Security Analysis in Hierarchical Resource Arrangements in Grid Computing

R.Menaka
Research Scholar,
Anna University of Technology,
Coimbatore, Tamilnadu, India

Dr.R.S.D Wahidha Banu
Supervisor
Anna University of Technology,
Coimbatore, Tamilnadu, India

ABSTRACT
As the days going on, desires of netizens increases to meet unforeseen emergency online demands for accessing the services, owning the resources for required period which exists in the e-pool distributive. The needs met by the on-the-fly framed infrastructure (i.e) virtual organization which assimilates the dynamic collection of diverse resources and provides the large-scale resource sharing and multi-institutional collaboration called grid computing. The prominent feature of this computing is the collaboration of manifold units to perform coordinated sharing tasks that basically rely on two functions: communication and resource sharing through the internet usage. As the Internet is not designed on strong security basis, there exist various attacks, due to malicious internal and external users. Securing grid communication and controlling access to shared resources in a fine-tuned manner are important issues for grid services. This paper is proposed to give remedial for key escrow and analysis of security attributes by mounting a secured grid infrastructure with Identity Based Cryptography**.**

General Terms
Grid Computing, Security, Public key cryptography, Identity based Cryptography, Security attributes.

Keywords
Discrete logarithms, Bilinearpairing, Private Key Generator, Single sign on, Delegation,

1. INTRODUCTION
Grid computing is a virtualized disseminated computing environment, that enables the runtime selection, sharing, and aggregation of (geographically) distributed autonomous (autonomic) resources based on the availability, capability, performance, cost, and simultaneously based on an organization's specific baseline and/or burst processing requirements.

The term "Grid" refers to systems and applications that integrate and manage resources and services distributed across multiple control domains in a coordinated fashion. All of which can be reached to fly-flung users whose aspiration is to solve urgent and/or resource-intensive problem (like data processing, network bandwidth, data storage and combination).

1.1 The Power of the Grid gorgeous
- To end users requiring fast transactional processing capabilities.
- To IT planners looking to control costs and reduce data center complexity.
- To each end-user as a service to increase business pace, reduce complication, streamline managing processes, and lower operational costs.

1.2 Purpose
Grid is a system which mingles the software and or hardware that provides and manages logically seamless access to those resources to meet desired objectives. Each an every entity exists in amalgamated environment likely to be from diverse setting either in the form of policy or platform. Compared to traditional systems which share the homogenous natured policy based network, our grid system need a stronger authentication and protection to user, data and code in which system computations executes. To inflict the guarantee in protection for key disclosure, key distribution and management among the entities participating in Grid community is essential. As the grid plausibly couples multiple resources hold by different individuals or organizations, the empathy of the right model sophisticated with security will avoid the forgery, malpractice and illegal action among the network users.

1.3 Study of the Cryptography
The Cryptography methods start from the classical cryptography which allow substitution and transposition operation, followed by symmetric model which adopt the possession of single key used both for encryption and decryption. Maintaining unique symmetric keys for each communicating pair of entities would require the management of (nxn-1)/2 keys, i.e., n is the number of nodes in the network. Risk in symmetric key management are : much keys for each pair, pain efforts taken to protect the key compromising , possibility of statistical analysis attack, absents of digital signature features and key agreement problem. Symmetric model mainly deals with encryption and decryption operation.

Next model which is named as the name called as the asymmetric model or as Public Key Cryptography. It generates a mathematically related pair of keys, one key as public and other key as private. The mathematical properties say that with the public key, we can verify an entity that has knowledge of the private key — but you cannot derive the private key from the public key. It implies the preeminent use of mathematical functions whose inverse is difficult to calculate**. E**xamples are RSA, ECC, Diffie_Hellman and DSS algorithm.

RSA plays an important role in this area. RSA gets its security from the difficulty of factoring very large numbers, i.e the multiplication of two large prime numbers together is (relatively) easy, but reverse is difficult. Diffie Helman uses the discrete logarithm problem which

concerns with hardness in finding a logarithm of a number within a finite field arithmetic system than applying the exponentiation of a number (e.g in equation $y= a^x \bmod p$ exponentiation is easy but the $\log_{a,p} y=x$ is hard where y, x is integer and a is primitive roots of prime number p**.** Thus the security based on the intractability of the discrete logarithm problem that allow remote entities wishing to confidentially communicate to meet each other to agree on a secret encryption key.

Grid Secure **I**nfrastructure with RSA depends on the integer factorization problem whose inverse is significantly more difficult. This scheme is more than enough to provide the security level but when compared with key size, it consumes more computational cost (shown in Fig 1).

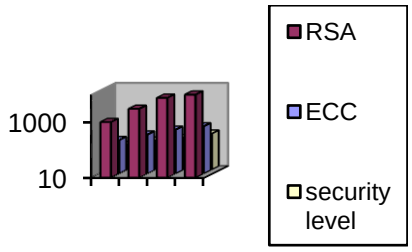

Fig.1 Comparative plots of RSA and ECC's bits strength

This makes us to progress towards IBC (Identity Based Cryptography) which capture the feature of elliptical curve cryptography (ECC). This ECC is related with the discrete logarithm problem. This approach follows the concept like Bilinear pairing, finite fields and**,** Bilinear Diffie-Hellman problem.

1.4 Security Requirement

CIA is a widely used benchmark for evaluation of information systems security, focusing on the three core goals of confidentiality, integrity and availability of information **(CIA Model)**

***1.4.1 Confidentiality*:** The protection of data from unauthorized disclosure through encryption.

1.4.2Availability: refers to the availability of information resources.

1.4.3 Integrity: The assurance that data received are exactly as sent by an authorized entity through hash function.

In order to be immune against any threats the valuable resources must be protected in such a way: when two parties are communicating, they want to ensure that the data they exchange are not eavesdropped (confidentiality), they are not altered by a third-party (data integrity), and that the sender is actually who he claims to be (entity authentication).

2. PAIRING INTRODUCTION

A pairing is a bilinear map defined over elliptic curve subgroups like $e : G_1 * G_2 \rightarrow G_T$ where G_1 and G_2 are cyclic groups of prime order q (i.e G_1 & G_2 denotes for additive and multiplicative notation respectively. The pairing is symmetric if G_1 is equal G_2 else it is asymmetric which satisfy the following properties:

2.1 Mathematical properties

A bilinear pairing on (G_1, G_T) is a map

$e : G_1 * G_2 \rightarrow G_T$

2.1.1. Bilinearity:

For all P,Q, S,T ε G_1 , G_2, respectively and a,b ε Z^* then it holds : $e(aS, bT) = e(S, T)^{ab}$.

The pairing is linear in both arguments in the sense of equation:
e(P + Q,R)= e(P,R)e(Q,R) for all points P, Q,R [7]
e(P,R + S) = e(P,R)e(P,S) for all points P,R, S

2.1.2. Non-degeneracy: $e(R, R) \neq 1$.

The pairing is alternating:
$e(P,Q) = e(Q, P)^{-1}$ for all points P, Q.

Consequently, the self-pairing of a point P is always 1 (eg.) $e(P, P) = e(P, P)^{-1} = 1$ for all points P.

2.1.3. Computability:

Pairing e can be efficiently computable.
The pairing is non-degenerate:
e(P,Q) = 1 for all points Q if and only if P = 1

2.1.4.Implemented proofs :

Verify of the pairing equation follows:
e(P1+P2,Q)=e(P1,Q)e(P2,Q)
Points denoted as PI,P2 and temp variable as Y,M1 and M2.

```
public class pairingtest
{public static void main(String args[]  {
 Pairing e = Predefined.ssTate();
//Select the Point P1 using the function like
e.RandomPointInG1(new Random());
//Select another point P1in G2
EllipticCurve g1 = e.getCurve();
//from the resulting curve add  P1& P2->Y;
// Choose Q from G2.
// Compute pairing(Y,Q)->M1;
FieldElement res =e.compute(P1,Q);
FieldElement res1 = e.compute(P2,Q);
Field gt = e.getGt();
FieldElement M2 = gt.multiply(res,res1);
if(M1.equals(M2res2))
        { System.out.println("Correct!
          e(P1+P2,Q)=e(P1,Q)e(P2,Q)");
} else
{ System.out.println("Something is wrong!
                e(P1+P2,Q)=e(P1,Q)e(P2,Q)");
} }}
```

2.2 Variant diffiehellmanproblem and key escrow

Let G be an Abelian group. In group G we can discuss the problem such as DDH (Decisional Diffie Hellman), CDH (Computational Diffie Hellmann) and DLP(Discrete logarithm problem). In DLP it is difficult to find the least positive integer 'a' such that the equation $h = g^a$ holds, when the elements g, h a, G are given. This problem is believed to be computationally hard. The computational Diffe-Hellman problem is to find g^{ab} when g, g^a, g^b a, G are given. The DDH has the problem of deciding if $h = g^{ab}$ when g, g^a, g^b , h, a , G are given.

2.3 Hardness visibility

If there is an algorithm A to solve CDH in polynomial time, then we will also able to construct A for solving DDH. It is noted as CDH → DDH. Another problem relation says DLP→ CDH**.**

2.4 Key escrow problem

The secret key generated by PKG (private key generator) is known to itself. Using this secret key it may decrypt the cipher text and forge the signatures, so no chance of influencing the aspect of the CIA model for secure transfer in distributed environment [12].

3. ID-BASED CRYTOGRAPHY (IBC):

It is suitable for grid environment because it is one amidst public key Infrastructure. The potential of IBC's features may well match with the dynamic traits in providing the security for grid environments are given below [13].

3.1 IBC's features

***3.1.1 Flexibility*:** More lightweight key usage and management and its certificate-free approach

3.1.2 Scalability : It maintain only N keys for N users instead of (n x n - 1) / 2 keys compared with symmetric cryptography.

3.1.3 Friendly: Communicating entity is represented by its own identity as public key (e.g IPaddress,mail id,etc.)

3.1.4 Intractability: Backward operation **is** better than forward operation as mentioned in discrete logarithms problem.

3.1.5 Suitability: As it uses the elliptical curve cryptography, the status says that alone 160 bits key of ECC can provide secure level equal to 1024 bits of RSA.

3.1.6 Single sign on: user login once and gains access to all multiple resources exist in diverse area without being prompted to log in again at each of them.

3.1.7 Delegation: **A**ssignment of authority and responsibility to another person to carry out specific activities

3.2 IBC TRAITS compared with public key infrastructure (PKI)

In traditional public key encryption, Bob's public key is a random string unrelated to his identity. When sender wants to send a message to anyone,she must first obtain receiver's id that is known aspect of his identity such as email id,Ip address etc is denoted as public key. The corresponding private key is are generated and distributed by a Trusted Authority (TA) in possession of a system master secret. This TA roughly corresponds to the Certificate Authority/Registration Authority (CA/RA) combination in a traditional PKI. The primary inspiration of IBC is to remove the public key distribution problem (i.e) maintained and stored in the form of directories and risk in secure distribution of public keys of users. Then the sender wants to send a message to anyone, she merely derives Bob's public key directly from his identifying information. Public key directories are unnecessary.

3.3 IBC basic functions

In IBC, the public key is usually related to the public identity of the user (for example, email addressee).
The following functional aspects are always present in any IBC cryptosystem.

3.3.1: ***System Setup***: IBC systems rely on a trusted central authority that manages the parameters with which keys are created. This authority is called the Private Key Generator (PKG). The PKG creates its parameters, including a master secret S used to generate the private keys for users. The system parameters are: the order q, the prime number p, the generator point P, PKG public point Ppub= S.P and the hash functions.

3.3.2: Encryption: When a user (A) wishes to send an encrypted message to another user (B), she encrypts the message to him by computing or obtaining the public key, KpubB, and then encrypting a plaintext message M with KpubB to obtain cipher message C.

3.3.3 ***Key Extraction*:** When Bob receives the message, he wants to decrypt it. He authenticates himself to the PKG and obtains the secret key KprivB that he uses to decrypt the cipher message C.

3.3.4. ***Decryption***: When Bob receives KprivB, he decrypts the cipher message C to obtain the plaintext message M.

3.4 Impetus shift to hierarchical id- based cryptography (HIBC)

In IBC, to single TA alone generate the private key to the entire node in the network using their respective public key. So, IBC model lacks in scalability [2] and key escrow remains. It need another model to suit with our grid, in which the size and member of VO dynamically changes. Hierarchical IdBased Crytography(HIBC) [4] allows TA to verify the identity and allow the root PKG need only to generate private keys for domain-level PKGs, who in turn generate private keys for users in their domains in the next level. Thus root PKG distribute the workload by delegating private key generation and identity authentication to lower-level PKGs. Like this burden of root PKG is avoided. Complete removal of key escrow cannot be achieved by HIBC as each domain level nodes act as PKG to generate secret key to the lower level nodes. We cannot expect each domain level PKG in this hierarchical approach may be benevolent. So at any case the secret may be leaked which may lead again to key escrow problem.

3.4.1 Shift from HIBC to HCLC (Hierarchical Certificateless Crytography)

It follows the same approaches as HIBC [15] but PKG generates only partial private key. It is the individual risk to compute the private key using the partial private key. As the root PKG or delegated PKG cannot compute others private key so it avoids key escrow.

3.4.2 *Hierarchical Certiificateless Crytography* (HCLC)

There are applications which do not tolerate key escrow, which is a feature of identity-based cryptosystems. This has led to the development of escrow-free variants of pairingbased public key cryptography, including Al-Riyami and Paterson's certificateless public key cryptography (CLPKC) [5][4].

A user's private key in the certificateless setting consists of two components: (i) an identity-dependent partial private key (generated in the same way as in the normal identity-based approach); and (ii) a full private key which can be produced using the partial private key and some secret known only to the user. Succinctly, this approach uses input from the PKG and the user to generate a private key,thereby eliminating key escrow. We now briefly describe a hierarchical certificateless encryption (HCLE) scheme and a hierarchical certificateless signature (HCLC) scheme.

3.4.2.1 Root Setup: The root PKG chooses a generator P_x belong to the group G1, picks a random s_x from Z^*_q , and sets Q_x (the product of generator and random number). It also selects required cryptographic hash functions. Keeping the secret key of the root

PKG"s secret, remaining all the param are made as public.

3.4.2.2 Lower-level setup: A lower-level entity (lower-level PKG or user) at level t picks a random no belong to Z^*_q which will be kept secret.

3.4.2.3Partial-private-key extract: sets the D_t to be $\sum_{i-1}^{t} s_{i-1} P_i = D_{t-1} + s_{t-1} P_t$ (eq.1) as entity's partial private key, where D_{t-1} is the parent's partial private key and s_{t-1} is a secret value only known to the parent.P the generator at each level

3.4.2.4 Set-private-key: This algorithm transforms a partial private key Dt of an entity at level t with ID-tuple (ID1,… IDt) into a private key St = stDt, where s_t is the secret value that the entity has chosen in LOWER-LEVEL SETUP.

3.4.2.5 Set-public-key: This algorithm sets a public key of an entity at level t with ID-tuple (ID1, … , IDt) as (s_t,P0, stQ0).

3.4.2.6 Encrypt: Given a message m and ID-tuple (ID_1, . . . , IDt), this algorithm computes the ciphertext taking as input the ID-tuple(ID_1, . . . , ID_t);, the public key (s_tPx, s_tQx) and the related Q-values to compute the ciphertext. The ciphertext is formulated as (U_0,U_2, . . . ,U_t, V,W)

3.4.2.7 Decrypt: Given a ciphertext, (U_0,U_2, . . . ,U_t, V,W) in this form takes as input the associated private key St and different Q values and it recovers m.

3.4.2.8 Sign: Given a private key St and a message m Є {0, 1}*,the signer with ID-tuple (ID1, . . . , IDt) computes h = H_3(ID_1, . . . , ID_t,m) Є G1 and sig = St +s_th. The algorithm outputs the (sig,Q_1, . . . ,Q_t) where each component is in G1.

3.4.2.9 Verify: Given a signature(sig,Q_1, . . . ,Q_t) of a message m, this algorithm takes as input the associated public key, computed from ID-tuple (ID_1, . . . , ID_t), and returns a message indicating the success or failure of the verification.

4. GRID WORKING MODEL

As the grid plausibly couples multiple resources hold by different individuals or organisations, how the interaction exists between components is described briefly with diagram,Fig.2.

1. User first authenticate and submit their request to the authenticator. Authenticator capture the valid credential as the long term credential which consists of private and public key[3].
2. Authenticator takes the credentials and job request and process with the help of Grid Secuirty Infrastructure which create User Proxy agent which will intiate the further process with short term public and private key.
3. User Proxy through Grid Middleware assign the task to the Grid Resource Broker GRB[1]

I4. Grid Resource Broker GRB contact the Grid Information Services (GIS) to find the resource available

5,6,7. Depending on the resource status given by GIS and authorized given by GSI , GRB generate the resource proxyagent and submit the job.

8,9,10. This Resource agent use the crediatial and delegation to process until the task ends.

5. SECURITY ANALYSIS:

5.1. Achievement of key secrecy: In the proposed scheme, all signers' private keys are generated by (eq.1 & eq.2). That is PKG generate only partial private key. Thus the individual node has to compute its private key using the partial private key. Even the attacker knows the public parameter first he has to break the intracability of BDH problems

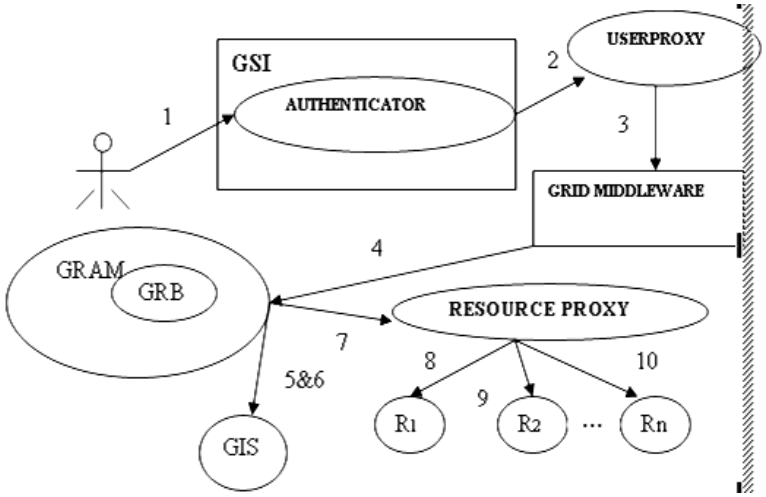

Fig: 2 Grid Working Model

GSI - Grid Security Infrastructure
GIS - Grid Information Services
GRB - Grid Resource Broker
GRAM - Grid Resource Allocation Mgmt

(Bilinear Diffe Helman Problem) for computing the secret value from captured partial secret[2].

5.2 Achievement of collision resistance:

As our Encryption and Decryption depends on Hash function which accept a variable size message M as input and produces a fixed size hash code H(M) [14]. The special feature of Collision resistance is given below:

1. For any given block x, it is computationally infeasible to find y!=x with H(y)=H(x)

2. It is computationally infeasible to find any pair (x,y) such that H(x)=H(y).

5.3 Achievement of possible security attributes

5.3.1 Known-key security:

If one session key is compromised then neither the private keys nor session keys (both past and future) are compromised as a result. As key formation is depends on the pre node and post node secret computation and especially on Bilinear DLP,the compromised of one's key will never expose the others**.**

5.3.2 *Forward secrecy*:

If private keys are compromised, the secrecy of previous session keys should not be affected. Eg: If the node relation is like this: Alice – User Proxy(UP)- Resource Proxy(RP1)-(RP2)_RP3—RPn. If RP3's secret key compromised, this will never leak the RP1, RP2 because RP2 by computation packed with id of user, user proxy, RP1.

5.3.3 Key-compromise impersonation resilience:

If RP1's private key is exposed, it does not enable an adversary to impersonate other entities to RP1.

5.3.4 Unknown key-share resilience:

Entity1 cannot be coerced into sharing a key with entity2 without entity1s knowledge**.**

5.3.5 Key control:

Neither entity1 nor entity2 can predetermine any portion of the shared secret key being established.

6. CONCLUSION

Any distributive networks commonly follow the PKI using public key cryptography such RSA,DiffeHelman keyexchange etc for their privacy in secure manner. Alternatively our proposed cryptography, such as IBC suits well for the flat structure of resource collaboration framework because authentication identity is endusers identity entity instead of random choice. The usage of IBC makes network lacks in scalability. Impetus change in the hierarchical pattern of resource collaboration framework which forms the virtual organization for processing large tasks in the Grid Environment adopts HIBC. Even it is security imbibed cryptography for scalability improvement it lacks in key escrow problems.With the help of next version HCLC the key escrow is removed. In addition to this, attacks possibilities for grid environment which share more than one resources in the hierarchical manner to compute the large computational task has been discussed shortly. The intractability of variant of discrete logarithm has been dealt with the help of bilinear pairing properties.

7. REFERENCES

[1] RajKumarBuyya, SteveChapin,and DavidDi Nucci, Architecture Models For Resource Management in the Grid.

[2] R.Sindhuja ,P.Varsha and Dr.G. Sumathi, An Improved ID-Based Entitled Verifier Crytography for

Grid Systems, International Journal of Recent Trends in Engineering, Vol2, No.1 November 2009.

[3] SianiPearson,MarcoCasassa-Mont & Manny Novoa, HewleetPackard labs,SecurityInformati-on Transfer in Distributed Computing Environment, IEEE Security and Privacy 2007.

[4] CraigGentry, Alice Silverberg,Hierarchical ID-Based Crytography,ASIACRYPT 2002, LNCS 2501,pp. 548-566,2002.

[5] Jason Crampton,Hoon Wei Lim,Kenneth G.Paterson,A Certificate-Free Grid Security Infrastructure Supporting Password-Based User Authentication, In Proceedings of the 6th Annual PKI R&D Workshop 2007. NIST Interagency Report.

[6] H.W. Lim and M.J.B. Robshaw. On identity-based cryptography and GRID computing.

[7] I.Foster, C. Kesselman, G. Tsudik, and S. Tuecke. A security architecture for computational Grids. In Proceedings of the 5th ACM Computer and Communications Security Conference, pages 83–92. ACM Press, 1998.

[8] Changyu Dong, Jpair: A Quick Introduction Oct.2010

[9] Anoop MS, Elliptic Curve Cryptography, An Implementation Guide.

[10] Certicom, Standards for Efficient Cryptography, SEC 1*:* Elliptic Curve Cryptography,Version 1.0, September 2000, Available at http://www.secg.org/ download /aid385/sec1_final.pdf.

[11] P. Barreto, H. Kim, B. Lynn, and M. Scott. Efficient Algorithms for Pairing-Based Cryptosystems. In CRYPTO 2002, volume 2442 of Lecture Notes in Computer Science, pages 47{53. Springer Verlag, 2002.

[12] Byoungcheon Lee, Colin Boyd et.al , Secure Key Issuing in ID-based Crytography. In Australasian Information Security Workshop 2004 [AISW 2004].

[13] D. Boneh and M. Franklin. Identity-Based Encryption from the Weil Pairing. Adv. in Cryptology | Crypto 2001, LNCS vol. 2139, Springer-Verlag, pp. 213{229, 2001. Full version to appear in SIAM J. Computing and available at http://eprint.iacr.org/2001/090.

[14] M. Naor and M. Yung. Universal One-Way Hash functions and Their Cryptographic Applications. 21st ACM Symposium on Theory of Computing (STOC), ACM, pp. 33{43, 1989.

[15] M.H. Au, J.K. Liu, T.H. Yuen, and D. S. Wong.(2006) Practical Hierarchical Identity Based Encryption and Signature schemes Without Random Oracles. http://eprint.iacr.org/2006/368