A GRID AUTHENTICATION SYSTEM WITH REVOCATION GUARANTEES

Babu Sundaram, Barbara M. Chapman

Computer Science Department
University of Houston
Houston, TX, 77204, USA
http://www.cs.uh.edu

Technical Report Number UH-CS-05-24
January 25, 2006

Keywords: Grid computing, Authentication, Digital Identities, Credential Revocation

Abstract

Credential revocation is a critical problem in grid environments and remains unaddressed in existing grid security solutions. We present a novel grid authentication system that solves the revocation problem. It guarantees instantaneous revocation of both long-term digital identities of hosts/users and short-lived identities of user proxies. With our approach, revocation information is guaranteed to be fresh with high time-granularity. Our system employs *mediated RSA* (mRSA), adapts Boneh's notion of *semi-trusted mediators* to suit security in virtual organizations and propagates proxy revocation information as in Micali's NOVOMODO system. Our approach's added benefits include a configuration-free security model for end-users of the grid and fine-grained management of users' delegation capabilities.

A GRID AUTHENTICATION SYSTEM WITH REVOCATION GUARANTEES

Babu Sundaram, Barbara M. Chapman

Abstract

Credential revocation is a critical problem in grid environments and remains unaddressed in existing grid security solutions. We present a novel grid authentication system that solves the revocation problem. It guarantees instantaneous revocation of both long-term digital identities of hosts/users and short-lived identities of user proxies. With our approach, revocation information is guaranteed to be fresh with high time-granularity. Our system employs mediated RSA (mRSA), adapts Boneh's notion of semi-trusted mediators to suit security in virtual organizations and propagates proxy revocation information as in Micali's NOVOMODO system. Our approach's added benefits include a configuration-free security model for end-users of the grid and fine-grained management of users' delegation capabilities.

Index Terms

Grid computing, Authentication, Digital Identities, Credential Revocation

I. Introduction and Motivation

Kohnfelder introduced the notion of a "digital certificate" [27]. In public-key (or, asymmetric key) encryption systems [26][34], digital certificates ascertain the identities of users, hosts and services (collectively termed as end entities). A digital certificate C is a trusted third-party's signature that validates the binding of a public key (PK) to an entity's identity (I). The trusted third-party is called a Certificate Authority (CA) and the CA uses its private key to sign and issue end-entity certificates (EEC). The clients that require a certificate generate a public-private key pair and submit the public key along with other required identity information to the CA. The CA verifies this information and if satisfied, signs the client's public key and includes information such as a serial number (SN), a start date d1 and end date d2 of its validity. In essence, a digital certificate C = Sign_{CA}(I, PK, SN, d1, d2). Now, any acceptor wanting to verify the identity of an entity E does so by checking that E's certificate includes a valid signature from a trusted CA.

Often, the life-time of digital certificates is in the order of years after which they expire. However, situations might arise that warrant immediate revocation of a certificate even before its actual expiration time. For instance, a trusted user Alice might leave her company or suspect that her private key has been compromised. Now, it is essential to immediately revoke her public key certificate so as to prevent acceptors from honoring stale or compromised credentials. An important design consideration in any CA implementation is handling prompt certificate revocation. Some studies [29] estimate that roughly 10% of public keys certified by a CA are revoked before they expire. Lack of a scalable revocation mechanism with freshness guarantees about revocation information inhibits widespread deployment of public-key based systems.

We now stress the "revocation problem" and its importance in grid environments. Grids [17] are persistent infrastructures for securely sharing distributed and diverse hardware and software resources among dynamic collections of individuals, institutions and resources called virtual organizations (VO) [20]. Grid security demands generality and transparency in authentication and has to ensure system integrity in a networked environment and integrity of data communicated between grid entities. The Globus project [16][18] develops protocols and services for constructing grids and provides an implementation of grid middleware tools. Grid Security Infrastructure (GSI) [8] is the grid authentication protocol of Globus Toolkit (GT) and provides services such as single sign-on, credential delegation and identity mapping. It is primarily based upon and requires a public key infrastructure (PKI)

[46] for its operation.

GSI allows grid entities to mutually authenticate using X.509 [53] certificates. Resource-specific identity of a user is derived from the digital identity included in his certificate. GSI introduced the notion of *"proxies''*, an additional set of temporary, short-lived credentials derived from user's long lived certificate to perform delegation on-behalf of the user. This eliminates the need for the users to remain online or enter passwords repeatedly whenever grid resource access is desired. However, GSI provides only little support for revocation capabilities for long-term certificates and no support for revoking compromised user proxies. Moreover, due to the scale of resources involved in a VO, hierarchical CAs have become a common approach to manage digital identities of grid entities. This exacerbates the revocation problem in grids as recent works [7] indicate such hierarchical public key certification is increasingly becoming the target for attackers. Hence, it has become vital to ensure revocation support for the long-lived digital identities of the grid resources and users as well as the short-lived proxy identities.

In short, our scheme makes use of a variant of RSA cryptosystem called mediated RSA (mRSA) [33] as defined by Boneh and others [4] and extends the notion of "`semi-trusted mediators"' (SEM) to fit security in virtual organizations. Each virtual organization hosts a SEM-like entity, discussed in detail in section 4, to handle revocation of long-term X.509 certificates. However, in our model, the private key of a grid entity (E) is split by mRSA into two parts based on a simple 2-by-2 threshold cryptography [21]. Knowledge of a half-key cannot be used to derive the entire private key. Part of the key is held by E and the other part is held by E's SEM. Hence revocation of compromised host credentials is instantaneous as the trusted SEM can simply stop using its part of the key. Implicitly, the problem of key escrow is also eliminated by the design of mRSA. We envision that the number of VOs will be much lesser than that of resources and individuals hosted by them. To allow support for delegation and resource-side user mapping, we retain features of GSI to manage user identities via long-term certificates. But, we add revocation capabilities for user proxies by modifying the proxy-creation process to handle validity and revocation targets as in Novomodo scheme. The complete architecture and the protocols of our system are discussed in detail in section 4.

We give complete details of our system and its functioning in section 4. The rest of the paper is organized as follows. Section 2 gives an overview of the related work to handle the certificate revocation problem. Section 3 is a brief overview of GSI component of Globus toolkit, the de-facto standard in grid security. In section 4, we present a detailed explanation of our system components and the protocols for grid authentication. We summarize the effort and conclude in section 5.

II. Overview of Related Work

Many techniques exist to realize revocation of digital identities. The following list indicate the major efforts in this direction.

Certificate Revocation List, (CRL) is the traditional, well-known and most-popular PKI proposal [55] to manage revocation by using explicit revocation structures. A CRL is a CA-signed list of certificates that are revoked before their intended expiry time. Any CA implementation produces such lists periodically and distributes them through online servers or repositories. The acceptor of an entity E's certificate simply consults a CRL to ensure that E's certificate is not in the latest CRL. If a certificate is not expired, but present in a CRL, then the acceptor can safely reject the certificate. If E's certificate is not present in the CRL, then it is considered as valid.

Unfortunately, this is a very inefficient mechanism. CRLs tend to grow into unmanageable sizes with time and hence pose severe bandwidth requirements and transmission costs. In some major PKI implementations [45], it has been noted that CRLs form the most expensive component. Though improvements have been suggested over conventional CRLs in the form of delta CRLs, the long intervals between CRL distribution often result in stale revocation information. This is also known as the "time granularity problem"} inherent in CRLs by design and is present in the segmented CRL proposals too. Also, the associated infrastructural and bandwidth costs prove to be prohibitive in VOs. Further, CRLs are issuer-driven approaches and hence lack facilities to address the recency requirements as may be mandated by the credential acceptors. On some occasions, implementations of certain CRL infrastructures have been openly questioned from several security professionals and part of major security vulnerabilities.

Online Certificate Status Protocol, (OCSP) is another PKI proposal [54] where a CA replies to certificate status queries with a freshly generated signature. It is a simple request/reply protocol allowing an acceptor to query a CA, based on serial number, for the current status of a certificate under question. When the CA gets the query, it checks the certificate corresponding to the serial number for any revocation. If the identity is revoked, the CA indicates it to the querying acceptor. If the certificate is valid, it confirms it by generating and issuing a fresh certificate.

However, this model requires the CA/validation server to be available online and if the validation server implementation is centralized, it becomes vulnerable to Denial of Service (DoS) attacks. Though it reduces the reply size per a single status query, it poses significant computational demands on the CA due to the computationally expensive signature operations. This setup could easily outrun the CA resources under a heavy stream of incoming certificate status queries.

NOVOMODO Certificate Validation System is a novel revocation scheme proposed by Micali [29]. The overall system includes the CA aided by a few servers referred to as "directories" that distribute revocation information. Briefly, the CA, at the time of issuing the certificate to the client, includes a 160-bit hash value indicative of the revocation information about the certificate. This hash value is derived as below.

> *The CA generates two random 20-byte value X0 and Y0 and uses a publicly-known one-way hash function to hash these values. The successive hash values of X0 are indicated as X1, X2, X3 and so on. That is, X1 = H(X0); X2 = H(X1);... X{365}=H(X{364}). And, Y1=H(Y0). To generate a certificate with a lifetime of 1 year, the CA computes X1 through X{365}. Micali refers to X{365} and Y1 as validity target and revocation target respectively.*

Then, the CA creates the certificate as C=Sign{CA}(U, PK, SN, d1, d2,...,Y1,X{365}). On any given day n, the CA distributes the targets of all its clients after checking their revocation status. For revoked clients, the CA distributes their corresponding Y0 values to the "directories". For valid clients, the CA distributes X{365-i} to the directories. Hence, a verifier of a certificate C on day i will query the "directories" for the status and obtain a target value (X or Y depending on the validity of C) in reply. When a X value is returned, the verifier can ensure C's validity by checking H^i(X{n-i}) equals Xi. Otherwise, when Y0 is returned, the verifier can confirm C is revoked by checking Y1 in the certificate equals H(Y0). NOVOMODO directory responses are concise (20-bytes) and directories cannot forge validity targets since one-way hash functions are known to be hard to invert. Also, the computational demands on the CA are minimal as hashing is orders of magnitude cheaper to compute than signatures.

However, NOVOMODO involves third-party queries and as noted in [22], it is often necessary to deincentivize third party queries. As the certification base can grow large over time all servers are required to keep track of the certificate status of all clients. And, the number of queries for a given certificate could increase dramatically as the number of querying verifiers increases, a common scenario in grid environments.

Identity-based Encryption(IBE) is a public-key cryptosystem where any arbitrary string, such as a person's e-mail address or host's IP address, can act as the public key. A trusted third-party called a Private Key Generator (PKG) issues private keys to its clients based on their public keys and some shared security parameters params. Shamir [35] conceptualized this idea in 1984 with simplified certificate management as the primary motivation. Fully practical IBE schemes have been developed only in the recent past. An IBE system comprises of the following four algorithms: Setup, Extract, Encrypt and Decrypt. The Setup operation takes some security parameters params and generates the master key s held secret by the PKG. s is masked and published to its clients. Extract takes in params, an arbitrary public key (ID) and derives its private key K{ID}. Encrypt allows senders to produce the cipher text C corresponding to message M based on the public key IDc of a client U. That is, C = Encrypt(M, IDc, params). Decrypt allows the intended recipient to decipher C to get M using its private key Ku. That is, M = Decrypt(C, params, Ku). With this setup, U's certification is implicit because U can decrypt C only after getting Kc from the trusted PKG}. Implicit certification eliminates the need for third-party certificate status queries. The security of such an IBE system is shown to have its security based on a Bilinear Diffie-Hellman (BDH) [25] assumption (that is, given the points P, aP, bP, cP on an elliptic curve, it is impossible to efficiently compute f(P,P)^{abc}). IBE exhibits instantaneous revocation since PKG can simply stop issuing Ku if U's identity is to be revoked and U can decrypt no further. Also, this model minimizes exposure of clients' private keys and allows implementation of simpler privilege delegation models. More details on this approach can be found in [6][28].

IBE scheme has two major drawbacks. First, the key escrow problem, that is, a PKG can decrypt messages intended for its clients. Also, a compromised PKG will let the attacker to learn all the clients' private keys and decrypt the communications. Nevertheless, on some limited occasions, key escrow is cited as a desired feature, for example, a company might want to retain access to encrypted files of an employee even after she leaves the company. Second is the key distribution problem where the PKG has to communicate the private keys to its clients over secure channels to thwart eavesdroppers. This has been a known problem even in the traditional secret key systems.

Certificate-based Encryption(CBE) is an cryptosystem proposed by Gentry [22] where the certificate serves its traditional purpose and additionally acts part of the decryption key. It tries to make use of best aspects of IBE and PKI. It eliminates the problem of key escrow by using double encryption. In this model, as with PKI, the client generates the public-private key pair and requests a certificate from the CA. The CA now uses an IBE scheme as in [5] to generate the contents of and sign the certificate. So, Alice wishing to send a encrypted messages to Bob doubly encrypts her message. This requires Bob to have his private key as well a up-to-date certificate from the CA to decrypt the message. Thus, certification in this model is implicit as the CA can stop issuing fresh certificates (part of decryption key) to Bob preventing him further decryption capabilities.

But, this approach poses significant computational burden on the CA since it has to respond to queries with freshly generated certificates. Gentry refined the basic CBE scheme to make use of subset covers} to reduce the computational demands. Thus, a CA with N clients only needs to compute an average of R{total} * log (N/R{total}) certificates, where R{total} is the total number of clients with revoked certificates. Further, hierarchical CBE models with subset covers have been proposed so a CA has to compute only R{period} * log (N/R{period}) certificates per period, where R{period} is the number of certificates revoked in the previous period. For example, for 10\% certificate revocation rate with a CA having N = 250 million, R{hour}~.1N/(365*24) \approx 2850. Hence, the CA has to recompute only 13 reconfirmation certificates per second.

However, CBE model requires CA to be an online entity in order to answer status queries. Further, such a CA setup become attractive targets for attackers and susceptible to DoS attacks. Also, the CA is necessitated to take part in multiple revocation queries from various acceptors, even for the same certificate and thereby increasing the transmission costs.

Semi-trusted Mediator (SEM) Architecture was introduced by Boneh [4] in conjunction with a mediated RSA cryptosystem to realize fine-grained control over security capabilities. Mediated RSA (mRSA) is a simple threshold variant of RSA public key cryptosystem. With this approach, the clients do not generate their keys. The CA entity initializes and distributes the mRSA keys to its clients. The threshold variant splits the private key d} of an entity into two parts d{sem} (distributed to SEM) and du (distributed to the client) such that

$$d = d\{sem\} + du \bmod \phi(n)$$

where n is the product of two large primes p and q as in any RSA implementation. mRSA is completely transparent to the encryption and signature verification operations. Complete details of the algorithms for mRSA key generation, message encryption/decryption and signature/verification can be found in [4].

For successful decryption or signature generation, the client and SEM must co-operate and exercise their respective portions of the private key. No private key-based operation is possible without mutual consent between the client and the SEM. The mediators (SEMs) are only semi-trusted} because an acceptor trusts the SEM to have verified the revocation status of a client. As in IBE and CBE, the certification is implicit and allows fast revocation. In a typical setup, there will be one CA and a set of SEMs that cater to the needs to a larger number of clients. SEMs cannot issue forged messages on behalf of revoked users (since it does not have du portion of the key to generate signatures).

Part of our work SEM as it provides a good model for achieving mutual authentication between grid resources and users. However, its scope disallows handling revocation of user's delegated credentials. Partly, this is because the assignment of SEMs to users is a fixed, static setup and it requires key generation support from the CA. But, user proxies are generated dynamically by the users as need for resource access arises and cannot expect CA's involvement for such frequent credential generation.

In the next section, we briefly describe the widely-used existing grid security solution, GSI of the Globus toolkit.

III. Grid Security Infrastructure (GSI)

GSI of the Globus toolkit is a stand-alone security model that accounts for authentication and secure communication between elements in a grid. It uses public key cryptography as the basis for its functionality. Users/resources in GSI require a long-term (with lifetime in the order of years) public key certificate-private key pair (typically RSA keys) issued from a CA trusted by the target resources. In grid settings, the user intervention cannot be requested every time a need for a resource access arises either from the user or a job running on behalf of the user. So, GSI uses "proxies" for the purposes of authentication and delegation on-behalf of a user. The user's grid identity is mapped into resource-specific user identity based on grid map files. Thus, this mechanism is independent of and can work in tandem with any local security mechanisms the individual resources might employ.

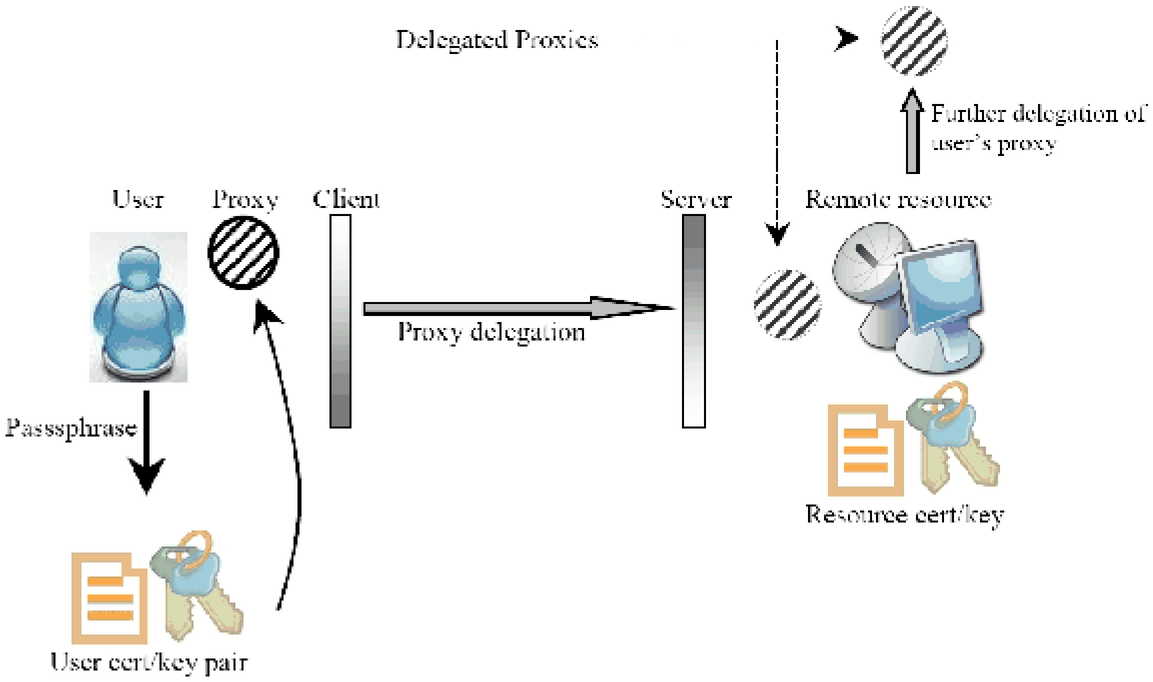

Fig. 1. Certificate-based Authentication and Delegation in GSI

The user's long-term private key is encrypted with his passphrase (similar to password, but deliberately longer in size to prevent dictionary-based password guess attacks) to protect against compromised keys. To sign on to a grid resource, he can create a proxy (with a few hours of lifetime) and signing it using his private key. This proxy is presented by the user to remote resources for authentication and possibly further delegation. For increased security and to protect against exploitation of the user's proxy by a malicious resource, the user can choose to restrict the proxies from any delegation, thereby disallowing further remote access. However, there is currently no support in

GSI to protect against compromised user proxies. Typically, the proxy credential acceptor is a remote host (that is expected either to spawn a computation on behalf of the user or to perform a data transfer operation).

Any resource with the delegated user proxy can now make additional resource requests on behalf of the user. Figure 1 illustrates the authentication and delegation process used by GSI in order to achieve single sign-on and mutual authentication between elements of a grid. In the figure 1, the grid middleware components, such as Globus resource manager (GRAM) [13] or GridFTP [2] , on the user-side and resource-side are respectively indicated as client and server.

IV. SYSTEM ARCHITECURE

The architecture of our grid authentication model allows instantaneous revocation capabilities and simplified credential management. The major entities involved are the user (U), his certifier (CA_u), a grid resource (R) and its certifier (CA_R). The user and resource belong to certain organizations or grid communities (VO). Each VO's SEM and mRSA signature operations are handled by a daemon, what we call Grid Security Mediator (GSM). This is to eliminate the need for CA to remain online and answer credential status queries. In the setup shown in figure 2, we have two mediators GSM_u and GSM_R, respectively for the user's and resource's VO. The GSMs generate their standard private keys, submit the public keys with required information to the respective CAs and obtain certificates.

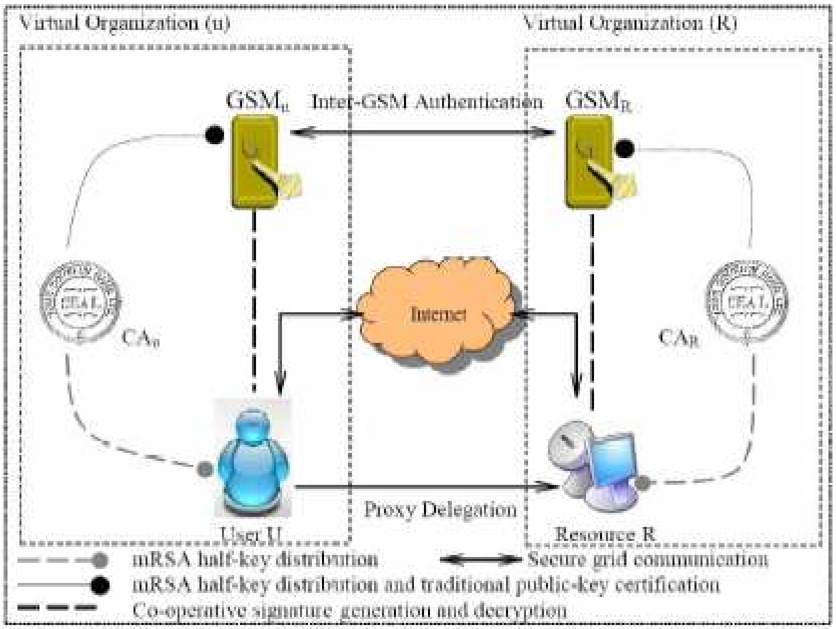
Virtual Organization (u)
Virtual Organization (R)
GSM_u
Inter-GSM Authentication
GSM_R
Internet
CA_u
CA_R
Proxy Delegation
User U
Resource R
mRSA half-key distribution
Secure grid communication
mRSA half-key distribution and traditional public-key certification
Co-operative signature generation and decryption

Fig. 2. Architecture of the Authentication Model

As with GSI, the user and the resource, collectively termed as clients, need to obtain their respective certification from the CAs. However, the clients do not generate the keys themselves. But, the CAs generate a set of simple 2-by-2 threshold keys du and d{sem-u} or dR and d{sem-R}) based on mediated RSA for its clients, as indicated in the algorithms above. These half-keys are communicated to the intended recipients and this does not need any *secure channels*.

We briefly discuss the generation of the mRSA keys. For each requestor, the CA generates a unique set: {p, q, e, d, d{gsm}, du} where p and q are large primes and n = pq and ɸ(n) = (p-1)(q-1). e is a random number prime to \phi(n), that is, g.c.d(e, ɸ (n)) = 1. d is the multiplicative inverse of e modulo ɸ (n), that is, d = e^{-1} mod ɸ (n).

Now, (n,e) forms the public enciphering key. Now, d{gsm} is a random integer in [1,n] and du = d - d{gsm} mod ф (n).

The mRSA keys intended for the GSMs, d{gsm}, (that now act as SEMs) can be relayed to them after encryption with their certified public keys and signed with CA's private key. The half-key of the user, du, can be optionally encrypted with a password chosen by the client as part of the key request process. After this initial key distribution, the client and its SEM has to co-operate by using their respective half-keys to complete any signature or decryption operations. As already noted this process is transparent to any entity performing verifying signatures from or encrypting to the client.

Inter-GSM Authentication: In a typical grid collaboration, the number of VO communities, and hence the number of associated GSMs, is much smaller than that of the users and resources they serve. The GSM in our model provides the SEM functionalities to the VO. Also, they act as the "directories" required for NOVOMODO operations, that is, they hold the validity and revocation targets for the user proxies. By securely interacting with another VO's GSM, they realize the objectives of freshness guarantees for user/resource credentials. For this purpose, "trust" should be established between the GSMs. The GSMs are certified by their respective CAs and it is a straight-forward task to setup "Inter-GSM trust", for instance, using traditional certificate-based handshake approach. However, the purpose of this "trust" setup is not extensive in scope as a Kerberos inter-realm setup. Its sole purpose is to confirm the identities of the two GSMs to each other.

After verification, the GSMs are configured to accept the credentials issued by each other's CAs. It will help to note that this process is similar to setting up a grid resource to trust a user's CA in classic GSI model. Optionally, it is trivial to achieve authentication between the GSMs and their respective clients (users/resources) by allowing the clients to choose a password as part of the setup process.

Protocol for Grid Resource Authentication: Once inter-GSM authentication setup is completed, the grid entities can now interact across VO boundaries. This authentication process proceeds as follows. Whenever a user wants to use a remote resource (R), for example, to make a job submission, the user's client program (P) queries the local GSM for the target resource's public key (PKR). For first-time communication with R, the GSM will not have PKR available locally and hence obtain it from target VO's authenticated GSM. Once the key is obtained, the local GSM caches it for future use and includes in its reply to P as well. P then sends a randomly generated message (M) to R encrypted with PKR, that is, C = Encrypt{PKR}(M). If R can successfully decrypt C and communicate M to P, implicitly R stands authenticated. This is because R possesses only half of the private key corresponding to PKR and cannot perform decryption without co-operation from GSMR. If R's identity has been revoked, GSMR will not exercise its half of R's private key and decryption will not be possible for R. The mere ability of R to decrypt C implies the validity of R's identity at the time the decryption operation was performed.

Protocol for Grid User Authentication: To complete mutual authentication, the user U now has to prove his identity to the authenticated resource R. This operation is very similar to conventional GSI. As indicated in {rivest}, it is always the credential acceptor that runs the risk of accepting stale credentials and, thus, should have the ability to set the recency requirements for the credential on behalf of U. This is also relevant to check for the revocation of U's certificate. R can verify that U's certificate CU is issued by a trusted CA. R can obtain CU from some online service or it can be sent by U himself as part of the authentication request. R now generates a secret (S and this later acts a shared secret between U and R) and encrypts it with the public key of U as indicated in Cu and sends it to U. U, if valid, can decrypt this message to get the original secret S by sorting co-operation from its local VO's GSM. Decryption operation will succeed only if the GSM and U co-operate. To confirm its validity status, U sends an OK message to R encrypted with S. R can repeat the secret-key operation on this message to obtain the OK message and this implicitly confirms U's valid identity to R. If U's credentials have been revoked, GSMu will not exercise its half of U's mRSA private key. Hence U cannot decrypt to get S in the previous step. Optionally, once this protocol completes, S can now be used as a shared secret between R and U to encrypt their communications. This also gives the added benefit of secrecy to U-R communication free from eavesdropping.

User Privilege Delegation Model: Usage scenarios might warrant delegation of user's privileges to processes on remote resources. For instance, grid applications tend to generate dynamic resource requests or acquire newly available resources during the course of computation progress for reasons of performance or fault-tolerance. Globus proxies were designed to allow user identity and privilege delegation to support this model. In our system, the user proxy generation is extended to include support for revocation. When the proxy is created, the proxy creator makes use of NOVOMODO approach to indicate to the acceptor whether the proxy is fresh or stale. As with NOVOMODO, the proxy certificate is enhanced with the validity and revocation target values. Depending upon the intended lifetime of the proxy, the values of Xis are calculated and X and Y values are included as part of the proxy certificate extensions. This structure is ensured to comply with the proxy certificate profile specifications {proxy}. The functionality of a NOVOMODO "directory" is implemented as part of the GSMs and the GSM details are included as part of the proxy certificate. That is, for a proxy with lifetime of n periods, C{proxy} = SignU(PK{proxy}, K{proxy}, d1, d2, Y1, Xn, GSMu). Now, with a delegated proxy, the acceptor (as mandated by its recency requirements) can query the user's GSM for revocation status. For a period i, the presence of X{n-i} or Y0 in GSM's reply indicates the valid or revoked status of U's proxy respectively.

We argue that this modification is mandatory because, inspite of the proxies' limited lifetime, the scale of accessible resources with a full proxy on a grid raises serious concerns. Further, compromised grid resources could act maliciously by exploiting user proxies delegated to them. Further, user authorization on a grid is granted solely based upon a valid proxy and hence proxy compromise proves to be a serious threat.

Implementation and Discussion: Currently, our implementation of GSM is in C and uses the SEM libraries available as part of {sucses}. We have completed an implementation of the grid proxy creation program that adds NOVOMODO-like target values to proxy certificate extensions. Also, this client's interface to GSM for target value distribution is functioning. Prototype modifications to GRAM and GridFTP servers have been completed to verify proxy certificate validity from a GSM daemon. Complete software information is available publicly.

By using mRSA, we inherit the benefits of binding signature semantics in grids. That is, a signature's validity is equivalent to checking the public key's validity at the time the signature was generated. This could prove to be beneficial to grid accounting systems in guaranteeing non-repudiation. Also, our system is quite easy to setup and the administrative overhead involved with continued operation is trivial. For the end-users, no setup procedures or security configuration is required to use a grid. Our model could completely eliminate the relatively cumbersome process in existing grid security. The simplified credential management in our model can aid developing simpler co-allocation tools across VOs with multiple CAs and complex trust relations. Inherently, our system eliminates "key escrow" problem because no single entity possesses the entire private key for grid users/resources and hence cannot decrypt communications. Additionally, it is trivial to achieve encrypted grid communication in our system. This is aided by the establishment of a shared secret at the end of mutual authentication protocol between grid resources and users. Caching public keys of the resources at various GSMs helps in reducing communication between GSMs over time. Optionally, per recency requirements, a client can make the associated GSM to request a fresh copy of a chosen resource's public key. For handling the revocation of the GSMs certificates, approaches involving IBE extensions can be used and we plan to address this as part of our future work.

V. SUMMARY

Certificate validation and revocation is universally recognized as a crucial problem. We presented an authentication system that solves the revocation issues in grid environments. The main idea is to employ mRSA approach to handle the identities of the grid users and resources. Also, user proxies are enhanced to contain revocation information using aspects of the NOVOMODO scheme. This allows for instantaneous revocation of both long-term digital identities of hosts/users and short-lived identities of user proxies. We introduced a SEM-type mediator for virtual organizations. With this approach, the users enjoy a simplified view and usage of grid security.

REFERENCES

[1] Z. Adelman and M. Houyoux, Processing the National Emissions Inventory 96 (NEI96) version 3.11 with SMOKE. The Emission Inventory Conference: One Atmosphere, One Inventory, Many Challenges, 1-3 May, Denver, CO, U.S. Environmental Protection Agency, 2001.

[2] W. Allcock, J. Bester, J. Bresnahan, A. Chervenak, I. Foster, C. Kesselman, S. Meder, V. Nefedova, D. Quesnel, and S. Tuecke, Data Management and Transfer in High-Performance Computational Grid Environments. Parallel Computing 2001.

[3] G. Appenzeller and B. Lynn, Minimal Overhead IP Security using Identity-Based Encryption. Submitted for Publication, Available at http://rooster.stanford.edu/~ben/pubs/

[4] D. Boneh, X. Ding and G. Tsudik, Fine-Grained Control of Security Capabilities. ACM Tranactions on Internet Technology, Vol.4, No. 1, pages 60-82, February 2004.

[5] D. Boneh and M. Franklin, Identity-Based Encryption from the Weil Pairing. In Proc. of Crypto 2001, LNCS 2139, pages 213-229. Springer-Verlag, 2001.

[6] D. Boneh and M. Franklin, Identity-Based Encryption from the Weil Pairing. Cryptology ePrint Archive: Report 2001/090, 2001. http://eprint.iacr.org.

[7] M. Burmester, Y.G. Desmedt, Is Hierarchical Public-Key Certification the Next Target for Hackers?" Communications of the ACM August 2004/ Vol. 47, No. 8, 2004.

[8] R. Butler, D. Engert, I. Foster, C. Kesselman, S. Tuecke, J. Volmer, V. Welch, A National-Scale Authentication Infrastructure. IEEE Computer, 2000.

[9] D. W. Byun, J. Pleim, R. Tang, and A. Bourgeois,1999: Meteorology-Chemistry Interface Processor (MCIP) for Models-3 Community Multiscale Air Quality (CMAQ) Modeling System. Washington, DC, U.S. Environmental Protection Agency, Office of Research and Development.

[10] D. W. Byun, K. Schere, EPA's Third Generation Air Quality Modeling System: Description of the Models-3 Community Multiscale Air Quality (CMAQ) Model. Journal of Mech. Review, 2004

[11] B.M. Chapman, Y. Li and B. Sundaram, and J. He, Computational Environment for Air Quality Modeling in Texas. Use of High Performance Computing in Meteorology, World Scientific Publishing Co, 2003

[12] B.M. Chapman, H. Donepudi, J. He, Y. Li, P. Raghunath, B. Sundaram and Y.Yan, Grid Environment with Web-Based Portal Access for Air Quality Modeling. Parallel and Distributed Scientific and Engineering Computing, Practice and Experience, 2003

[13] K. Czajkowski, I. Foster, N. Karonis, C. Kesselman, S. Martin, W. Smith, and S. Tuecke, A Resource Management Architecture for Metacomputing Systems. Proc. IPPS/SPDP '98 Workshop on Job Scheduling Strategies for Parallel Processing, pg. 62-82, 1998.

[14] K. Czajkowski, I. Foster, and C. Kesselman, Resource Co-Allocation in Computational Grids. Proceedings of the Eighth IEEE International Symposium on High Performance Distributed Computing (HPDC-8), pp. 219-228, 1999.

[15] W. F. Dabberdt, M. A. Carroll, D. Baumgardner, G. Carmichael, R. Cohen, T. Dye, J. Ellis, G. Grell, S. Grimmond, S. Hanna, J. Irwin, B. Lamb, S. Madronich, J. McQueen, J. Meagher, T. Odman, J. Pleim, H. P. Schmid, D. L. Westphal, Meteorological research needs for improved air quality forecasting. Report of the 11th Prospectus Development Team of the U.S. Weather Research Program., Bull. Amer. Meteor. Soc., 85, 563-585. 2004.

[16] I. Foster and C. Kesselman, Globus: A metacomputing infrastructure toolkit. International Journal of Supercomputer Applications, Summer 1997.

[17] I. Foster and C. Kesselman, The GRID: Blueprint for a new Computing Infrastructure. Morgan Kauffman Publishers, 1999.

[18] I. Foster, C. Kesselman, The Globus Project: A Status Report. Proc. IPPS/SPDP '98 Heterogeneous Computing Workshop, pp. 4-18, 1998.

[19] I. Foster, C. Kesselman, G. Tsudik, S. Tuecke A Security Architecture for Computational Grids. ACM Conference on Computers and Security, 1998, 83-91.

[20] I. Foster, C. Kesselman, S. Tuecke, The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International Journal of High Performance Computing Applications, 15 (3). 200-222. 2001.

[21] P. Gemmel, An Introduction to Threshold Cryptography. RSA Cryptobytes 2, 7.

[22] C. Gentry, Certificate-based Encryption and the Certificate Revocation Problem. Cryptology ePrint Archive: Report 2003/183, 2003. http://eprint.iacr.org.

[23] G. Grell, J. Dudhia, and D. Stauffer, A Description of the Fifth-Generation Penn State/NCAR Mesoscale Model (MM5) NCAR/TN-398+STR. NCAR Tech Notes http://www.mmm.ucar.edu/mm5/

[24] M. R. Houyoux, J. M. Vukovich, C. J. Coats, Jr., N. W. Wheeler, and P. S. Kasibhatla, Emission inventory development and processing for the seasonal model for regional air quality (SMRAQ) project. J. Geophys. Res., Atmospheres, 105, D7, 9079-9090. 2000.

[25] A. Joux, A One Round Protocol for Tripartite Diffie-Hellman. Proc. of Fourth Algorithmic Number Theory Symposium, Lecture Notes in Computer Science, Vol. 1838, Springer-Verlag, pages 385-394, 2000.

[26] N. Koblitz, A Course in Number Theory and Cryptography. Series: Graduate Texts in Mathematics, Vol. 114, second

edition, Springer-Verlag, 1994.

[27] L. M. Kohnfelder, Towards a Practical Public-Key Cryptosystem. B.S. Thesis, supervised by L. Adleman, MIT, May 1978.

[28] B. Lynn, Authenticated Identity-Based Encryption. Cryptology ePrint Archive: Report 2002/072, 2002. http://eprint.iacr.org

[29] S. Micali, Novomodo: Scalable Certificate Revocation and Simplified PKI Management. In Proc. of 1st Annual PKI Research Workshop 2002, available at http://www.wisdom.weizmann.ac.il/~kobbi/papers.html

[30] J. W. Nielsen-Gammon, Initial Modeling of the August 2000 Houston-Galveston Ozone Episode. A Report to the Technical Analysis Division, Texas Natural Resource Conservation Commission, December 19, 2001

[31] J. Novotny, S. Tuecke, V. Welch, An Online Credential Repository for the Grid: MyProxy. Proceedings of the Tenth International Symposium on High Performance Distributed Computing, pages 104-111, IEEE Press, August 2001.

[32] R. L. Rivest, Can We Eliminate Certificate Revocation Lists? in Financial Cryptography, Rafael Hirschfield, Ed., Anguilla, British West Indies, February 1998, vol. 1465, pages 178-183, Springer Verlag 1998.

[33] R. Rivest, A. Shamir, A. Adleman, A Method for Obtaining Digital Signatures and Public-Key Cyptosystems. Communications of the ACM 21, February 1978, pages 120-126, 1978.

[34] B. Schneier, Applied Cryptography. John Wiley & Sons, second edition, 1996.

[35] A. Shamir, Identity-based Cryptosystems and Signature Schemes. In Proc. of Crypto 1984, LNCS 196, pages 47-53, Springer-Verlag, 1985.

[36] J. Steiner, B. C. Neuman, J. Schiller,Kerberos: An Authentication system for Open Network Systems. Proceedings of Usenix Conference, 1988, 191-202.

[37] B. Sundaram, C. Nebergall, S. Tuecke, Policy Specification and Restricted Delegation in Globus Proxies, Poster presented in SuperComputing Conference 2000, SC2000, 2000.

[38] S. Tuecke, D. Engert, I. Foster, M. Thompson, L. Pearlman, L. C. Kesselman, Internet X.509 Public Key Infrastructure Proxy Certificate Profile. IETF Draft draft-ietfpkix-proxy-06.txt, 2003.

[39] J. Vukovich, J. McHenry, C. Coats and A. Trayanov, Supporting Real-Time Air Quality Forecasting using the SMOKE modeling system. Denver, CO, EPA Emissions Inventory Conference, April 30-May 2, 2001

[40] Air Quality Modeling Project, University of Houston, http://www.math.uh.edu/aqm

[41] Comprehensive Air quality Model with extensions (CAMx): http://www.camx.com/, 2003

[42] Environmental Protection Agency, http://www.epa.gov

[43] The HPCTools Group, Department of Computer Science, University of Houston.

[44] National Centers for Environmental Prediction (NCEP), http://www.ncep.noaa.gov/

[45] Public Key Infrastructure, Final Report; MITRE Corporation; National Insitute of Standards and Technology, 1994.

[46] Public Key Infrastructure Standards, http://csrc.nist.gov/pki/panel/warwick

[47] PURSe: Portal-Based User Registration Service, http://www.grids-center.org/solutions/purse/

[48] Secure Sockets Layer Specification 3.0, http://www.netscape.com/eng/ssl3

[49] The SUCSES Project, http://sconce.ics.uci.edu/sucses/

[50] Texas Commission on Environmental Quality (TCEQ), http://www.tceq.state.tx.us/index.html

[51] Web Services - Resource Framework, Specifications of the WS-Resource construct, http://www.globus.org/wsrf/specs/ws-wsrf.pdf

[52] Weather Research and Forcasting Model (WRF), http://wrf-model.org

[53] X-509 Certificate Format, http://www.w3.org/PICS/DSig/X509_1_0.html

[54] X.509 Internet Public Key Infrastructure Online Certificate Status Protocol (OCSP), IETF RFC 2560, http://www.ietf.org/rfc/rfc2560.txt

[55] X.509 Internet Public Key Infrastructure Certificate and CRL Profile , IETF RFC 2459, http://www.ietf.org/rfc/rfc2459.txt