Securing Collaborative Environments[1]

Deborah Agarwal, Keith Jackson, and Mary Thompson
Lawrence Berkeley National Laboratory
DAAgarwal@lbl.gov, KRJackson@lbl.gov, MRThompson@lbl.gov

Abstract

The diverse set of organizations and software components involved in a typical collaboratory make providing a seamless security solution difficult. In addition, the users need support for a broad range of frequency and locations for access to the collaboratory. A collaboratory security solution needs to be robust enough to ensure that valid participants are not denied access because of its failure. There are many tools that can be applied to the task of securing collaborative environments and these include public key infrastructure, secure sockets layer, Kerberos, virtual and real private networks, grid security infrastructure, and username/password. A combination of these mechanisms can provide effective secure collaboration capabilities. In this paper, we discuss the requirements of typical collaboratories and some proposals for applying various security mechanisms to collaborative environments.

1. Introduction

Scientific experiments today tend to be multi-institutional and often global ventures. The participants in these collaborations are from many types of institutions including national laboratories, universities, and companies. The resources provided to the collaboratory may reside at the same wide variety of organizations. Each of these institutions has its own security infrastructure to identify its users and protect its resources. Collaborative environments need to be designed to allow groups of people from a diverse set of organizations and locations to work together easily and securely.

Users need to be able to easily and securely identify themselves to the other collaboratory users and resources. This authentication needs to work from an office workstation, from home, from a conference, from an airport, or when visiting another institution. While these requirements are common to most distributed computing environments, a collaborative environment has one additional characteristic: it may be built incrementally and in a distributed fashion. This creates a requirement to incrementally build trust between people and to facilitate the granting of access to resources to newly trusted partners by the resource owners.

2. Security Mechanisms

A secure collaborative environment needs to provide mechanisms for authentication (identity of participant), authorization (privileges of participant), privacy (access control for and encryption of sensitive data), and data integrity. There are many tools that can be applied to the task of securing collaborative environments and these include public key infrastructure (PKI), authorization servers, secure sockets layer/transport layer security (SSL/TLS), Kerberos, virtual and real private networks, grid security infrastructure (GSI), and username/password. Each of these tools has its advantages and disadvantages.

2.1 Public Key Infrastructure (PKI)

PKI X.509 identity certificates[1] used with the appropriate transport protocol can be used to provide strong mutual authentication. They have the additional advantage in a collaborative environment of providing an organization-neutral collaboration-wide identity for people and processes. A collaboration can run its own Certificate Authority (CA) to issue certificates or can accept certificates from existing CAs. A major strength of PKI is that the private key and passphrase can be used to authenticate without transmitting them over the network.

The biggest challenge of using such credentials is the management of the private keys and the certificates. Private keys must be kept securely while at the same time be available for on-line use. The private key is usually stored on the local host encrypted by a normally unvetted passphrase supplied by the user each time the private key

[1] This work was supported by the Director, Office of Science, Office of Advanced Computing Research, Mathematical Information and Computing Sciences Division, of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098. See the disclaimer at http://www-library.lbl.gov/disclaimer. LBNL report number LBNL-50427.

is used. In this case, the private key is no more secure than the user's workstation and the passphrase that was chosen. In addition, the burden of having to repeatedly type in a passphrase at every use of the key tempts the user to use a passphrase that is short, non-existent, or stored somewhere for automatic use.

On-line certificate repositories have been proposed as a mechanism for storing long-lived identity certificates on a secure server instead of on the local machine. Smart cards are a hardware-based method of storing PKI identity certificates. The hardware to provide this capability for a large number of users is relatively expensive.

The GSI proxy certificates can be used to enable a single sign-on by creating a short-lived certificate signed by the user's private key that can represent the user. The proxy certificate has a non-encrypted private key and can be created and stored locally or can be stored on a server (myProxy) where it can later be unlocked using a username and password. It can be used to delegate rights to a process. There are a variety of proposed mechanisms for specifying limits on the access privileges of delegated certificates.

An advantage of having a single collaboration-wide user identity is that authorization policies can be based on these credentials. An authorization server such as the Akenti policy-based authorization[3] or the new Globus Community Authorization Server[5] can provide this capability. Both of these systems use X.509 identity certificates (or proxy certificates) to authenticate and identify users. The user's access rights are contained in a signed capability certificate (Akenti) or in a delegated restricted proxy certificate (CAS scheme).

2.2 PKI-Based Communication Mechanisms

The underlying standard for secure point-to-point protocols that use PKI certificates and keys to establish authenticated and encrypted communication channels is secure sockets layer (SSL)/transport layer security (TLS). It is an IETF standard that defines a handshake protocol that commonly uses X.509 identity certificates to provide a mutually authenticated, integrity-checked and/or encrypted channel. The Globus grid security infrastructure (GSI) builds on TLS to secure a Grid environment[2]. HTTPS (HTTP over TLS) can be used with server-side certificates to provide an encrypted channel over which a Web-based client can securely send information, e.g. a name and password.

For communication involving groups, there is a secure and reliable group protocol (SGL) with properties similar to TLS that leverages off X.509 certificates to create an authenticated and encrypted multicast group[4].

2.3 Username/Password

Usernames and passwords are one of the simplest methods of establishing identity and have the advantage of being easily understood by users. There are several disadvantages of relying on passwords alone in a distributed environment. Usernames are usually assigned by each domain and are often too short to apply to many different domains. The username and password are stored encrypted at the server and sent by the user over the network to authenticate. To maintain site security, the user is often required to use a different password for each site. Thus, a user typically ends up with several user name password pairs. Also an additional mechanism such as TLS is needed to provide secure communication to avoid sending a password unencrypted over the network.

2.4 Other Mechanisms

Kerberos[6] uses symmetric keys that are kept on a central server and unlocked by a username and password to obtain tickets for use in accessing services. Each transaction in the system is preceded by a request to this server for a ticket. The overhead of running a Keberos system means that it is usually only run by large organizations. Using a ticket from one Kerberos realm for identification in a different realm requires substantial trust negotiations between the two realms and is not undertaken lightly. These characteristics make Kerberos difficult to use in a less established collaborative style environment.

Virtual Private Networks (VPN) is another solution that enterprises use to allow remote users to securely connect to a distributed environment. In the past VPN hardware has tended not to be interoperable between brands. VPN's based on IPSec may improve this situation, but would still require that all the participating sites support IPSec. Thus in a collaboration consisting of resources and people from a large variety of organizations this is often not a viable solution.

3. A Proposed Solution

There have been several attempts to build collaborative tools based on a single security mechanism (e.g. PKI, username/password, or Kerberos). These systems have not adequately addressed all our requirements. The security mechanisms taken individually each have shortcomings when applied to collaboratory environments. But, a combination of them can provide comprehensive and flexible security for collaboratories that meets our requirements.

PKI and the tools that have been built to work with it provide very good mechanisms for authentication, au-

thorization, and secure communications. But, there are two issues in using PKI. The first is the significant infrastructure required (e.g. certificate authorities, certificate repositories, authorization servers, and credential repositories). The second issue is storage of the private key. Copying the user's private key to every location the user works from is not trivial and is often not an option for security reasons.

Within an established collaboratory, an infrastructure can be maintained to store and find the public certificates, and to either store long-term private keys (credential repository) or short-term keys for delegated certificates (proxies) on a secure server. These schemes allow a user to unlock the PKI certificate from anywhere by providing the username and password over a TLS connection. This removes the requirement for many computers to have the user's private key but adds another server to the required infrastructure.

In order to work within an ad hoc collaboration as well as one with an established certificate management infrastructure, a collaboratory tool needs to support a simpler authentication mechanism such as username and password in addition to PKI authentication and authorization. The idea is to allow users to log into the system using only a username and password when the PKI infrastructure is non-existent or not able to provide authentication. Username and password enables user authentication and authorization in ad hoc and small collaborations that may not want to support a certificate management infrastructure. The username and password logins can also be used to quickly bootstrap new or infrequent users into the system. The new user can be given a username and password to use initially and then a PKI identity certificate once they are an established user. In systems where the PKI infrastructure is operating, it is likely that the authorization server will be configured to identify and limit the access rights of users that are only authenticated via a username and password.

While multiple means of authentication may have been difficult to achieve in the past, the emerging protocols which support secure communication, authentication and authorization for Web Services are designed to support various identity tokens. For example, WS-security[7] will accept a username/password, an X.509 certificate or a Kerberos ticket as a security token. The SAML protocol [8] for security assertions recognizes application defined Name Identifiers and supports passwords, various sorts of keys and Kerberos tickets for Confirmation Methods. Thus a tool using one of these protocols could implement several authentication methods behind the same interface.

4. Conclusion

The existing mechanisms taken individually provide a poor solution to securing collaborative environments. But, a combination of them can provide effective secure collaboration capabilities. The solution discussed in this paper relies on a combination of the public key infrastructure, certificate proxy mechanisms, on-line certificate repositories, group security and policy mechanisms, and username/passwords to provide comprehensive and flexible security for collaboratories. The advantages of this solution are: the core components of the collaboratory can be protected using public key infrastructure; collaborators can authenticate and gain authorization to the collaboratory via a username and password mechanism when they are away from their offices; new collaborators can be brought on board quickly by another user; and the same collaborative tools can be used by ad hoc and short-lived collaborations without investing in a full certificate management infrastructure.

5. References

[1] http://www.ietf.org/html.charters/pkix-charter.html
[2] http://www.gridforum.org/2_SEC/GSI.htm
[3] http://www-itg.lbl.gov/Akenti/homepage.html
[4] http://www-itg.lbl.gov/CIF/GroupComm/
[5] http://www-fp.mcs.anl.gov/dsl/scidac/security/
[6] http://web.mit.edu/kerberos/www/
[7] http://msdn.microsoft.com/ then click on WebServices
[8] http://www.oasis-open.org/