GFD-I.17
Grid Certificate Policy Working Group

Mary R. Thompson, LBNL
Doug Olson, LBNL
Robert Cowles, SLAC
Shawn Mullen, IBM
Mike Helm, ESnet
June 2003

CA-based Trust Issues for Grid Authentication and Identity Delegation

Status of This Memo

This memo provides information to the Grid community regarding trust issues involving reliance on certificate authorities and X509 identity certificates to authenticate people and services. Distribution is unlimited.

Copyright Notice

Abstract

This paper examines the basis for the trust a relying party places in an identity certificate signed by a certificate authority (CA). It is an informational document resulting from extensive discussion among the members of the security working group about current and best public key infrastructure authentication practices. It does not cover authorization issues or trust relationships between the CA and its registration authorities or between the end entities and the resource providers.

Table of Contents

Status of This Memo....1
Copyright Notice....1
Abstract....1
1 Background....2
2 Grid Environment....2
2.1 Grid Requirements for Authentication....2
2.2 Using PKI for Authentication....2
2.3 "Trusting" an Identity Certificate....3
3 Role of a Grid CA....3
3.1 Elements of Distinguished Names....4
3.2 Identity Vetting....4
3.3 Significance of the Subject Name....4
3.4 Virtual Organization Authority vs. Grid CA....4
3.5 Cross-Signing and Subordinate CAs....5
3.6 Kerberos CA....5
4 Delegation of Identity....5
5 Responsibility of the Subscribers....6
6 Technology for Protection of Private Keys....6
6.1 FIPS PUB 140-2....6
6.2 Smart Cards, PKCS 11, PKCS 15....7
6.3 Online Credential Server....8
6.4 Offline CA....8
7 Conclusions....8
Security Considerations....9
Author Information....9
Intellectual Property Statement....9
Full Copyright Notice....9
References....9

1 Background

This paper was suggested at the Fourth Global Grid Forum (GGF4) after it was determined that the GGF mission would not include responsibility (and thereby liability) to either vet certificate policies (CPs) or audit certification authorities (CAs). The paper focuses on the issues faced by a relying party when deciding to trust an unaudited Grid CA and the certificates that it issues.

The scope of this paper is limited to examining the basis for the trust a relying party places in an identity certificate signed by a CA. Also presented are some of the working group's discussion of the authentication needs of a Grid and the ways in which use of public key infrastructure (PKI) identity certificates can fulfill those needs. The paper does not cover issues such as using these certificates to authorize use of resources, trust relationships between the CA and its registration authorities (RAs), or trust relationships between the end entities and the resource providers.

2 Grid Environment

A computational Grid has been defined as "a hardware and software infrastructure that provides dependable, consistent, pervasive and inexpensive access to high-end computational capabilities" [8]. Typically, Grid resources are provided by various organizations and are used by people from diverse sets of organizations. A Grid may support (or define) a single virtual organization, or it may be used by more than one virtual organization. Individual pieces of hardware may be used in more than one Grid, and people may be members of more than one virtual organization. The different resources in a Grid may have different access policies, including how they authenticate and authorize users. If no common or overlapping authorizations exist among the resources, however, they do not form a usable Grid. Thus, in this paper we will assume that for any set of resources there are some common authentication or authorization procedures.

2.1 Grid Requirements for Authentication

Users, hosts, and services need to be able to authenticate themselves in the Grid environment. Experience in using Grids for remote computations has demonstrated the need for unattended user authentication in addition to interactive authentication. Unattended authentication of users is needed (1) when a user is making frequent requests to remote servers and does not want to repeatedly type in a passphrase and (2) when a long-running job may need to authenticate itself after the user has left. Servers specific to a single host may need to be started at system boot time and run with their own or the host's identity. Some services may need to be started periodically on many different hosts and be able to authenticate themselves with a known identity.

Basically, authentication between two entities on remote Grid nodes means that each party establishes a level of trust in the identity of the other party. In practical use an authentication protocol sets up a secure communication channel between the authenticated parties, so that subsequent messages can be sent without repeated authentication steps, although it is possible to authenticate every message. The identity of an entity is typically some token or name that uniquely identifies the entity.

2.2 Using PKI for Authentication

One commonly used identity token is provided by an X.509 public key certificate. Such a public key certificate is defined in the I-TUT standard [12] as "the public key of a user, together with some other information, rendered unforgeable by encipherment with the private key of the certification authority which issued it" [Section 3.3.44].

In practice, an X.509 certificate, as defined by the PKIX X.509 Certificate Profile RFC [10], contains a public key, a subject name in the form of a multicomponent distinguished name (DN), and a validity period and is signed by a trusted third party, or certification authority (CA). While the standards documents refer to these certificates as "public key certificates" or "X.509 certificates," this document also uses the term "identity certificates" to emphasize their use to securely identify an entity in a Grid environment. The X.509 certificates are used with the TLS (Transport Level Security, i.e. SSL) security protocol [5] to make a secure authenticated connection between two parties. X.509 certificates are exchanged between entities, usually in a phase where no security has yet been negotiated (no data integrity or confidentiality). The certificates are first tested for validity by checking the expiration dates, possible revocation, acceptable key usage, and signature by a trusted CA. If the certificates pass all these checks, their public keys are then used to build a challenge handshake to prove that each entity that sent a

certificate has the corresponding private key. Passing these tests gives each party a level of confidence that it has established a secure connection to the party represented by the certificate presented. The identity token could be the public key, which is unique by virtue of being a member of a large sparse space, or the certificate subject name, which should be unique within the certificates issued by a single CA. Typically the subject name is used rather than the public key because it is human readable and can be expected to name the individual or service provided in some reasonable manner.

Other public and private key-based schemes include PGP keys [9], SSH keys [1], and SPKI [7] keys and protocols. SPKI focuses on authorization certificates more than identity certificates. SSH is primarily a private/public key mapping with no real attempt to provide global names. The PKIX/X.509 scheme has a small set of trusted third parties (CAs) to sign identity certificates that contain a subscriber's public key. This improves the scaling properties of public key distribution in that only the CA's public key needs to be distributed in an out-of-band secure manner. In systems without a trusted third party, such as PGP, each key holder must find some secure way of establishing the association of his identity with his public key, to each party with which he wishes to establish authenticated communication. In the X.509 infrastructure, the individual subscriber’s public key can be transmitted in a public key certificate as part of a TLS connection handshake and can be accepted as valid if the certificate is signed by a trusted CA.

Another feature of the X.509-TLS infrastructure is that it supports multiple independent CAs. In a Grid each site may chose which CAs it will accept for binding DNs and public keys. Most of the current Grid tools are built on GSI or https, both of which use X.509 certificates for securely establishing a Grid identity.

2.3 “Trusting” an Identity Certificate

Simply put, the only assurances provided by using TLS with mutual authentication are the following:

- Every time this public key or DN is presented through the secure protocol, the same private key is known on the other end of the connection.
- If knowledge of the private key is assumed to be restricted to the entity that the CA originally associated with the DN, then the party on the other end of the connection must be the entity named by the DN.

These assurances depend on the correctness of the TLS and certificate validation implementation at all the sites that take part in establishing a secure connection, the diligence of the individual in protecting the private key, and the certificate policy (CP) and certification practice statement (CPS) of the trusted CAs. In order for a relying party (e.g., a Grid resource provider) to trust identity certificates signed by a given CA, it must trust that:

- Only the CA has access to the CA's private key, so that no other entity can sign certificates with that key (see Section 6).
- The CA always associates each DN and public key with the same individual or entity. That is, the DN and the public key are permanently assigned to this entity (even after revocation or expiration, they cannot be reused by another entity).
- The CA did an acceptable job of verifying that the holder of the certificate is the individual named by the DN (discussed further in Section 3).
- Only the named entity associated with the DN has control of the certificate's private key. What this means for nonhuman entities must be carefully defined by policy (discussed in Section 7).

The CP and CPS documents of each CA define how these objectives will be achieved. If the CA has been independently audited, then a relying party can trust that the auditor agrees that the CA is correctly implementing its CP. If the CA is unaudited, the relying party must trust that the operators of the CA are, in fact, following their avowed policy. The CP should spell out the obligations of a certificate holder to keep the private key secure; but both the CA and the relying party must trust the user to follow the procedures. If each of these assertions is true, then the relying party can trust that the person or other entity that has connected to it through a secure protocol is the unique entity that is named in the X.509 certificate. Thus, the certificate can be used to make a secure and authenticated connection.

3 Role of a Grid Certificate Authority (CA)

A Grid CA is defined as a CA that is independent of any single organization and whose purpose is to sign

certificates for individuals who may be allowed access to the Grid resources, hosts or services running on a single host. Typically, a Grid CA will only sign certificates for these end entities and not for subordinate CAs. A Grid CA is substantially different from a traditional organizational CA, which signs certificates only for members of its organization and is closely linked with the authority that defines who those members are. Those certificates are then used to access resources within the organization. There are two implications of this difference: one in the format of the DNs and the other in the methods of vetting user identification.

3.1 Elements of Distinguished Names (DN)

In identity certificates issued by an organizational CA, the DN often contains a number of attributes taken from the organization's X.500 or LDAP directory (e.g., organizational unit, location, and email). Often, an underlying assumption is that the X.509 certificate is stored in the directory entry for the user. An organizational CA is in the position to find existing LDAP entries, verify the correctness of the name elements, issue certificates for such a user, and store the certificate back in the LDAP entry. As a result of this paradigm a DN could have several vetted components. A Grid CA breaks this paradigm by being independent of its subscribers. Even in the organizational environment there are often problems related to putting too much information in a DN, since whenever any part of the information changes (e.g., an employee changes departments or gets a new email address), the certificate must be reissued.

Since a Grid CA is independent of the organizations to which its subscribers belong, it does not have a way to verify much information about a subscriber or to know when such information changes. The prudent approach for a Grid CA is to put as little information in the certificate as possible. A minimal set that has been chosen by several Grid projects [2, 6, 11, 16] is:

- an organization element that identifies the Grid to which the CA belongs,
- a class designator that identifies the certificate as representing a person, host, or service, which is intended to be used when storing and retrieving certificates in the Grid CA's publishing directory, and
- a common name that reasonably identifies the entity for which the certificate is issued.

An email address can be added as an alternative name for the sake of convenience, but not for identification.

3.2 Identity Vetting

Since the operator of a Grid CA does not personally know the persons who are requesting certificates and does not have access to a trusted directory of such users, it must rely on registration agents (RAs). These are individuals who are likely to know a subset of subscribers firsthand or secondhand. If the users of a Grid can be grouped by actual or virtual organizations, an RA may be chosen for each such organization and given the responsibility to approve requests from members of that organization only. The rules for establishing member identities should be published by each RA, and the procedures for verifying the identities and certificate requests should be consistent among all the RAs and approved by the CA.

3.3 Significance of the Subject Name

A topic of much discussion is the meaning ascribed to the subject name. On the one hand, most CAs specify that the common name component of the subject name should be an official and recognized name for the person who requested the certificate and the identity vetting process should assure this. On the other hand, the name should be treated by a relying party simply as an identity token that can be used to assure that the entity making the current connection is the same entity that has used this token before. In either case, before the name is added to any lists that authorize access to resources, the name must be checked by the authorizing party against some other database or virtual organization authority to see what rights should be allowed for the holder of this certificate. Using only the subject name for authorization is not safe, because subject names are guaranteed to be unique only within the domain of a single CA. Hence, either both the subject name and the issuer (CA) name must be used, or some other means must be used to limit what namespaces may be signed for by which CAs.

3.4 Virtual Organization Authority vs. Grid CA

Another subject of discussion is the role of a virtual organization (VO) as a trusted third party. If an entity is authorized to use resources because it is a member of a VO, the relying party needs to verify that a token belongs to a member of the VO. In this case, it might be more efficient to have the VO issue the certificates in the first place. In this case any entity that holds a certificate from the VO could be assumed to be a member of that VO. This approach has two drawbacks, however. First, since not every user of Grid resources is a member of an accepted VO, having

VO CAs does not eliminate the need for a broader Grid CA. Second, some persons are members of more than one VO, and they would end up with a certificate from each VO. This has the advantage that it would allow a user to act in different capacities (or roles) by using different identity certificates. On the other hand, managing different certificates is also a burden on the user, especially in view of the primitive tools available for certificate handling. A more serious objection is the merging of authorization into the concept of identity and authentication. The consensus is that X.509 certificates should be used purely for authentication of identity and that authorization should be handled as a separate issue.

3.5 Cross-Signing and Subordinate CAs

A relying party may want to independently evaluate each CA that it is going to trust, before installing its certificate. Alternatively, a CA may sign another CA certificate once it has verified that the other CA enforces policies that are as secure as its own. A subordinate CA is similar except that it has a CA certificate signed only by the parent CA, whereas a cross-signed CA has either a self-signed CA certificate or a CA certificate signed by another CA. The advantage of this alternative trust model is that a relying party can accept subscribers from more CAs without having to individually evaluate and install all the CA certificates. A relying party must know, however, whether a trusted CA signs for other CAs' certificates. Otherwise, the relying party may be inadvertently widening its trust domain. Note that not all path validation software recognizes CA chains but may insist on having the certificates for every trusted CA.

3.6 Kerberos CA

Another solution to facilitate user access to the Grid is to run a CA that will create a short-term X.509 credential based on a Kerberos identity. Thus, Kerberos users can transparently use Grid resources without having to apply for a long-term Grid identity. A relying party should be aware if a trusted Grid CA either cross-signs or is a parent of a Kerberos CA, in order to decide whether this model of user identification is acceptable. The relying party is heavily dependent on the quality of the Kerberos key distribution center's security. The automatic issuance of certificates requires that this CA be online at all times.

4 Delegation of Identity

As was mentioned in Section 2.1, Grid experience has shown the need for unattended authorization. It has also demonstrated the need for both local and remote processes to run on behalf of a user and with his authorization rights. These issues have been addressed by the Globus Toolkit® Grid Security Infrastructure (GSI) by allowing for short-term proxy certificates, stored with unencrypted private keys, to which a user has delegated his identity. These certificates are correctly formatted X.509 certificates, except that they are marked as proxy certificates and are signed by an end entity rather than a CA. The choice of the lifetime of proxy certificates requires a compromise between allowing long-term jobs to continue to run as authenticated entities and the need to limit the damage that might be done in the event that a proxy is stolen. Proxy certificates with restricted rights are another way of limiting the damage done by a stolen proxy.

Authorization software run by relying parties must be able to recognize proxy certificates and search the certificate chain until the end-entity certificate is found in order to do the authorization based on that identity token. Such software may also want to enforce policy decisions based on the lifetime of the proxy or on the number of levels of delegation that have been done. While restriction of proxy rights may make a site more secure, it will likely break some Grid software attempting to run at that site.

The delegation of credentials may take place on the machine on which the original credential resides, or may take place between two machines in different administrative domains. In the latter case, the delegation expands the trust relationships to include an additional domain (and the delegation software that runs there). The relying party should be aware this situation; but in the absence of secure DNS, it is difficult to include trusted domain name information in a certificate chain. The Grid Forum document on proxy certificates [15] discusses this matter in greater depth. Although several schemes for including trace delegation information in the proxy delegation chain were discussed in the preparation of the X.509 proxy extension document 15, no standard was agreed upon. At the current time, only the number of times a proxy has been delegated can be deduced from the chain of delegated proxies.

5 Responsibility of the Subscribers

In the trust model discussed above, the subscriber is responsible for keeping complete control over the private key. If there is any evidence that the private key has been compromised, the certificate should be revoked. In the case of certificates issued to people, if the holder of the certificate believes that the certificate has been compromised then the holder is responsible for ensuring that the certificate is revoked as quickly as possible. Any actions taken with a stolen key will be attributed to the subscriber. In addition, if an RA is presented with sufficient evidence that a key has been compromised, he should revoke the certificate. For certificates issued to hosts or services, however, it is not so obvious who is responsible for the key. A CA that issues such certificates should make a clear statement of what the host or service certificate is supposed to represent and how the private key is to be protected. For example, a CP may contain the following requirements for subscribers

- For a personal certificate
  - Only the named user has control of the certificate's private key
  - When the key is stored on a computer, it is always encrypted with a passphrase at least as strong as commonly accepted guidelines for account passwords. Preferably, the passphrase is significantly longer that an eight character password and is not in any dictionary or common phrase book. A strong passphrase is highly desirable because of the inconvenience of revoking a compromised key and certifying a new one.
- For a host or service certificate that is used only on a single host
  - The key may be stored unencrypted only on local storage of the host named in the certificate
  - It should be used only to represent services run on that host
  - The system administrator of the host is responsible for revoking the certificate if there is any evidence that the host has been compromised.
- For a service certificate that is used on multiple hosts
  - The key may be kept unencrypted on more than one host.
  - The requester of this certificate is responsible for the security of the key on each host.
  - Relying parties must be able to find out who the responsible party is.

6 Technology for Protection of Private Keys

Trust in the user certificates depends on how well the private keys of the CA, the RAs, and the users are protected. There exist several technologies and standards relating to key management as well as hardware and software implementations of these standards. Since these standards or products can be mentioned in CA policy or practices statements, a brief overview is included here. Because CA and RA keys are more limited in number and more critical than user keys, a more secure and expensive means of protection is likely to be required of them. Convenience of use may be a bigger factor in the options recommended for storing subscriber keys. A relying party should be aware of what technologies are required or allowed for key storage.

6.1 FIPS PUB 140-2

FIPS PUB 140-2 (May 2001), "Security Requirements for Cryptographic Modules," is a standard from the National Institute of Standards and Technology (NIST) [14]. The document specifies four different security levels and defines standards for key management (including key generation and storage), physical security of the module, cryptographic algorithms, and security functions that the module should perform. NIST sponsors a testing and certification system; see the NIST Validation List Web page for a list of validated products [17]. Customary usage is to refer to rated modules by the shorthand formula "rated at FIPS-140 level 3."

Several manufacturers—Chrysalis (Luna), nCipher (nFast), Rainbow, Baltimore, and IBM—produce hardware security modules (HSMs), hardware-based cryptographic modules, several of which have been issued FIPS-140 validation certificates. They represent only a few of the products in the NIST validation list. The HSM controls access to the CA signing private key (and other private keys) through a well-known API such as PKCS #11 [4]. The HSM is usually responsible for creating the key pair; depending on FIPS rating and configuration, the HSM may forbid, or severely restrict, export of private keys from the HSM device by limiting or disabling key export functions. The HSM's cryptographic module will use the CA's signing private key to perform operations, such as signing certificate requests or other objects. The HSM will perform these operations through commands issued through the module API (PKCS #11). The CA will never have the private key in plaintext on the computer host. These devices minimize or eliminate the danger to the CA or other server's private keys in the event of a service

compromise, operating system compromise, or breach of CA's physical security. The best of these devices have been evaluated under certain conditions and system configurations by NIST as meeting security level 2 and security level 3 specifications. The FIPS-140 certification process is also applied to other devices and software and may demonstrate a product's level of maturity and quality of implementation of cryptographic algorithms and random number generators.

Security level 1 certification shows that the module executes cryptographic functions in an objectively reasonable way. This is the basic certification for a hardware cryptographic module.

Security level 2 certification requires that the module provide tamper evidence, meaning the physical device will be sealed with a coating or otherwise provide evidence of physical access, should an intruder attempt to read the contents of the module. The device must also implement an operator role, requiring a basic authorization process before executing certain functions.

Security level 3 certification requires the module to be tamper resistant. According to the language of the FIPS-140 standards document, it should have a high probability of detecting and responding to attempts to physically access the module contents. Identity-based authentication is required before functions are executed, and there are classes of functions assigned to different roles. Level 3 modules do not allow the unencrypted export of keys to the host operating system.

Security level 4 certification requires a high degree of assurance that physical access to the module cannot result in a breach of security. Modules should demonstrate a high likelihood of detecting module penetration, and high resistance to environmental compromise (such as operation outside normal temperature and voltage levels). Security level 4 devices might be very useful in a physically unprotected or physically demanding location. There are relatively few level-4 certifications.

Numerous other conditions and capabilities are associated with the different security levels. Commercial HSMs typically are used in conjunction with a smart card system (see Section 6.2), which implements the roles and authentication requirements of the standard. In addition, these smart card services further restrict the capabilities of the service (eliminating or encrypting plain text key uploads and downloads), enhancing the manageability of the CA or other service for the CA administrators and other personnel.

6.2 Smart Cards, PKCS 11, PKCS 15

Smart cards (or integrated circuit cards) are also used to store credentials off-line. The private key is kept only on the smart card, which is inserted into a reader device when it is needed to sign a document or make a secure connection. A number of manufacturers make the cards and readers. Currently there is a widely used standard, RSA PKCS11 "Cryptographic Token Interface Standard" [4] for signing and cryptographic functions. In the current state of the art, each smart card vendor must provide its own PKCS 11 driver. A new standard, PKCS 15, "Cryptographic Token Information Format Standard" [3], seeks to standardize the format in which cryptographic information is stored on smart cards, thus enabling a standard reader to handle different manufacturers' cards. A smart card with a PKCS 11 interface is a possible solution for protecting CA keys, RA keys, or end user keys. The key would be vulnerable only when the smart card was connected to an online computer and unlocked by the personal identification number (PIN). This scheme provides credential portability if every machine on which a user wants to use his credential is equipped with a compatible smart card reader, but makes it impossible to use the credential on any other machine. Therefore, it is unlikely to be required of general Grid users but could be feasible for CA or RA keys.

Smart-card implementations are usually limited in onboard CPU power, memory, and bandwidth between the host computer and the smart-card device. The most serious of these today is the limited amount of onboard memory. In order to store a key pair or two, 16K–32K of RAM is adequate, provided the certificates are reasonably small. But the need to support multiple vendor standards on a single card, store procedures, and possibly use the cards for security services based on other algorithms makes the system designer's job difficult and the security issues complex.

Since the chips have limited capability, the user's cryptographic work must be offloaded to the host computer. Hence, these devices are no substitute for cryptographic accelerators and HSMs. It is also difficult to ensure the integrity of the user's private key, either electronically or physically, although most cards have some tamper-evidence features.

Incorporating biometric tests, at some cost in internal memory and system complexity, can enhance the security provided by PINs.

Smart cards may improve the security of certain roles in the PKI, such as CA administrator, RA administrator, or system administrator. When relying on a smart card for additional security, an administrator should be aware that a secure key storage device, even if it does its own public key operations, may still have unintended data presented to it by compromised host software and may authorize transactions the user does not intend. The best that can be hoped for with a compromised host is that keys stored on the smart card will not also be compromised.

6.3 Online Credential Server

Another suggested technology for storing credentials is an online server. This is a secure server on which a user can store his credential protected by a password. When the credential is needed, the user connects to the server via a server-side protected TLS connection and types a password that allows the credential server to use the stored credential on his behalf. The rationales for such a server are as follows: it could provide a more secure environment than the user's home machine; it would allow people without home machines to use credentials; and it would allow people away from their home machines to use their credentials. The objections to such a server include the fact that it becomes a single point of attack and potential failure and the security of the user's key is only as good as the security of the server and the password that is used to protect the key. The server can, however, vet the user password and refuse one that is too easy to break. In contrast, when the user stores a key on a workstation, there is no central enforcement of the existence or quality of the passphrase.

The trust issues raised by an online credential server are similar to those of the Kerberos CA mentioned in Section 3.6. In each case, a trusted, online server issues X.509 certificates on behalf of a user when authorized by that user presenting a password. If a relying party chooses to trust these certificates, it is trusting the credential server software, the machine that is it running on, and the strength of the passwords used. However, it needs only to trust that the end user can manage to keep a password secure rather than to securely manage an identity credential.

The CP section on subscriber obligations should specify how the user's private key is to be stored. If storage on on-line credential servers is allowed, or specifically excluded, that information should be stated here.

6.4 Offline CA

A certificate authority can be configured such that the signing engine is available only to the CA administrator, on a host that is never available on any network or by any other means except personal use by the administrator. The host should be kept in a locked, secured facility, and the administrators' access and use carefully logged and controlled. Signing requests are conveyed to the offline CA through removable media, and signed certificates and revocation lists published by writing on removable media and transferring this data back to the "public" part of the CA. While wholly dependent on the administrators' behavior and subject to a variety of theoretical attacks, in practice this is a very good security solution for private key protection for a small PKI.

7 Conclusions

Because of the limited trust relationship between the subscribers, relying parties, and a Grid CA, it is strongly recommended that the X.509 certificate by itself be used only for the purposes of authenticating a user or a service. In other words, in the TLS handshake model, an entity presenting a valid certificate from a trusted CA will be able to connect to the relying party's server. Any authorization to use specific resources should be done explicitly by the server supplying the resources or by a virtual organization that verifies the identity of the person holding the private key associated with the certificate. The certificate or the contained DN can be used to identify a user who should be granted access, as in the case when GSI looks for a DN in a site-specific Grid-map-file. However, no authorization should be granted a user simply because he presents a valid X.509 certificate signed by a Grid CA. The same caveat applies to users of services. Such users should check the DN contained in a service certificate to see that it agrees with an expected name, in addition to checking that a trusted CA issued the certificate.

Before accepting a CA as a trusted third party, a site should carefully review the certificate policy and certification practice statement document(s). Items that should be carefully evaluated include how well the CA and RA keys are protected, how thoroughly the subscriber identities are checked by the RAs; what the obligations of the subscribers are to keep their private keys secure; and whether the CA signs subordinate CA certificate or cross-signs for other independent CAs.

Security Considerations

This entire paper is about evaluating the security risks when relying on X.509 credentials for Grid authentication.

Author Information

Mary Thompson, LBNL - mrthompson@lbl.gov
Doug Olsen, LBNL - dloslon@lbl.gov
Robert Cowles, SLAC - rdc@slac.standford.edu
Shawn Mullen, IBM - shawnm@austin.ibm.com
Mike Helm, ESnet - helm@fionn.es.net

Intellectual Property Statement

Full Copyright Notice

References

1. D. Barret and R. Silverman, *SSH: The Secure Shell,* O'Reilly & Associates, 2001
2. CERN CA http://globus.home.cert.ch/globus/ca/
3. "Cryptographic Token Information Format Standard," PKCS 15, http://www.rsasecurity.com/rsalabs/pkcs/index.html
4. "Cryptographic Token Interface Standard," PCKS11, http://www.rsasecurity.com/rsalabs/pkcs/index.html
5. T. Dierks and C. Allen, The TLS Protocol, Version 1 IETF RFC 2246; http://www.ietf.org/rfc/rfc2246.txt
6. DOE Science Grids CA http://www.doegrids.org

7. C. Ellison, SPKI Requirements, IETF RFC 2692 1999, http://www.ietf.org/rfc/rfc2692.txt
8. I. Foster and C. Kesselman, eds., *The Grid, Blueprint for a New Computing Infrastructure,* Morgan Kaufman, 1999
9. S. Garfinkel, *PGP: Pretty Good Privacy,* O'Reilly & Associates, 1994
10. R. Housley, W. Polk, W. Ford, and D. Solo, Internet X.509 Public Key Infrastructure Certificate and CRL Profile <RFC3280>, http://www.ietf.org/rfc/rfc3280.txt
11. INFN http://security.fi.infn.it/CA/
12. ITU-T Recommendation X.509 | ISO/IED 9594-8, "The Directory: Public-Key and Attribute Certificate Frameworks," Draft V4, February 23, 2001
13. J. Novotny, S. Tuecke, and V. Welch, "An Online Credential Repository for the Grid: MyProxy," presented at the Tenth International Symposium on High Performance Distributed Computing (HPDC-10), 2001
14. "Security Requirements for Cryptographic Module,". http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf
15. S. Tuecke, D. Engert, I. Foster, M. Thompson, L. Pearlman, C. Kesselman "Internet X.509 Public Key Infrastructure Proxy Certificate Profile," IETF Internet Draft draft-ieft-pkix-proxy-01.txt
16. UK HEP Testbed CA http://www.gridpp.ac.uk/ca/
17. "Validation Lists for Cryptographic Standards," http://csrc.nist.gov/cryptval/