ISSN: 2250-3676
Volume-3, Issue-3, 111-116

Authentication & Key Establishment in Grid Computing Environments Using GDC

N.Sandeep Chaitanya[1], S.Ramachandram[2] , R Suhasini[3], S. Siva Skandha[4]

[1]Research scholar & Assoc.Professor, Dept of CSE, CMRCET n.sandeepchaitanya@gmail.com.
[2]Dean & Professor,Dept of CSE, College of Engineering Osmania University,Hyd, AP,India
[3] Asst .Prof, Dept of CSE, CMRCET, Hyd, hasini.r04@gmail.com
[4]Asst .Prof, Dept of CSE, CMRCET, Hyd, sivaskandha@gmail.com

Abstract: Grid computing has recently gained tremendous momentum but still is in its infancy. It has the potential for significant cost reduction and the increased operating efficiencies in computing. Although security issues are delaying its fast adoption, gri d computing is an unstoppable force and we need to provide security mechanisms to ensure its secure adoption. Grid security is a key component in Grid computing.. The mainstream Grid security solution, Grid Security Infrastructure (GSI) for Globus Toolkit (GT), offers comprehensive security services. This is achieved by applying public-key cryptography, cryptographic protocols methodologies and the necessary infrastructural supporting services in which public-key authentication framework (PKI) is the main component. A desired distinction would be that security services for Grid security should manifest and facilitate the Grid feature of advanced resource sharing. In this article, we focus on Grid data storage security, which has always been an important aspect of quality of service. To ensure the correctness of users' data in the grid we Propose providing security using GDC. Public-key digital certificate has been widely used in public-key infrastructure (PKI) to provide user public key authentication. However, the public-key digital certificate itself cannot be used as a security factor to authenticate user. In this paper, we propose the concept of generalized digital certificate (GDC) that can be used to provide user authentication and key agreement. A GDC contains user's public information, such as the information of user's digital driver's license, the information of a digital birth certificate, etc., and a digital signature of the public information signed by a trusted certificate authority (CA). However, the GDC does not contain any user's public key. Since the user does not have any private and public key pair, key management in using GDC is much simpler than using public-key digital certificate. The digital signature of the GDC is used as a secret token of each user that will never be revealed to any verifier. Instead, the owner proves to the verifier that he has the knowledge of the signature by responding to the verifier's challenge. Based on this concept, we propose discrete logarithm (DL)-based protocol that can achieve user authentication and secret key establishment.

Keywords—*Grid Computing, Security, Public-key digital certificate, user authentication, key management, TPA.*

--********--

1. INTRODUCTION

Grid Computing has been envisioned as the next generation architecture of IT enterprise, due to its long list of unprecedented advantages in the IT history: on-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk [1]. As a disruptive technology with profound implications, Grid Computing is transforming the very nature of how businesses use information technology. One fundamental aspect of this paradigm shifting is that data is being centralized or outsourced into the Grid. From users' perspective, including both individuals and IT enterprises, storing data remotely into the grid in a flexible on-demand manner brings appealing benefits: relief of the burden for storage management, universal data access with independent geographical locations, and avoidance of capital expenditure on hardware, software, and personnel maintenances, etc [2].

While Grid Computing makes these advantages more appealing than ever, it also brings new and challenging security threats towards users' outsourced data. Since grid service providers (GSP) are separate administrative entities, data outsourcing is actually relinquishing user's ultimate control over the fate of their data. As a result, the correctness of the data in the grid is being put at risk due to the following reasons. First of all, although the infrastructures under the grid are much more powerful and reliable than personal computing devices, they are still facing the broad range of both internal and external threats for data integrity. Examples of outages and security breaches of noteworthy grid services appear from time to time [3]–[5]. Secondly, for the benefits of their own, there do exist various motivations for grid service providers to behave unfaithfully towards the grid users regarding the status of their outsourced data. Examples include grid service providers, for monetary reasons, reclaiming storage by discarding data that has not been or is rarely accessed, or even hiding data loss incidents so as to maintain a reputation [6]–[8]. In short, although outsourcing data into the grid is economically attractive for the cost and complexity of long-term large-scale data storage, it does not offer any guarantee on data integrity and availability. This problem, if not properly addressed, may impede the successful deployment of the grid architecture.

The primary requirements by Grid security are:

- Need of secure communications for the Grid setting of virtual organizations (VOs). A VO typically is composed of users and resource providers beyond conventional organizational boundaries. Thus a centrally-managed security solution won't suit GSI.
- Ease of use by users. An important element in this requirement is the need for provisioning "single sign-on" for users, including delegation of credentials for computations that involve multiple resources and/or sites.
- Applications of standard technologies. This not only facilitates fast and ready deployment of the Grid technologies, but also helps to ensure correct applications of security techniques. GSI actually meets these requirements very well.

In specific, these are achieved via innovative applications of public-key certification infrastructure (PKI) in a novel notion of proxy certificate, and Security Proxy online servers. We shall omit describing these techniques here as they should already be familiar to the expected audience of this document. It is however our understanding that the current GSI practice does not make a noticeable impact on meeting the first requirement in Section 1.3. Let us consider the most general setting for a VO of users and resource providers. In order for the VO to be able to define flexible and may be ad hoc security policies which need to be applicable to these entities in a uniformed manner, it is desirable that each of these entities has strong security means which can enforce them in the execution of the policy. For example, a VO policy may stipulate that a resource (or a file) can become usable (accessible) by a user only after the user has conducted certain work to have satisfied a collaboration or service policy. However, in the current GSI practice, security means that a user has an exclusive entitlement to an action provided a cryptographic credential is available. Indeed, in a non-TC environment, it is usually the case that a user has the full access to, and thereby unlimited usage of, an owned cryptographic credential, and this is a consequence of missing behavior conformation as an important security service. Without behavior conformation, it is very difficult for the VO to apply fine granularity control on VO policies. To put the problem in another way, the current GSI practice has coarse policy enforcement on VO entities: an entity is either an insider who then can do everything or otherwise an outsider who is supposed and hopefully to be able to do nothing.

As users no longer physically possess the storage of their data, traditional cryptographic primitives for the purpose of data security protection can not be directly adopted. Thus, how to efficiently verify the correctness of outsourced grid data without the local copy of data files becomes a big challenge for data storage security in Grid Computing. Note that simply downloading the data for its integrity verification is not a practical solution due to the expensiveness in I/O cost and transmitting the file across the network. Besides, it is often insufficient to detect the data corruption when accessing the data, as it might be too late for recover the data loss or damage. Considering the large size of the outsourced data and the user's constrained resource capability, the ability to audit the correctness of the data in a grid environment can be formidable and expensive for the grid users [8], [9]. Therefore, to fully ensure the data security and save the grid users' computation resources, it is of critical importance to enable public auditability for grid data storage so that the users may resort to a verifier , who has expertise and capabilities that the users do not, to audit the outsourced data when needed. Based on the audit result, verifier could release an audit report, which would not only help users to evaluate the risk of their subscribed grid data services, but also be beneficial for the grid service provider to improve their grid based service platform. A digital certificate is the combination of a statement and a digital signature of the statement. The well-known digital certificate is the "X.509 public-key digital certificate" [1]. The statement generally contains the user's public key as well as some other information. The signer of the digital signature is normally a trusted certificate authority (CA). The X.509 public-key digital certificate has been widely used in public-key infrastructure (PKI) to provide authentication on the user's public key contained in the certificate. The user is authenticated if he is able to prove that he has the knowledge of the private key corresponding to the public key specified in the X.509 public-key digital certificate. However, the public-key digital certificate itself cannot be used to authenticate a user since a public-key digital certificate contains only public information and can be easily recorded and played back once it has been revealed to a verifier or Third party Auditor (TPA).

In this paper, we propose an innovative approach which enables a user to be authenticated and a shared secret session key be established with his communication partner using any general form of digital certificates, such as a digital driver's license, a digital birth certificate or a digital ID, etc. We call this kind of digital certificate as a generalized digital certificate (GDC). A GDC contains user's public information and a digital signature of this public information signed by a trusted CA. However, in GDC, the public information does not contain any user's public key. Since user does not have any private and public key pair, this type of digital certificate is much easier to manage than the X.509 public-key digital certificates. The digital signature of the GDC is used as a secret token of each user. The owner of a GDC never reveals signature of GDC to a verifier in plaintext. Instead, the owner computes a response to the verifier's challenge to prove that he has the knowledge of the digital signature. Thus, owning a GDC can provide user authentication in a digital world. In addition, a secret session key can be established between the verifier and the certificate owner during this interaction.

There are three entities in a digital certificate application. They are the following:

a) Certificate Authority (CA): CA is the person or organization that digitally signs a statement with its private key. In PKI applications, the X.509 public-key digital certificate contains a statement, including the user's public key, and a digital signature of the statement. The difference between the GDC and the existing public-key digital certificate is that in a GDC, the public information does not contain any user's public key.
b) Owner of a GDC: The owner of the GDC is the person who receives the GDC from a trusted CA over a secure channel. The owner needs to compute a valid "answer" in response to the verifier's challenged "question" in order to be authenticated and establish a secret session key.

c) Verifier: The verifier is the person who challenges the owner of a GDC and validates the answer using the owner's public information and CA's public key.

In this paper, our goal is to propose a similar solution in electronic-world applications. We call it the generalized digital certificate (GDC). A GDC contains public information of the user and a digital signature of the public information signed by a trusted certificate authority. The digital signature will never be revealed to the verifier. Therefore, the digital signature of a GDC becomes a security factor that can be used for user authentication.

Related Work

Our proposed scheme is closely related to the ID-based cryptography [22]. In an ID-based cryptographic algorithms, each user needs to register at a private key generator (PKG) and identify himself before joining the network. Once a user is accepted, the PKG will generate a private key for the user. The user's identity (e.g. user's name or email address) becomes the corresponding public key. In this way, in order to verify a digital signature of a message, the sender sends an encrypted message to a receiver, a user only needs to know the "identity" of his communication partner and the public key of the PKG, which is extremely useful in cases like wireless communi- cation where pre-distribution of authenticated public keys is infeasible. However, in an ID-based cryptographic algorithm, it is assumed that each user already knows the identity of his communication partner. Based on this assumption, there is no need, nor have feasible ways, to authenticate the identity. This is the main advantage of ID-based cryptography. Due to this assumption, ID-based cryptography is only limited to applications that communication entities know each other prior to communication. While in our proposed GDC scheme, the user does not need to know any information of his/her communication partner. The public information of a GDC, such as user's identity, can be transmitted and verified by each communication entity. Furthermore, this information is used to authenticate each other. In other words, our proposed schemes support general PKI applications, such as Internet e-commence, that communication entities do not need to know each other prior to the communication. Our proposed solution is based on the combination of a conventional DL based digital signature scheme and the well-known (generalized) Diffie- Hellman assumption.

2. ElGamal Digital Signature

In the ElGamal scheme [25], a large prime p and a generator g in the order of $p-1$ are assumed to be shared by all users. The signer selects a random private key $x \in [1, p-2]$ and computes the corresponding public key $y = gx \bmod p$. The signer first randomly selects a secret parameter $k \in [1, p-1]$ with $\gcd(k, p-1) = 1$ and computes $r = g^k \bmod p$. Then, s is solved by knowing the signer's secrets, x and k, as
$m = ks + rx \bmod p - 1$,
where m represents the message digest of the message m'. (r, s) is defined as the digital signature of the message m'. The signature (r, s) can be verified by checking whether the equation is correct
$g^m = y^r r^s \bmod p$,
In an ElGamal signature scheme, the parameter r of the signature can be computed off-line as $r = gk \bmod p$. The signature component s is computed on-line. Readers can refer to [26] for more discussion on the design of DLbased signature schemes. Without loss of generality, we can represent the generalized signing equation for all DL-based signature schemes as $ax = bk + c \bmod p - 1$ where (a, b, c) are three parameters from the set of values (m, r, s). More specifically, each parameter can be a mathematical combination of (m, r, s). For example, the parameter a can be m, r or s. The verification equation is determined accordingly as $y^{\mathrm{a}} = r^b g^c \bmod p$. There are 18 generalized ElGamal-type signature variants.
Assume A and B have their private keys, xA and xB, and their corresponding public keys, $yA = g^{xA} \bmod p$ and $yB = g^{xB} \bmod p$, respectively, where p is a large prime integer and g is a primitive element of the multiplicative group modulo p. Only A and B can compute a shared secret $K_{A\mathrm{B}} = y_B{}^{xA} = y_A{}^{xB} = K_{BA} \bmod p$. DHA refers to the assumption that it is computationally infeasible to determine KA, without knowing the private key x_A or x_B. However, solving the private key x_A or x_B from the corresponding public key y_A or y_B is equivalent to solving
the discrete logarithm problem.

3. User Authentication and Key Establishment Protocol

1) Registration at CA: Let A be the certificate owner and B be the verifier. A needs to register at a CA to obtain a GDC. The CA generates an ElGamal signature (rA, sA) for user A's statement $m'A$ according to equation (1), where mA is the message digest of the statement $m'A$. Since the signature component rA is a random integer and does not

depend on mA, it does not need to be kept secret. However, the signature component sA is a function of the statement. Each owner needs to keep it secret from the verifier in the authentication process. Our user authentication and key establishment protocol is illustrated in Fig. 1.

2) Protocol: The authentication and key establishment protocol contains the following four steps:
a) The user A passes his user information $m'A$ and parameters(r_A, S_A) to the verifier B, where
$S_A = r_A{}^{sA}$ mod p.
b) After receiving m'_A and (r_A, S_A), the verifier checks whether $g^{mA} = y_{SA}{}^{rA}$mod p,
where y is the public key of the CA. If this equality holds true, the verifier B first randomly selects an integer $vB \in [1, p - 2]$, then computes a challenge $c_B = r_A{}^{vB}$ mod p and send c_B to the user A. Otherwise, the user authentication fails and the protocol is stopped.

3) The user A first uses his secret sA to compute the Diffie-Hellman secret key $K_{A,B} = c_B{}^{sA}$ mod p, $K'_{A,B} = D(KA,B)$, where $D(K_{A,B})$ represents a key derivation procedure with $K_{A,B}$ as an input. Then user A randomly selects an integer $vA \in [1, p - 2]$, computes $c_A = r_A{}^{vA}$ mod p and the response $Ack = h(K'_{A,B}, c_B \| c_A)$, where $h(K'_{A,B}, c_B \| c_A)$ represents a one-way keyed hash function under the key $K'_{A,B}$. The user A sends Ack and c_A back to B.

4) After receiving the Ack and c_A from the user A, the verifier B uses his secret v_B to compute the Diffie-Hellman shared secret key $K_{B,A} = S_A{}^{vB}$ mod p, $K'_{B,A} = D(K_{B,A})$, and checks whether $h(K'_{B,A}, cB \| cA) = Ack$ is true. If this verification is successful, the certificate owner A is authenticated by the verifier B and a onetime secret session key ${}_B{}^{vA} = r_A{}^{vAvB} = c_A{}^{vB}$ mod p is shared between A and B. This shared key can provide perfect forward security.
In order to be authenticated successfully by the verifier, in our protocol, the certificate owner needs to compute and send a valid pair (r_A, S_A) and Ack to the verifier in steps 1) and 3). The parameters (r_A, SA) need to satisfy
$g^{mA} = y^{rA} S_A$ mod p.
This pair of integers can be easily solved by anyone. However, we want to show that only the certificate owner A who knows the secret exponent of S_A can compute a valid Ack. This is because the verifier B can compute the one-time secret key $K_{B,A}$ used in generating the Ack as $K_{B,A} = S\ A^{vB} = r_A{}^{sAvB}$ mod p.

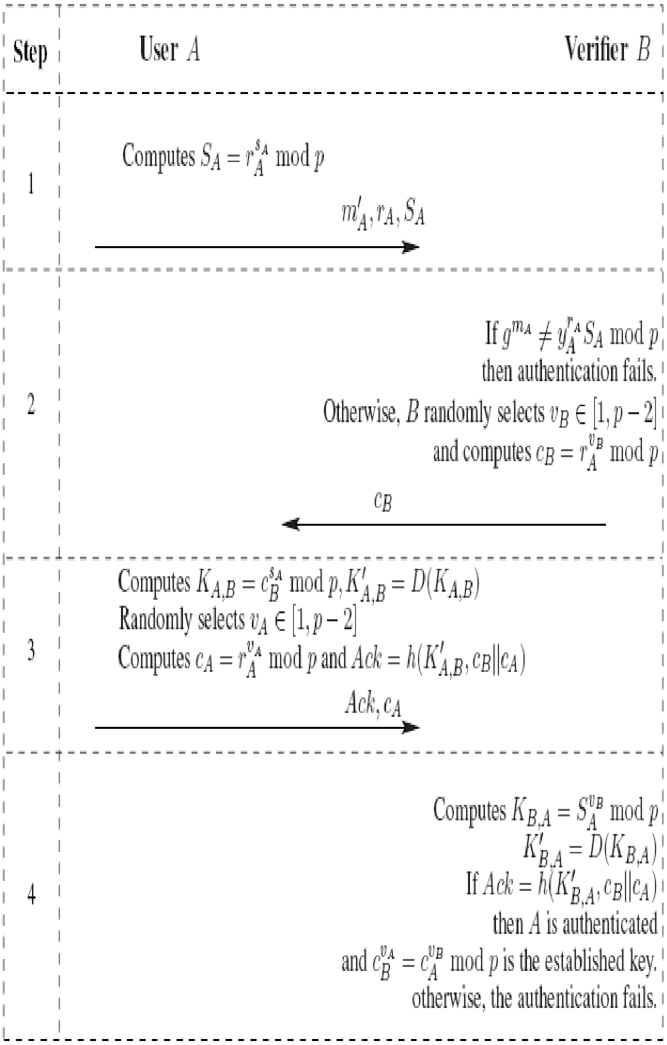
Step
User A
Verifier B
1
Computes $S_A = r_A^{s_A} \bmod p$
m'_A, r_A, S_A
2
If $g^{m_A} \neq y_A^{r_A} S_A \bmod p$
then authentication fails.
Otherwise, B randomly selects $v_B \in [1, p-2]$
and computes $c_B = r_A^{v_B} \bmod p$
c_B
3
Computes $K_{A,B} = c_B^{s_A} \bmod p, K'_{A,B} = D(K_{A,B})$
Randomly selects $v_A \in [1, p-2]$
Computes $c_A = r_A^{v_A} \bmod p$ and $Ack = h(K'_{A,B}, c_B \| c_A)$
Ack, c_A
4
Computes $K_{B,A} = S_A^{v_B} \bmod p$
$K'_{B,A} = D(K_{B,A})$
If $Ack = h(K'_{B,A}, c_B \| c_A)$
then A is authenticated
and $c_B^{v_A} = c_A^{v_B} \bmod p$ is the established key.
otherwise, the authentication fails.

Fig:1 DL based Authentication & Exchange of Keys

According to the DHA, the certificate owner A who knows the secret exponent of S_A can also compute $K_{A,B}$ as $K_{A,B} = c_B{}^{sA} = r_A{}^{sAvB} = K_{B,A}$ mod p. Thus, the certificate owner can interact with the verifier and be authenticated successfully.

4. Security Analysis and Discussion

In this section, we will analyze the security of the proposed user authentication and key establishment protocol for the unforgeability, one-wayness and nontransferability.
a) Unforgeability: In order to perform a forgery attack, the attacker needs to present a valid pair (r_A, S_A) in step 1)
and the corresponding Ack in step 3) in order to impersonate the certificate owner successfully. A valid pair (r_A, S_A) alone in step 1) cannot be used to authenticate the certificate owner since this pair of parameters can be solved easily by the attacker from equation (3). However, it is computationally infeasible for the attacker to find the discrete logarithm of SA because the security of the ElGamal signature

scheme.Therefore, it is computationally infeasible for the attacker to get a pair (r_A, S_A) to satisfy
$g^{mA} = y^{rA}r^{sA}{}_A$ mod p.
Due to the DHA, without knowing the secret exponent of S_A, it would be infeasible for the attacker to compute $K_{A,}$ and forge a valid Ack in step (3). On the other hand, the certificate owner obtains the secret exponent of S_A from
CA during the registration and the certificate owner can be authenticated successfully in step 3). In summary, the security of the unforgeability of our proposed protocol is provided through combination of the security of the ElGamal signature scheme and the DHA.Therefore, the proposed user authentication and key establishment protocol is secure against forgery attacks.
b) One-wayness: In step 1), the certificate owner presents SA to the verifier. The computation of secret s_A from S_A is infeasible since computation of s_A from the S_A is a discrete logarithm problem. Also, in step 3), the certificate owner uses the secret s_A to compute the Diffie-Hellman key $K_{A,B}$. Although the verifier knows the Diffie-Hellman key $K_{A,\,,}$ but due to the DHA, the verifier cannot obtain the secret s_A. Therefore, our proposed protocol satisfies the onewayness property.
c) Nontransferability : Due to the DHA, a valid response Ack can only be generated by a certificate owner who knows the secret digital signature component s_A such that $r\ {}_A{}^{sA} = S_A$ mod p, or by a verifier who knows the random secret of a random challenge selected by the verifier. As the verifier selects a random challenge each time, the response is only valid for a one-time authentication. Since the digital signature of a GDC is never passed to the
verifier, the verifier cannot pass the complete GDC to any third party. There is no privacy intrusion problem in our protocol. Therefore, a valid response Ack cannot be transferred into a response of another verifier's challenge.
Our protocol enables a certificate owner to be authenticated and two one-time shared secret keys $K_{A,}$ and $c_B{}^{vA} = r\ {}_A{}^{vAvB} = c$ A^{vB} mod be established between A, the certificate owner, who knows s_A such that $r_A{}^{sA} = S_A$ mod p, and the verifier B through the authentication protocol. The former is used to generate the Ack, and the latter is established shared secret key between A and B. In addition, it enables the owner to send a confirmation Ack to the verifier. Since the Diffie- Hellman secret shared key can be generated by either A or B, the certificate owner A can deny participating in the protocol.
In the original DHA, it is assumed that the generator g is a primitive element of the multiplicative group modulo p; while the parameter $rA = gk$ mod p in Theorem 1 is not necessarily a generator. However, we can ensure that rA is a primitive element of the multiplicative group modulo p by requiring $(k, p{-}1) = 1$ [27]. Particularly, when $p = 2p'{+}1$ is a safe prime, where p' is also a prime, we can ensure rA is a primitive element of the multiplicative group modulo p if k is an odd number.
Similar to the ID-based cryptographic algorithms, our proposed protocol also has the key escrow problem, that is the CA knows the one-time secret session key shared between the users. Some cryptographic algorithms have been proposed to solve the key escrow problem of the IDbased signature (IBS) while enjoying the benefits of the IBS, such as certificateless digital signature (CDS).

5. Conclusion

In this paper, we propose a privacy-preserving users auditing system for data storage security in grid Computing. We utilize GDC & the Hash key generated through Elgamal Digital Signature & Deffie Hellman Key exchange Protocol to guarantee that Verifier or TPA would not learn any knowledge about the data content stored in the grid during the efficient auditing process, which not only eliminates the burden of grid user from the tedious and possibly expensive auditing task, but also alleviates the users' fear of their outsourced data leakage. And also the user does not have any private and public key pair, this type of digital certificate is much easier to manage than the X.509 public-key digital certificates Considering TPA may concurrently handle multiple audit sessions from different users for their outsourced data files, we further extend our privacy-preserving public auditing protocol into a multi-user setting, where TPA can perform the multiple auditing tasks in a batch manner.

REFERENCES

[1] Network Working Group, "Internet X.509 public key infrastructure certificate and crl profile, RFC: 2459," Jan. 1999.

[2] C. Tang and D. Wu, "An efficient mobile authentication scheme for wireless networks," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 1408-1416, Apr. 2008.

[3] G. Yang, Q. Huang, D. Wong, and X. Deng, "An efficient mobile authentication scheme for wireless networks," *IEEE Trans. WirelessCommun.*, vol. 9, pp. 168-174, Jan. 2010.

[4] J. Chun, J. Hwang, and D. Lee, "A note on leakage-resilient authenticated key exchange," *IEEE Trans. Wireless Commun.*, vol. 8, pp. 2274-2279, May 2009.

[5] D. Chaum and H. van Antwerpen, "Undeniable signatures," *Advances in Cryptology - Crypto'89*, Lecture Notes in Computer Science, vol. 435, pp. 212-217, 1989.

[6] M. Bohøj and M. Kjeldsen, "Cryptography report: undeniable signature schemes," Tech. Rep., Dec. 15, 2006.

[7] X. Huang, Y. Mu, W. Susilo, and W. Wu, "Provably secure pairing-based convertible undeniable signature with short signature length," *Pairing- Based Cryptography -C Pairing 2007*, vol. 4575/2007 of *Lecture Notes in Computer Science*, pp. 367-391, Springer Berlin / Heidelberg, 2007.

[8] M. Jakobsson, K. Sako, and R. Impagliazzo, "Designated verifier proofs and their applications," *Advances in Cryptology - EUROCRYPT*,pp. 143-154, 1996. LNCS Vol 1070.

[9] D. Chaum, "Private signature and proof systems," 1996.

[10] R. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," *Advances in Cryptology-ASIACRYPT*, Lecture Notes in Computer Science, vol. 2248/2001, Springer Berlin / Heidelberg, 2001.

[11] J. Ren and L. Harn, "Generalized ring signatures," *IEEE Trans. Dependable Secure Comput.*, vol. 5, no. 4, Oct.-Dec., pp. 155-163, 2008.

[12] S. Saeednia, S. Kremer, and O. Markowitch, "An efficient strong designated verifier signature scheme," *ICISC 2003*, vol. 2836 of *SpringerLecture Notes in Computer Science*, pp. 40-54, 2003. HARN and REN: GENERALIZED DIGITAL CERTIFICATE FOR USER AUTHENTICATION AND KEY ESTABLISHMENT FOR SECURE COMMUNICATIONS 2379

[13] C. Schnorr, "Efficient signature generation by smart cards," *J. Cryptology*, vol. 4, no. 3, pp. 161-174, 1991.

[14] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption << cost (signature) + cost (encryption)," *Advances in Cryptology - Crypto'97*, Lecture Notes in Computer Science vol. 1294, pp. 165-179, 1997.

[15] F. Laguillaumie and D. Vergnaud, "Designated verifier signatures: anonymity and efficient construction from any bilinear map." IACR eprint.

[16] R. Steinfeld, L. Bull, H. Wang, and J. Pieprzyk, "Universal designated verifier signatures," in *Asiacrypt'03*, vol. LNCS 2894, pp. 523-542,2003.

BIOGRAPHIES

1. N.Sandeep Chaitanya, doing his research in JNTUH, received his B.Tech from DVRCET, Hyd, AP in Computer Science & Information Technology and M.Tech in Information Technology from Sathyabama University, Chennai. Presently he is working as Associate Professor at CMRCET.. He has published several papers in various International & National Conferences and Journals.. His research interests include Cloud Computing, Grid Computing, Mobile Networking, Computer Networks
and Network Security.

2. Dr S. Ramachandram received M.Tech & Ph. D Osmania University,. He has published several papers in International, national conferences and journals. He guided 9 research scholars. Presently he is working as Professor & Dean Dept of Computer Science, Osmania University. His research interests include Mobile ad-hoc networks, Cloud Computing, Grid Computing

3 R Suhasini recieved B.Tech & M.Tech from JNTUH. Presently she is working as Asst Professor at CMRCET.. She has published several papers in various International & National Conferences and Journals.. Her research interests include Image & Speech Processing, Cloud Computing, Grid Computing, and Network Security.

4. S.Siva Skandha, received his M.Tech from Bharath University chennai, in Computer Science Engineering Presently he is working as Asst. Professor CMR College of Engineering & Technology,AP. He has published several papers in various International & National Conferences and Journals.. His research interests include Grid computing & Computer Networks