.If this is true,then user U_i believes that they have shared the same session key K_{ij}.Otherwise user U_i terminates the conversation.

V. EXPERIMENTAL RESULT

The implementation of the algorithm increases the efficiency of the single sign-on mechanism. It also provides the secure access of the services. It prevents the misuse of the credential by unauthorized users or service providers.It eliminates the flaws of the previous schemes.

CONCLUSION

The main purpose of implementing a RSA-VES algorithm is to provide security to single sign-on. The algorithm involves the token privacy and soundness. The current implementation focus on authentication mechanism. It can be also possible to implement on a real time system such as web services. It is interesting to formally define authentication soundness and construct efficient and provably secure single sign-on schemes. A new SSO scheme is efficient, safe and suitable for any networking infrastructure, for resource constrained. The analysis of SSO schemes that appear in the literature (including our new scheme) is done manually by observing traditional attacks (such as, the Reply Attack) and they can be mounted against a scheme run. In addition, the proofs given in most analyses are not based on sound logic. The automatic verification tools at the moment can handle concrete cryptographic protocols that have a narrower purpose that of SSO schemes. A possible future line of research would be to expand the functionality of current verification tools and logic analysers to contain SSO schemes. Choosing the right SSO solution for a specific organization is still a daunting and confusing task for most security professionals. To resolve the confusion we present the taxonomies of SSO solutions and their qualities, and by describing the architectures and operations of a selection of SSO solutions in use today.

REFERENCES

[1] A. C. Weaver and M. W. Condtry, "Distributing internet services to the network's edge," *IEEE Trans. Ind. Electron.*, vol. 50, no. 3, pp. 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.*, vol. 58, no. 6, pp. 2163–2172, Oct. 2010.

[3] W. B. Lee and C. C. Chang, "User identification and key distribution maintaining anonymity for distributed computer networks," *Comput.Syst. Sci. Eng.*, vol. 15, no. 4, pp. 113–116, 2000.

[4] 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.*, vol. 57, no. 2, pp. 793–800, Feb. 2010.

[5] A. Valenzano, L. Durante, and M. Cheminod, "Review of security issues in industrial networks," *IEEE Trans. Ind. Inf.*, vol. PP, no. 99, 2012, DOI 10.1109/TII/2012.2198666.

[6] T.-S.Wu and C.-L. Hsu, "Efficient user identification scheme with key distribution preserving anonymity for distributed computer networks,"*Comput. Security*, vol. 23, no. 2, pp. 120–125, 2004.

[7] Y. Yang, S. Wang, F. Bao, J. Wang, and R. H. Deng, "New efficient user identification and key distribution scheme providing enhanced security,"*Comput. Security*, vol. 23, no. 8, pp. 697–704, 2004.

[8] K. V. Mangipudi and R. S. Katti, "A secure identification and key agreement protocol with user anonymity (SIKA)," *Comput. Security*,vol. 25, no. 6, pp. 420–425, 2006.

[9] C.-L. Hsu and Y.-H. Chuang, "A novel user identification scheme with key distribution preserving user anonymity for distributed computer networks," *Inf. Sci.*, vol. 179, no. 4, pp. 422–429, 2009.

[10] B.Wang and Ms. Ma, "A server independent authentication scheme for RFID systems," *IEEE Trans. Ind. Inf.*, vol. 8, no. 3, pp. 689–696, Aug. 2012.

[11] B. Fabian, T. Esrmakova, and C. Muller, "SHARDIS: A privacy-enhanced discovery service for RFID-based product information," *IEEE Trans. Ind. Inf.*, vol. 8, no. 3, pp. 707–718, Aug. 2012.

[12] H.-M. Sun, Y.-H. Chen, and Y.-H. Lin, "oPass: A user authentication protocol resistant to password stealing and password reuse attacks,"*IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 651–663, Apr.2012.

[13] "Security Forumon Single Sign-On," TheOpenGroup [Online].Available:http://www.opengroup.org/security/l2-sso.htm

[14] J. Han, Y. Mu, W. Susilo, and J. Yan, "A generic construction of dynamic single sign-on with strong security," in *Proc. SecureComm'*, 2010, pp. 181–198, Springer.

[15] 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.
[16] C.-C. Chang and C.-Y. Lee, "A secure single sign-on mechanism for distributed computer networks," *IEEE Trans. Ind. Electron.*, vol. 59,no. 1, pp. 629–637, Jan.
2012.
[17] U. Feige, A. Fiat, and A. Shamir, "Zero-knowledge proofs of identity,"*J. Crytography*, vol. 1, no. 2, pp. 77– 94, 1988.
[18] G. Ateniese, "Verifiable encryption of digital signatures and applications,"*ACM Trans. Inf. Syst. Secur.*, vol. 7, no. 1, pp. 1–20, 2004.[19] H. Delfs and H. Knebl, *Introduction to Cryptography: Principles and Applications*, 2nd ed. Berlin, Germany: Springer, 2006.
[20] G. Wang, J. Yu, and Q. Xie, Security analysis of a singlesign-on mechanism for distributed computer networks
Cryptology ePrint Archive, Rep. 102, Feb. 2012 [Online]. Available:http://eprint.iacr.org/2012/107