ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 2, Issue 3, pp: (439-445), Month: July - September 2014, Available at: www.researchpublish.com

SECURITY ANALYSIS OF A SINGLE SIGN-ON MECHANISM FOR DISTRIBUTED COMPUTER NETWORKS

B. VASAVI

Abstract: **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. Recently, Chang and Lee proposed a new SSO scheme and claimed its security by providing well-organized security arguments. In this paper, however, we demonstrative that their scheme is actually insecure as it fails to meet credential privacy and soundness of authentication. Specifically, we present two impersonation attacks. The first attack allows a malicious service provider, who has successfully communicated with a legal user twice, to recover the user's credential and then to impersonate the user to access resources and services offered by other service providers. In another attack, an outsider without any credential may be able to enjoy network services freely by impersonating any legal user or a nonexistent user. We identify the flaws in their security arguments to explain why attacks are possible against their SSO scheme. Our attacks also apply to another SSO scheme proposed by Hsu and Chuang, which inspired the design of the Chang–Lee scheme. Moreover, by employing an efficient verifiable encryption of RSA signatures proposed by Ateniese, we propose an improvement for repairing the Chang–Lee scheme. We promote the formal study of the soundness of authentication as one open problem.**

Keywords: **Authentication, distributed computer networks, information security, security analysis, single sign-on (SSO).**

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 [1], [2] 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.

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 software systems without being prompted to log in again. Single sign-off is the reverse process whereby a single action of signing out terminates access to multiple software systems. As different applications and resources support different authentication mechanisms, single sign-on has to internally translate to and store different credentials compared to what is used for initial authentication. Single sign-on (SSO) mechanism provides a good remedy to this problem, as it allows a user with a single credential to access multiple service providers.

- ***Benefits of Single Sign-On***

1. Reducing password fatigue from different user name and password combinations
2. Reducing time spent re-entering passwords for the same identity
3. Reducing IT costs due to lower number of IT help desk calls about passwords

II. LITERATURE REVIEW & 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 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.

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 Ui registers a unique identity IDiwith a fixed bit length.
2. Obtain a secret token Si=(IDi||h(IDi))d mod N, from the SCPC through a secure channel where h(·) is a cryptographic one-way hash function.

C. User Identification Phase

Ui submits the request with a random nonce n1, m1 to Pj . On receiving m1, Pj chooses a random number k and then generates a random nonce n2. Pj calculates Z = gk mod n, u = h(Z||IDj ||n1), and the signature v = (u||h(u))dj mod Nj .Next, Pj sends the message m2 = {Z,v,n2} back to Ui. After receiving m2 from Pj, Ui computes u =h (Z||IDj ||n1) and performs the next step. Ui verifies the signature v by checking the equivalency of vej mod N?=(u||h(u))mod Nj . Otherwise, Ui informs Pj that someone has tampered with Z and aborts the protocol. Otherwise, Ui chooses a random number t to be his short-term private key and computes w = gt mod n. Ui calculates the parameter kij as kij = Zt mod n.

Ui generates a random nonce n3 and calculates three parameters Kij, x and y in accordance with the following equations: Kij = h(IDj ||k ij), the session key, x = Sih(Kij ||w||n2) mod N, y = E Kij (IDi||n3||n2), where E(·) is a symmetric crypto system such as DES or AES. Ui sends m3 = {w,x,y} to Pj After receiving m3, Pj computes kij as kij = wk mod n. Pj can obtain the session key Kij by computing Kij =h(IDj || kij). Pj uses Kij to decrypt cipher text y and retrieves IDi, n3, and n2. If n2 is valid, Pj computes SIDi = (IDi||h(IDi)). Pj verifies the validity of the identity IDi by checking SIDi h (Kij ||w||n2) mod N ? = xe mod N. If the equation holds, Pj trusts that Ui is a legal user. Pj computes V = h(n3) and sends m4 = {V } to Ui. After receiving m4 from Pj, Ui computes V ||= h(n3) and confirms that V? = V „. When both the equations are same, Ui trusts that Pj is an authorized service provider and Pj has really calculated the common session key Kij.

III. ARCHITECTURE

In the setting of RSA cryptosystem, such 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, the 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. Unfortunately, as we shall discuss later this efficient SSO scheme is not secure.

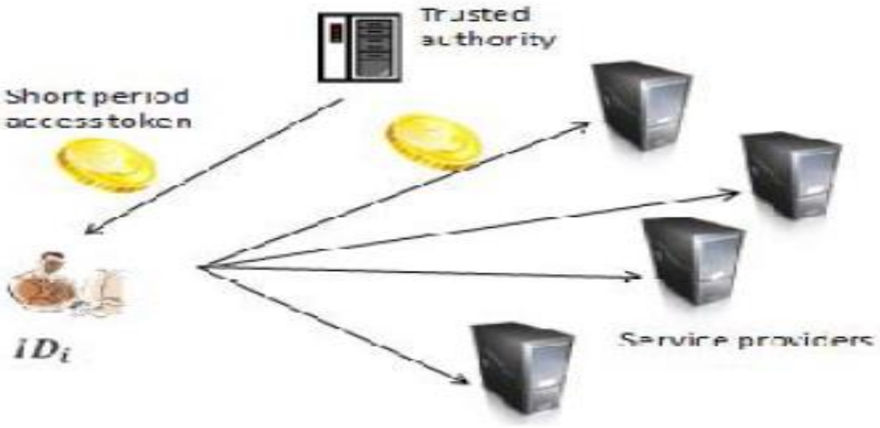


Fig1: Single Sign -On Mechanism

The Chang–Lee scheme is actually insecure by presenting two impersonation attacks, i.e., credential recovering attack and impersonation attack without credentials. In the first 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.

GOAL

The goal of distributed computing is to make such a network work as a single computer. Distributed systems offer many benefits over centralized systems, including the following:

- ***Scalability***

The system can easily be expanded by adding more machines as needed.

- ***Redundancy***

Several machines can provide the same services, so if one is unavailable, work does not stop. Additionally, because many smaller machines can be used, this redundancy does not need to be prohibitively expensive. Distributed computing systems can run on hardware that is provided by many vendors, and can use a variety of standards-based software components. Such systems are independent of the underlying software.

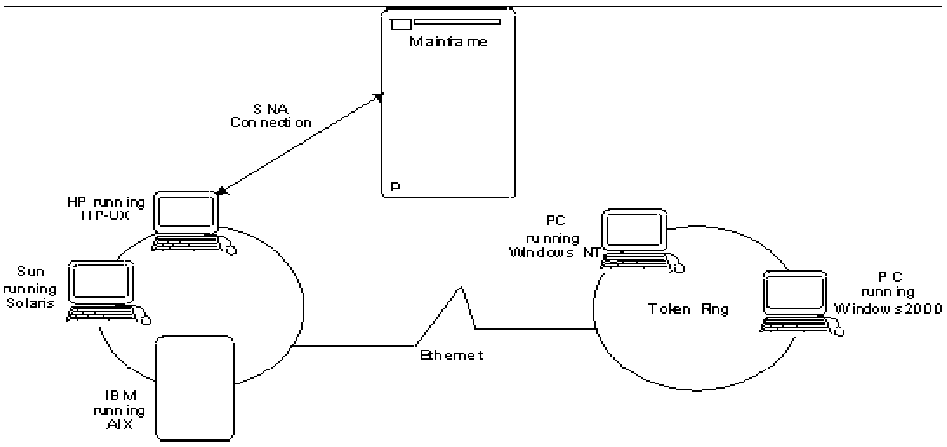


Fig 2: open distributed system

- ***Client – Server Model***

A common way of organizing software to run on distributed systems is to separate functions into two parts: clients and servers. A client is a program that uses services that other programs provide. The programs that provide the services are called servers.

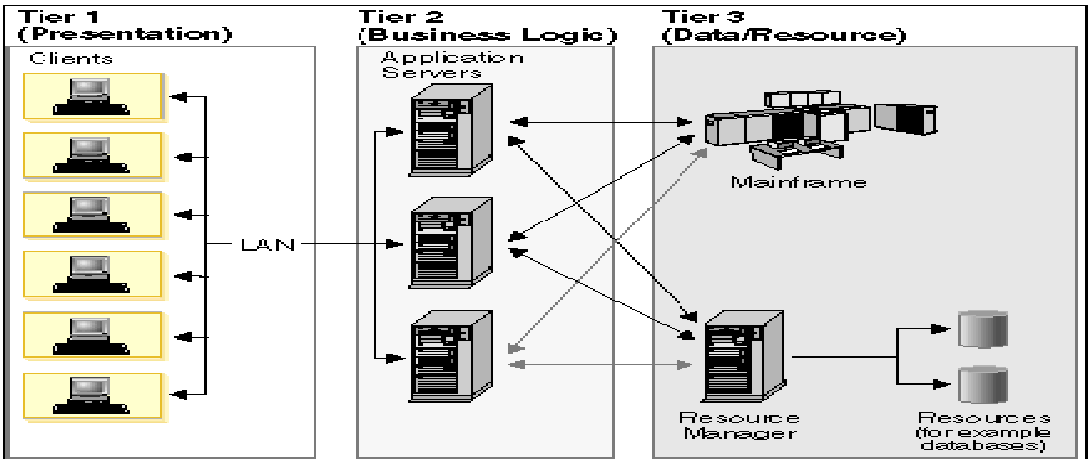


Fig 3: Three tier client / server architecture

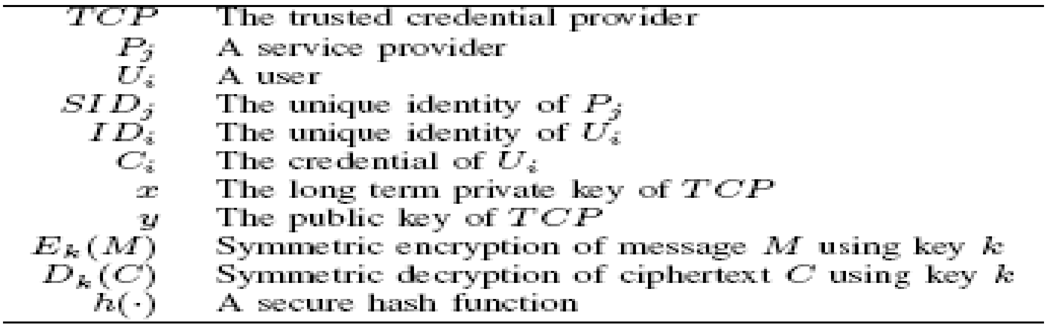
IV. PROPOSED SYSTEM

We use Schnorr signature [8][9] to overcome the drawbacks in Chang-Lee scheme as their user proof cannot provide soundness and credential privacy while Schnorr signature can. As a proveably unforgeable signature scheme [10], Schnorr signature allows a signer to authenticate him/herself by signing a message without releasing any other useful information about his/her private signing key. In the proposed scheme, the TCP first issues the credential for each user by signing the user's identity ID is according to Schnorr signature.

A) ***System Setup Phase:*** In this phase, TCP initializes his/her public and private parameters as Schnorr signature scheme. Firstly, TCP picks large primes p and q such that qjp□1, chooses a generator g of large safe prime order q in cyclic group G. Then, TCP sets its private key SK = x, where x 2 Z_q is a random number, and publishes its public key PK = y, where y = gx mod p.

B) ***Registration Phase:*** In this phase, user asks TCP for registration, then TCP issues a unique identity IDi via IdGen(RIi) and signs a Schnorr signature (a; e;C)

Table I
NOTATIONS USED IN THE SCHEME



TCP	The trusted credential provider
P_j	A service provider
U_i	A user
SID_j	The unique identity of P_j
ID_i	The unique identity of U_i
C_i	The credential of U_i
x	The long term private key of TCP
y	The public key of TCP
$E_k(M)$	Symmetric encryption of message M using key k
$D_k(C)$	Symmetric decryption of ciphertext C using key k
$h(\cdot)$	A secure hash function

for user's identity as credential generation algorithm CGen(IDi; SK). C is kept secret by user, while (a; e) will be made public. The details are given below.

➢ ***User Registration***: When a user Ui asks for registration, TCP selects a unique identity IDi and generates a credential Ci = (a; e;C) for Ui by selecting a randomness r 2 Z_q and computing a = gr mod p, e = h(a; IDi), and C = r +

xe mod q. Then, TCP sends identity IDi and credential Ci which is Schnorr signature for IDi to user Ui, where C should be kept as a secret.

➢ ***Service Provider Registration:*** Each Pj maintains a public/private key pair (PKj ; SKj) of any secure signature scheme. Here, algorithms SPPGen(_) and SPPV er(_) are identical to the signature generation and verification algorithms respectively.

C) ***Authentication Phase:*** In this phase, to authenticate him/herself user Ui signs a Schnorr signature the newly established session key Kij using credential C the signing Key, while Ui's session key material k2 is used as the commitment. Note that the corresponding verification key of C is gC, which can be recovered by computing gC = a _ ye mod p. For service provider authentication, any provably secure signature scheme can be used to authenticate a service provider in proposed scheme. The session key is established by using modified Diffie-Hellman key exchange scheme which has been formally proved in [11], and the user anonymity and unlink ability are preserved by using symmetric key encryption to encrypt a, e, and user's identity IDi. The details of this phase are illustrated in Figure 1 and further explained below.

1) User Ui chooses a random nonce n1 and sends M1 = (Req; n1) to Pj , where Req is a service request.

2) Upon receiving (Req; n1), Pj picks random number r1 2 Z_q , computes its session key material k1 = gr1 mod p, u = h(k1jjSIDj jjn1) and signs u to get a signature v = SPPGen(SKj ; u), and sends M2 =(k1; v; n2) to the user.

3) User Ui first computes u = h(k1jjSIDj jjn1) and verifies the signature v by checking if SPPV er(PKj ; u; v) = 1. If the output is "0", Ui terminates the protocol. Otherwise, Ui accepts the service provider Pj 's authentication, and then selects a random number r2 2 Z_q to compute k2 = gr2 mod p, kij = kr2 1 mod p, and the session key Kij = h(SIDj jjkij). After that, Ui signs Kij using his/her credential secret C by calculating ei = h(k2;Kij), z = r2+Cei mod q and ! = EK(IDijjn3jjn2jjejja), where n3 is a nonce chosen by Ui. Finally, Ui sends M3 = (!; z; k2) to service provider Pj .

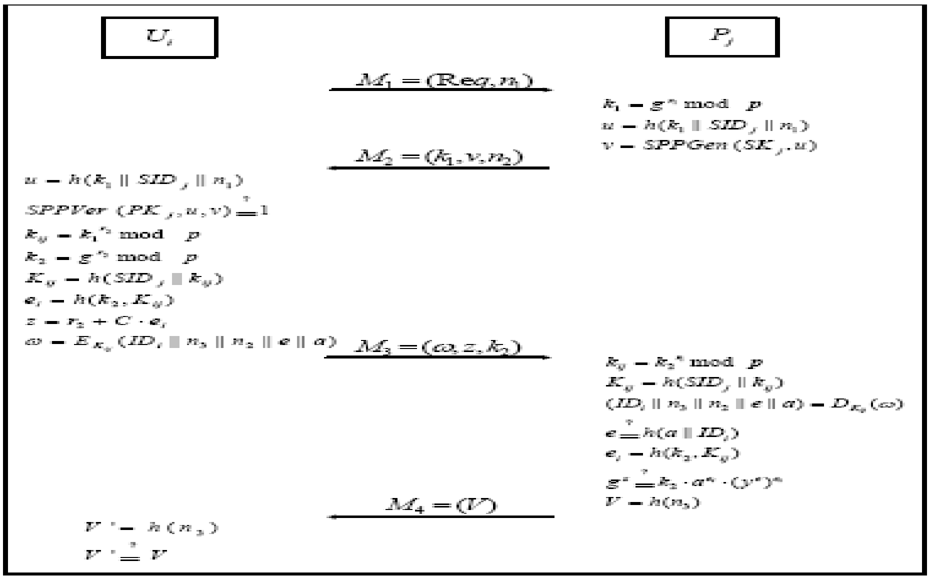

U_i
P_j
$M_1 = (Req, n_1)$
$k_1 = g^{r_1} \bmod p$
$u = h(k_1 \| SID_j \| n_1)$
$v = SPPGen(SK_j, u)$
$M_2 = (k_1, v, n_2)$
$u = h(k_1 \| SID_j \| n_1)$
$SPPVer(PK_j, u, v) \overset{?}{=} 1$
$k_{ij} = k_1^{r_2} \bmod p$
$k_2 = g^{r_2} \bmod p$
$K_{ij} = h(SID_j \| k_{ij})$
$e_i = h(k_2, K_{ij})$
$z = r_2 + C \cdot e_i$
$\omega = E_{K_{ij}}(ID_i \| n_3 \| n_2 \| e \| a)$
$M_3 = (\omega, z, k_2)$
$k_{ij} = k_2^{r_1} \bmod p$
$K_{ij} = h(SID_j \| k_{ij})$
$(ID_i \| n_3 \| n_2 \| e \| a) = D_{K_{ij}}(\omega)$
$e \overset{?}{=} h(a \| ID_i)$
$e_i = h(k_2, K_{ij})$
$g^z \overset{?}{=} k_2 \cdot a^{e_i} \cdot (y^e)^{e_i}$
$V = h(n_3)$
$M_4 = (V)$
$V' = h(n_3)$
$V' \overset{?}{=} V$

Figure 1. Participant Identification Phase

4) To verify z, Pj first calculates kij = kr1 2 mod p, derives session key Kij = h(SIDj jjkij) and decrypt ! with Kij to recover IDijjn3jjn2jjejja. Then, Pj checks if e = h(ajjIDi). If this does not hold, Pj aborts the protocol. Otherwise, the service provider computes ei = h(k2;Kij) and verifies z by checking if gz = k2_aei _(ye)ei mod p. If this holds, Pj accepts Ui's authentication, believes that they have shared the same session key Kij , and sends V = h(n3) as M4 to Ui.

5) User Ui computes V 0 = h(n3) and checks if V 0 = V . If this holds, Ui believes that he/she has shared the same session key Kij with Pj .

D) ***Security Analysis:*** We now analyze the security of the improved SSO scheme by focusing on the security of the user authentication part, especially soundness and credential privacy due to two reasons. On the one hand, the unforgeability of the credential is guaranteed by the unforgeability of RSA signatures, and the security of service provider authentication is ensured by the unforgeability of the secure signature scheme chosen by each service provider. On the other hand, other security properties (e.g., user anonymity and session key privacy) are preserved, since these properties have been formally proved in [12] and the corresponding parts of the Chang–Lee scheme are kept unchanged.

V. 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) = 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
}
```

Graph analysis

The simulated graphs evaluates the performance level of proposed system over existing system, the PDR , throughput enhancement factors evaluated through graphical analysis.

Advanced uses of AODV (ad hoc on-demand distance vector protocol)

- ✓ Because of its reactive nature, AODV can handle highly dynamic behavior of Vehicle Ad-hoc networks.
- ✓ Used for both uni casts and multicasts using the ’J’(Join multicast group) flag in the packets.

Limitations/disadvantages of AODV

- ✓ Requirement on broad cast medium.
- ✓ It is vulnerable to misuse: The messages can be misused for insider attacks including route disruption, route invasion, no deisolation, and resource consumption.
- ✓ AODV lacks support for high throughput routing metrics: AODV is designed to support the shortest hop count metric. This metric favors long, low-band width links over short, high-bandwidth links.
- ✓ High route discovery latency: AODV is a reactive routing protocol. This means that AODV does not discover a route until a flow is initiated. This route discovery latency result can be high in large-scale mesh networks.

DSDV - the destination sequenced distance vector protocol

DSDV is one of the most well known table-driven routing algorithms for MANETs. It is a distance vector protocol. In distance vector protocols, every node i maintains for each destination x a set of distances {dij(x)} for each node j that is a neighbor of i. Node i treats neighbor k as a next hop for a packet destined to x if dik(x) equals min j{dij(x)}.

Advantages of DSDV

- ✓ DSDV protocol guarantees loop free paths.
- ✓ Count to infinity problem is reduced in DSDV. We can avoid extra traffic with incremental updates instead of full dump updates.
- ✓ Path Selection: DSDV maintains only the best path instead of maintaining multiple paths to every destination. With this, the amount of space in routing table is reduced.

VI. CONCLUSION

In this paper, we demonstrated two effective impersonation attacks on Chang and Lee's single sign-on (SSO) scheme. The first attack shows that their scheme cannot protect the privacy of a user's credential, and thus, a malicious service provider can impersonate a legal user in order to enjoy the resources and services from other service providers. The second attack violates the soundness of authentication by giving an outside attacker without credential the chance to impersonate even a non-existent user and then freely access resources and services provided by service providers. We also discussed why their well-organized security arguments are not strong enough to guarantee the security of their SSO scheme. In addition, we explained why Hsu and Chuang's scheme is also vulnerable to these attacks. Furthermore, by employing an efficient verifiable encryption of RSA signatures introduced by Ateniese [13], we proposed an improved Chang–Lee scheme to achieve soundness and credential privacy. As future work, it is interesting to formally define authentication soundness and construct efficient and provably secure single sign-on schemes. Based on the draft of this work [14], a preliminary formal model addressing the soundness of SSO has been proposed in [15]. Further research is necessary to investigate the maturity of this model and study how the security of the improved SSO scheme proposed in this paper can be formally proven.

REFERENCES

[1] L. Lamport, "*Password authentication with insecure communication*",*Commun. ACM, 24(11)*: 770-772, Nov. 1981.

[2] Chin-Chen Chang, "*A secure single mechanism for distributed computer networks*," *IEEE Trans. On Industrial Electronics ,vol. 59, no. 1*, Jan 2012.

[3] Chin-Chen Chang, "*A secure single mechanism for distributed computer networks*," *IEEE Trans. On Industrial Electronics ,vol. 59, no. 1*, Jan 2012.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] C.P. Schnorr, "Efficient Identification and Signatures for Smart Cards", CRYPTO ,pp. 239-252, 1989.

[9] C.P. Schnorr, "Efficient Signature Generation by Smart Cards", J. Cryptology, vol. 4, no. 3, pp. 161-174, 1991.

[10] D. Pointcheval, J. Stern, "Security Arguments for Digital Signatures and Blind Signatures", J.Cryptology, vol.13, no.3,pp. 361-369, 2000.

[11] C.-C. Chang and C.-Y. Lee, "A Secure Single Sign-on Mechanism for Distributed Computer Networks", IEEE Transactions on Industrial Electronics, vol. 59, no. 1, pp. 629-637, 2012.

[12] 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.

[13] G. Ateniese, "Verifiable encryption of digital signatures and applications," *ACM Trans. Inf. Syst. Secur.*, vol. 7, no. 1, pp. 1–20, 2004.

[14] 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

[15] 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.