Organization of the Certification Authority for a University
# Table of Contents

Introduction .......................................................................................................... 1

What is the GMU CA? ........................................................................................ 1

The GMU CA Hierarchy ..................................................................................... 2

GMU CA Certification Practice Statement ........................................................ 3

   GMU CA Identity .......................................................................................... 3

   GMU CA Certification Infrastructure ........................................................... 4

   Registration Authority (RA) ........................................................................ 4

   Certificate Repository ................................................................................... 4

   Validation of Certificate Application .......................................................... 5

   Rejection of Certificate Application ............................................................ 5

   Certificate Issuance ...................................................................................... 5

   Certificate Revocation .................................................................................. 7

   Certificate Expiration .................................................................................... 8

   Certificate Renewal ...................................................................................... 8

Key components of GMU CA and their responsibilities ..................................... 8

   Overview of the certificate generation process .......................................... 9

   Overview of the certificate usage process .................................................. 10

Applications that will interact with the GMU CA .............................................. 11

   Email .......................................................................................................... 11

   University Software Applications ............................................................ 12

   Web Servers ............................................................................................. 12

Technical criteria for organizing the CA ............................................................ 12

   Key Generation ........................................................................................ 12

   Key Storage .............................................................................................. 12

   Key Distribution ....................................................................................... 13

   Key backup and recovery ....................................................................... 13

   Protocols ................................................................................................. 13
Introduction

In today’s world, parties involved in commerce conduct a significant portion of their transactions electronically. The requirements of a university to conduct electronic transactions are similar in nature to that of a commercial entity that uses electronic commerce. Some examples of electronic transactions at the university include, e-mails, course registration, publications of research, payments, receipts etc. When such transactions are conducted, it is important to ensure information security through confidentiality, data integrity, authentication and non-repudiation.

Cryptography, the study of mathematical techniques related to information security makes it possible to ensure information security in both theory and practice. Cryptography is a means for providing the necessary information security that is sought in electronic transactions. A fundamental goal of cryptography is to adequately address the four areas mentioned above. Cryptography is about the prevention and detection of cheating and other malicious activities.4

Cryptography has provided the means to create systems that ensure information security. One of the most important applications of cryptographic techniques is Certification Authorities (CA).

While conducting electronic transactions, it becomes necessary to ensure the identity of the parties involved in the transactions. Digital certificates offer an avenue for representing the identity of a party involved in a transaction. The recipient of such a digital certificate has to verify the identity of the person presenting the digital certificate. CAs, commonly known as trusted third parties (TTPs) play the role of the trusted middleman (between two parties that have never met) that certify the identity of the presenter of a digital certificate.

The objective of this paper is to present how such a CA may be organized in a university setting such as the George Mason University (GMU). This paper investigates the technical issues, administrative issues, and organizing the CA at GMU.

What is the GMU CA?

GMU CA is a central component of the GMU Public Key Infrastructure (PKI). The GMU CA certifies the identity of the GMU users and members affiliated to the GMU community, web service providers (particularly those service providers who do not have access to the GMU identification infrastructure). The GMU CA consists of multiple CAs organized in an appropriate hierarchy. A description of the hierarchy of proposed certification authorities is explained in detail in Chapter 3.

The primary responsibilities of the GMU CA are:

---

4 Handbook of Applied Cryptography, Menezes, P. Van Oorschot and S. Vanstone, Chapter 1, pg. 4
Accepting and processing applications for public key certificates

Signing public key certificates for end users who have been accepted according to the rules described in the PKI policy

Publishing the signed certificate in the CAs repository after successful delivery

Maintaining a certificate revocation list (CRL) for certificates issued and those which have been revoked due to various reasons

Ensuring the integrity and security of its private key. Compromise of the private key for a CA would result in the invalidation of all public key certificates issued by that CA.

Nearly all of these responsibilities can be met in an automated fashion by properly initiating and maintaining PKI software services.

The GMU CAs will issue three types of certificates - server certificates, client certificates and certificate authority certificates. Each server within the university will be issued a server certificate. Every user within GMU will be issued a client certificate. CA certificates will be issued by GMU Office of President CA (GMUOP CA) - root level CA to departmental CAs in order to establish a hierarchy of trust. Details of this hierarchy are listed in Chapter 3.

GMU CA will not generate or hold the private keys of Certificate Applicants. GMU CA’s private key will be secured against compromise through secure hardware products. The private key of the GMU CA will be stored in hardware that is placed in a physically secure environment. The password associated with the GMU CA will be known only to the president of GMU.

The GMU CA Hierarchy

The GMU Certificate Authorities will be arranged in a hierarchical structure with the Office of the President (GMU OP) CA as the root level CA and all campus Department Certificate Authorities as peer CAs at the next level. The Departmental CAs will issue client and server certificates within their departments. This hierarchy allows all of the campuses to recognize and accept any certificate issued within the GMU as a valid certificate. The GMU Office of President (GMUOP) is not recognized as a global or a national level CA in the most common commercially available browsers. When the GMUOP CA certificate is included in the browser, it will be automatically accepted as valid by external organizations.

The Coalition for Research and Education Networking (CREN) has established an educational domain with a root level CA that will recognize all certificate authorities at educational institutions that meet their Certification Practice Statement. This will improve interoperability between educational institutions. It is hoped that CREN CA will be recognized as a global root CA and provide a mechanism for automatic recognition of certificates issued by educational institutions by external organizations.

The GMU proposed CA hierarchy is shown in the following figure:
The CA Certification Practice Statement (CPS) describes the practices and standards employed by GMU CA to perform Certification Authority Services and to exhibit trust by providing evidence of the methods used to manage and complete tasks associated with certificate generation. The CPS is useful for not only setting the policies that GMU must adhere to, it resolves several administrative issues and problems as well.

**GMU CA Identity**

GMU CA certifies certificates in the name of the organization detailed below.

University Name: George Mason University

Location: George Mason University

Information Technology Services Department

Fairfax

Virginia
USA

**GMU CA Certification Infrastructure**

GMU CA will act as a trusted third party to facilitate the confirmation of identity within GMU community. Such confirmation is expressly represented by a certificate, i.e. a message which will be digitally signed and issued by the GMU CA. The high-level management of this certification process will include registration, naming, appropriate applicant authentication, issuance, revocation, and audit-trail generation.

GMU CA will offer three distinct levels of certification services, Client Certificates, Server Certificates and Certificate Authority Certificates. Certificate applicants will choose from this set of certificates according to their need. Applications may be made electronically to the GMU CA only for renewal of existing certificates.

In order to get a server certificate, a Department Head should submit a signed Secure Server Certificates Request for servers in their department. Secure Server Certificates can provide assurance of the existence and name of servers within the GMU.

Client certificates will be issued to individuals (such as students and staff) only. Client certificates provide assurance of the identity of individual certificate owners. Client certificate applicants need to appear before a Registration Authority Officer with a valid proof of identity like GMU Staff/Student ID card. The client certificates will typically be used for email services, online purchases, online subscription services or other web-based services.

Certificate Authority Certificates are issued by the GMU root level certificate authority to subordinate CAs in order to establish the CA hierarchy and chain of trust.

**Registration Authority (RA)**

GMU Registration Authority evaluates and approves or rejects certificate applications, exclusively on behalf of the GMU CA that actually issues the certificates.

Registration Authority Officer will be an assigned person (such as an office manager within a department of GMU) to coordinate certificate applications and validate certificate applicants’ identity and confirm the information they provide during the application process. The type, scope and extent of confirmation depend upon the class of certificate and various other factors.

Registration Authority Manager will be an assigned person (such as a head of a department), who must be a different person other than the Registration Authority Officer, to approve certificate applications, depend upon the class of certificate, after the validation procedure performed by the Registration Authority Officer and ensure that the whole certification application procedure is performed according to the practice in this CPS.

Registration Authority Console is a computer that will be setup for the Registration Authority Officer to submit certificate request to the Certification Authority after getting the approval from the Registration Authority Manager.

**Certificate Repository**

Certificate Internal Database is a database to keep track of the pending certificate request, issued or revoked certificate, private Certificate Revocation List (CRL), etc. Only RA and CA have the rights to update this database. A web user interface will be provided for users to query the status of their
certificate requests and any issued or revoked certificate. Various fields in certificate, such as serial no, expiry date, subject name, etc will be indexed. This will allow faster queries based on these standard attributes.

A high performance directory server, based on the LDAP standard, should be used as a public repository of Certificate Revocation List (CRL), user and CA certificates. The high performance directory server will be a centralized server for the entire GMU.

**Validation of Certificate Application**

Validation Requirements for Certificate Application

Upon receipt of a certificate application, GMU CA will perform all required validations as a prerequisite to certificate issuance. Once a certificate is issued, GMU CA will not continue to monitor and investigate the accuracy of the information in a certificate, unless GMU CA is notified in accordance of that certificate’s compromise.

Approval of Certificate Application

Upon successful performance of all required validations of certificate application, GMU CA shall approve the application. Approval is demonstrated by issuing a certificate according to this CPS.

If a validation fails, GMU CA will reject the certificate application by promptly notifying the applicant of the validation failure and providing a reason for such failure. A person whose certificate application has been rejected may reapply.

**Certificate Issuance**

Issuance & Publication

Upon approving a certificate application, GMU CA will issue a certificate. The issuance of a certificate indicates a complete and final approval of the certificate application by GMU CA. The issued certificate and the corresponding public key will be published to the GMU Certificate Repository and the GMU LDAP Directory server for public access.

Refusal

GMU CA may refuse to issue a certificate to any person.

Certificate Validity and Operational Periods

All certificates shall be considered valid upon:

- Issued by GMU CA, and
- Published on GMU LDAP Directory Server, and
- Is not on the GMU CA Certificate Revocation List, and
- Has not expired, and
Can be verified by a valid GMU Certification Authority certificate.

The standard operational period for Client and Server certificates is six months or time of revocation whichever occurs first. The CA certificates will be valid for twelve months or time of revocation whichever occurs first.

Certificate Format

The format of all certificates issued by GMU CA will be in accordance with ISO/IEC 9594 X.509 Version 3.
Certificate Revocation

General Reasons for Revocation of a GMU CA Certificate

A certificate shall be revoked if

- There has been a loss, theft, modification, unauthorized disclosure, or other compromise of the private key of the certificate’s subject.
- The performance of a person’s obligations under this CPS is delayed or prevented by a natural disaster, computer or communications failure, or other cause beyond the person’s reasonable control, and as a result another person’s information is materially threatened or compromised.
- There has been a modification of the information contained in the certificate of the certificate’s subject.

Revocation of a GMU CA Certificate

GMU CA must make a reasonable effort to revoke a certificate if it determines any of the following:

- A fact represented in the certificate is known or reasonably believed by GMU CA to be false.
- A prerequisite to certificate issuance was not satisfied.
- The private key or trustworthy system was compromised in a manner affecting the certificate’s reliability.
- The certificate’s subject has breached an obligation under this CPS.

Revocation at Certificate Owner’s Request

The certificate Owner must make a formal request to GMU CA to revoke their certificate. The request must be made either the following ways.

- Sending a paper Certificate Revocation Request form to GMU CA. The form must be signed with the same signature as on the original application for the certificate and/or with a valid proof of identity.
- On-Line Submission of a digitally signed Certificate Revocation Request Form. The online submission of the Certificate Revocation Request Form must be digitally signed by a valid GMU CA certificate.
Certificate Expiration

GMU CA must make a reasonable effort to notify certificate owners thirty (30) days before the expiration date, via email, of the impending expiration of their certificates. Such notice is intended solely for the convenience of the certificate Owner in the renewal process.

Certificate Renewal

For certificate renewal, certificate Owner should submit a signed written request to GMU CA before the expiration. Requests received after the expiration of the certificate will not be rejected.

Key components of GMU CA and their responsibilities

The GMU CA will consist of the following key components:

- **Registration Authority**

  The registration authority is a staff member of a GMU department. This could be a member of the GMU department administrative staff. The primary responsibility of the RA is to register students/staff into a certificate registration application. The RA will physically verify the identity of the student or staff before registering the student into the registration system. Once RA verifies the identity of the applicant, the RA will get the approval of the Registration Manager (RM) to proceed with registering the student/staff into the Registration Application system.

- **RA Terminals**

  These are computers setup in the department offices. These terminals that have access to the registration application. RA terminal resides on a machine that is not the departmental CA machine.

- **Registration Application**

  This is a web-based application that RA and students utilize to register and interact with the CA respectively. A robust web-based application is necessary to support large number of students and staff. However, the process of registering departmental CAs and servers will be limited in number. Therefore a separate application for Departmental CA and server registration is not foreseen.

- **GMUOP CA.**

  The GMUOP CA is the root CA for the university. This CA will be housed in a secure environment in the office of the President. The GMU OP CA will be kept offline most of the time. It will be brought on-line periodically to post certificates that it authorizes or publish a revocation list to the LDAP.
The GMUOP CA is responsible for issuing certificates to subordinate CAs such as departmental CAs. The GMUOP CA will process a public key and a copy of the hash that was generated by a department head. GMUOP CA will determine if there is a match. If a match exists, then the GMUOP CA will approve the departmental certificate and post it in the GMU LDAP.

- **Departmental CA**

  This CA will be housed in a secure environment in each department within GMU. The Departmental CA will be kept offline most of the time. It will be brought on-line periodically to post certificates that it authorizes or publish a revocation list to the LDAP. The departmental CA is responsible for issuing certificates to students, staff, and to departmental servers. The departmental CA will process a public key and a copy of the hash that was generated by a student or a staff (after they student or staff has physically identified themselves with the RA and RM). Departmental CA will determine if there is a match. If a match exists, then the departmental CA will approve the departmental certificate and post it in the GMU LDAP.

- **LDAP Directory Server**

  This directory server is central to all authorization and authentication activities. It is a centralized directory for GMU. The LDAP server will store all authorized university users, their passwords, certificates and certificate revocation lists. The server has to be on-line all the time. The GMU LDAP server will be a centralized server for the entire campus. It will be implemented in a high availability environment.

  In addition to being the central repository of certificates, the LDAP server will also specify the roles and access rights of users to all the computing resources in the GMU system. Since the LDAP server is central to the authentication mechanism at GMU, it has to be secured from compromise.

- **Clients for Users**

  These are computers through which students/staff will access the registration application, thereby interacting with their departmental CA. The client machines will be pre-loaded with software that will be capable of generating a public and private key pair. Once a user generates a public and private key pair, the user may interact with the CA to get their certificate published. At this point the user will be allowed to store his private key and certificate to a floppy disk, a smart card or to a token.

- **Servers**

  Servers host various pages relevant to a department that are to be accessed by students and other members of the GMU community. They will possess server certificates that are issued by the departmental CAs. These servers will be SSL enabled, thereby ensuring that users who connect to these servers have a means for verifying the authenticity of the server.

**Overview of the certificate generation process**

The following figure shows how the CA components are organized and how students/staff and department heads interact with the CAs at various levels. The
staff/student interactions are numbered S1 through S7. The interactions of departmental heads are numbered D1 – D5.

---

**Overview of the certificate usage process**

The users who want to conduct on-line transactions will utilize their certificates. The following figure describes this process:
Applications that will interact with the GMU CA

Email

All Email messages exchanged by the users within GMU will be encrypted using the public key of the recipient. Whenever a message is to be sent by the email application, the user can digitally sign the email. They may also choose to encrypt their email. In order to digitally sign their email, the email sender can attach their digital certificate to the email. If the sender wants to encrypt the email, the user can request the recipient to send a mail that they have signed. Once that email is received, the certificate associated with that email address will be stored locally. This certificate can be used by the sender to encrypt the message. The email client may also be set up to retrieve the certificate associated with the recipient from the GMU CA. Apart from signing and encrypting messages, the email will also have to be compressed.

At a minimum all email messages that are sent by GMU staff to students and that may contain sensitive information such as financial aid information, grades, medical records or other personal administrative information has to be signed, encrypted and compressed.
University Software Applications

GMU software applications that are accessible through internet or intranet need to be secure. Specific examples of applications at the GMU include registration, financial aid applications, health care records and bookstore transactions.

These applications that are accessible through a web browser can be setup so that the servers support SSL. When a user logs into an application the users browser will present their certificate. The users certificate will be verified against a directory (such as LDAP). If the user can be positively identified, the appropriate access to the application is provided to the user.

Web Servers

Users who are accessing GMU web servers need to be sure that the web server that they are interacting with is in fact a trusted GMU server and not one that poses as a university server. In order to accomplish this, all GMU Servers will be issued a server certificate and they will support SSL. When a user reaches a GMU web server through an intranet or internet, the user will be provided a message that asks the user if they trust the server. If the user wishes to verify or inspect the server for authenticity, they can examine the server certificate and the issuing hierarchy associated with that certificate.

A specific example of such servers might be the GMU Bookstore server or the student credit union server. When a student wishes to conduct a transaction with the GMU on-line bookstore, the server will present its certificate to the user. If the user accepts the certificate, the user may proceed with the transaction such as purchasing merchandise.

Technical criteria for organizing the CA

Key Generation

When a user creates a key pair, the application that generates the key pair is usually the browser that the user is utilizing. The client web browser has an open session with the key generating application at that time. Therefore the key pair generation has to occur within the client environment. Once the key pair is generated, the public key is passed on to the server for generating the user certificate. In programming environments such as Java, one can utilize objects such as iButton that will be rendered to the users' browsers. This object will be used to generate the private and public key inside the client's browser. The key generation application may then parse the public key value for use by the CA.

Key Storage

Once a private key is generated at the client, it is stored can be stored in a medium such as a floppy disk, disk drive, a network file system, or a smart-card. During storage, the private key may be encrypted by the user and protected using a password. This practice has advantages and disadvantages.

If private keys are stored on a medium such as a disk drive, the user can decrypt only when they are using their own machine. In addition, whenever students utilize a common computer, they have to
install and uninstall their keys each time. Smart-cards solve this with the requirement that every workstation have a smart-card reader. Currently there are several commercial vendors of Smart-cards. However, these vendors do not follow standards. Therefore, if the university were to utilize Smart-cards, all university users will have to buy Smart-cards from the same vendor. As things stand today, SmartCards may not be a viable option.

The next option available to users is to store their private keys on file systems that they might have access to. Placing one’s private key in an environment that is not in their control is not preferable.

Considering all these issues, GMU users private keys may be stored in a floppy drive, the user can carry it from one station to the next. Finally none of the GMU CAs may store private keys.

Key Distribution

For parties within the GMU to exchange information in an encrypted form, a sound key distribution method must exist. It is important to remember that symmetric key encryption is much faster than public-key encryption, but public-key encryption provides better authentication techniques. Therefore when GMU users interact with servers and conduct transactions, the initial user and server authentication can occur using client and server certificates. Once the users have been authenticated, the client and the server will create and share a pre-master secret for the session. This pre-master secret will then be used both by the client and the server individually to determine the master-secret. SSL protocol supports these requirements. Therefore GMU servers and clients can use SSL.

Key backup and recovery

GMU users must feel comfortable that encrypted data can be retrieved. If a user loses the private keys — for instance, due to a hard drive failure — an enhanced security solution should provide complete key backup and recovery. One of the considerations when building the CA is the seamless recovery of complete key history. Commercial products such as Entrust Security Manager offer this feature. Using this feature, the entire key history may be recovered, allowing the user to decrypt information no matter when it was encrypted. Typically users are not even aware that a key history exists. This feature will provide a secure and flexible policy to control which administrators can recover users, which users can be recovered by whom, and whether multiple authorizations are required to initiate key recovery. This practice is not recommended for GMU CA.

Protocols

This section discusses the key protocols that are utilized in order to organize the GMU CA. SSL, X.509 and LDAP have been reviewed in the following sections. The protocols selected for the GMU CA have to support the following key requirements:

- Authentication of clients and servers using standard techniques such as public-key cryptography
- Confidentiality of the communication between the server and the client using encryption
- Identification of alterations to data that is transmitted

Secure Sockets Layer (SSL) Protocol

The Secure Sockets Layer (SSL) protocol has become the universal standard on the Web for authenticating sites and for encrypting communications between users and Web servers. SSL is
perhaps the widest used security protocol on the Internet today and implements X.509 Certificates as interpreted by SSL's proponents. Because SSL is built into all major browsers and Web servers, installing a digital certificate or Server ID enables SSL capabilities. Therefore for the purposes of this study, we will utilize SSL. The GMU CA must support the SSL protocol. The following discussion describes the SSL protocol and how it functions.

The Transmission Control Protocol/Internet Protocol (TCP/IP) enables and governs the transport and routing of data over the Internet. Other protocols, such as the HyperText Transport Protocol (HTTP), Lightweight Directory Access Protocol (LDAP), or Internet Messaging Access Protocol (IMAP), are implemented "on top of" TCP/IP. HTTP, LDAP or IMAP use TCP/IP to support typical application tasks such as displaying web pages.

---

The following figure shows how SSL is organized.

The SSL protocol is intermediate between the TCP/IP and higher-level protocols such as HTTP or IMAP. SSL is above TCP/IP and below HTTP or IMAP. SSL uses TCP/IP on behalf of the higher-level protocols, and in the process allows an SSL-enabled server to authenticate itself to an SSL-enabled client, allows the client to authenticate itself to the server, and allows both machines to establish an encrypted connection.

These capabilities address fundamental concerns about communication over the Internet and other TCP/IP networks:

SSL server authentication allows a user to confirm a server's identity. SSL-enabled client software can use standard techniques of public-key cryptography to check that a server's certificate and public ID are valid and have been issued by a certificate authority (CA) listed in the client's list of trusted CAs. This confirmation might be important if the user, for example, is sending a credit card number over the network and wants to check the receiving server's identity.

SSL client authentication allows a server to confirm a user's identity. Using the same techniques as those used for server authentication, SSL-enabled server software can check that a client's certificate and public ID are valid and have been issued by a certificate authority (CA) listed in the server's list of trusted CAs. This confirmation might be important if the server, for example, is a bank sending confidential financial information to a customer and wants to check the recipient's identity.

An encrypted SSL connection requires all information sent between a client and a server to be encrypted by the sending software and decrypted by the receiving software, thus providing a high degree of confidentiality. Confidentiality is important for both parties to any private transaction. In addition, all data sent over an encrypted SSL connection is protected with a mechanism for detecting tampering—that is, for automatically determining whether the data has been altered in transit.

The SSL protocol includes two sub-protocols: the SSL record protocol and the SSL handshake protocol. The SSL record protocol defines the format used to transmit data. The SSL handshake protocol involves using the SSL record protocol to exchange a series of messages between an SSL-enabled server and an SSL-enabled client when they first establish an SSL connection. This exchange of messages is designed to facilitate the following actions:
- Authenticate the server to the client.
- Allow the client and server to select the cryptographic algorithms, or ciphers, that they both support.
- Optionally authenticate the client to the server.
- Use public-key encryption techniques to generate shared secrets.
- Establish an encrypted SSL connection
- The SSL Handshake

The SSL protocol uses a combination of public-key and symmetric key encryption. Symmetric key encryption is much faster than public-key encryption, but public-key encryption provides better authentication techniques. An SSL session always begins with an exchange of messages called the SSL handshake. The handshake allows the server to authenticate itself to the client using public-key techniques, then allows the client and the server to cooperate in the creation of symmetric keys used for rapid encryption, decryption, and tamper detection during the session that follows. Optionally, the handshake also allows the client to authenticate itself to the server.

**LDAP**

The Lightweight Directory Access Protocol was originally built as a lightweight alternative to the X.500 Directory Access Protocol. Since then it has been widely accepted by organizations as the preferred way of accessing X.500 like directory systems. Whilst an X.500 based Directory is general purpose and can hold information on objects of any kind, its most important initial applications involve facilitating communications between applications and people, holding information about the following sorts of objects: applications, people, roles, organizational units, groups, and distribution lists.

Typical information about the objects includes:

SMTP and X.400 Email addresses, fax and telephone numbers, photographs, security information (keys and passwords) and objects’ capabilities.

In principle any organization can hold a piece of the Directory, typically that concerning its people and processes. The X.500 standard governs how that piece can be connected to others to give the effect of a global information base, although the actual interconnection will be determined by the security policy of the organization.

Users are now building directory systems and services, typically using products conforming to the 1993 X.500 standard (for example, from major international suppliers such as SNI, ICL, Unisys and ISOCOR) as well as those supporting only LDAPv2 (for example, from major international suppliers such as Netscape and Microsoft.

The Lightweight Directory Access Protocol (LDAP) has evolved to become the Internet Standard way of accessing on-line directory systems that follow the X.500 information models. LDAP Directories can hold information on objects of any kind. Typical information about the objects includes: email addresses, fax and telephone numbers, photographs, security information (public key certificates and passwords), as well as an object’s capabilities and the policies that control them.
One of the confusing things about LDAP is that it looks like another kind of database. If you've had experience with SQL or any relational database, LDAP looks like a way to get at a simple kind of database.

First impressions are correct. You can use LDAP to retrieve data from a standard database, as Oracle and Computer Associates have demonstrated with their LDAP directories, which have been built on top of their relational databases.

LDAP server vendors, such as Netscape and Innosoft, don't like this comparison. For one thing, they don't want to compete with established database vendors. Selling database software against Oracle is only slightly easier than selling operating system software against Microsoft. However, they also want to emphasize the differences between their directories and a traditional relational database. LDAP sharply defines the operations you can do on the directory.

You can do five things to an LDAP directory. A client can connect or bind to it, search it, modify it, add to it and delete from it. All other features of a relational database (transactions, multitable queries, views and joins) have no counterparts in an LDAP realm. An LDAP server offers none of this power, at least not via the LDAP interface. Nevertheless, the trade-off comes with a big payoff. With an LDAP store, you get greater speed, lower cost, a simpler data model and easier management and implementation.

LDAP defines an access method to a database, but it also defines a database implementation philosophy. Technically, there is no such thing as an LDAP directory. However, LDAP so strongly colors the database design, that most people call the directory you talk to with LDAP an "LDAP directory." A directory optimized for LDAP is easier to build than a general-purpose database. For example, there is no concept of transactions or rollbacks in LDAP, so all the complexity of locking and journaling you see in relational databases and all the overhead that complexity brings is gone in an LDAP environment.

If there were a traditional database that mapped well to an LDAP directory, it wouldn't be the modern relational databases, but a hierarchical database such as IBM's Information Management System. This is because LDAP directories are arranged as hierarchical trees of information, designed to mimic the hierarchy of the organizations they describe: locations such as country, city and state, organizational units, departments, and so on down to people, conference rooms and printers.

The number of entries you can put in the directory and the rate at which you can query it segments the LDAP directory market. If you're running a customized Web portal site and you have 10 million members, and you get one million hits per day, you probably have an "extranet" or "e-commerce" LDAP server. With this type of product, you have the capability to do a particular type of query (exact match on member ID, for example), handle a lot of reads and have a huge database that doesn't change much from day to day.

On the other hand, if you're running a mail backbone for a Fortune 500 company, you probably get one-tenth of the queries of an extranet LDAP directory, but you may get a mix of exact match, fuzzy or wildcard searches. Plus you have a lot fewer entries in your database -- 50,000 employees would be a reasonable number for this class (an enterprise directory) of LDAP directory.

The distinction is somewhat arbitrary. Vendors such as Microsoft and Novell, which you would think fit better into the enterprise model of directories, won't hear of it. Both firms are ready and willing to fit a few million entries (in Novell's case, one billion entries) into their directories and want to go up against the best of them.

However, the reality is that no directory can be optimized for all applications. IPlanet (the new Netscape), which wants to be the premier extranet/e-commerce directory vendor, can bulk-load a one-million-entry directory in less than an hour. This is a common operation in LDAP directories, where it is often simpler to reload the directory from scratch than to keep it updated with changes throughout the
Microsoft's Active Directory isn't designed for that modus operandi and would have a hard time bulk-loading the same database in 12 hours.\(^6\)

**X.509**

X.509 is a specification used around the world and any applications complying with X.509 can share certificates. It defines a framework for provision of authentication services by the X.500 directory to its users. X.509 is based on the use of public-key cryptography and digital signatures. The standard does not dictate the use of a specific algorithm but recommends RSA\(^7\). The heart of the x.509 scheme is the public key certificate associated with a user. Most certificates today comply with X.509 Version 3 and contain the information listed in this table.

<table>
<thead>
<tr>
<th>Contents of an X.509 V3 Certificate.</th>
</tr>
</thead>
<tbody>
<tr>
<td>version number</td>
</tr>
<tr>
<td>certificate serial number</td>
</tr>
<tr>
<td>signature algorithm identifier</td>
</tr>
<tr>
<td>issuer's name and unique identifier</td>
</tr>
<tr>
<td>validity (or operational) period</td>
</tr>
<tr>
<td>subject's name and unique identifier</td>
</tr>
<tr>
<td>subject public key information</td>
</tr>
<tr>
<td>standard extensions</td>
</tr>
<tr>
<td>certificate appropriate use definition</td>
</tr>
<tr>
<td>key usage limitation definition</td>
</tr>
<tr>
<td>certificate policy information</td>
</tr>
<tr>
<td>other extensions</td>
</tr>
<tr>
<td>Application-specific</td>
</tr>
<tr>
<td>CA-specific</td>
</tr>
</tbody>
</table>

The certificates generated by CAs are unforgeable. Therefore they can be placed in a directory and the directory does not need to be protected extensively. Certificate authorities are the repositories for public-keys and can be any agency that issues certificates. A company, for example, may issue certificates to its employees, a college/university to its students, a store to its customers, an Internet service provider to its users, or a government to its constituents.

**PGP**

PGP is one of the most widely used public key cryptography programs. PGP is commercially available today and is also available as a plug-in for many e-mail clients, such as Microsoft's Exchange and Outlook, and Qualcomm's Eudora.

PGP can be used to sign or encrypt e-mail messages. Depending upon the version of PGP, the software uses SHA or MD5 for calculating the message hash; CAST, Triple-DES, or IDEA for encryption; and RSA or DSS/Diffie-Hellman for key exchange and digital signatures. When PGP is first installed, the user has to create a key-pair. One key, the public key, can be advertised and widely circulated. The private key is protected by use of a passphrase. The passphrase has to be entered every time the user accesses their private key.

---


\(^7\) Cryptography and Network Security, Principles and Practice, William Stallings, Chapter 11, pg 341
PGP Web of Trust

A PGP user maintains a local keyring of all their known and trusted public keys. The user makes their own determination about the trustworthiness of a key using what is called a "web of trust."

If Alice needs Bob's public key, Alice can ask Bob for it in another e-mail or, in many cases, download the public key from an advertised server; this server might be a well-known PGP key repository or a site that Bob maintains himself. In fact, Bob's public key might be stored or listed in many places. Alice is prepared to believe that Bob's public key, as stored at these locations, is valid.

Suppose Carol claims to hold Bob's public key and offers to give the key to Alice. How does Alice know that Carol's version of Bob's key is valid or if Carol is actually giving Alice a key that will allow Mallory access to messages? The answer is, "It depends." If Alice trusts Carol and Carol says that she thinks that her version of Bob's key is valid, then Alice may — at her option — trust that key. And trust is not necessarily transitive; if Dave has a copy of Bob's key and Carol trusts Dave, it does not necessarily follow that Alice trusts Dave even if she does trust Carol.

The point here is that who Alice trusts and how she makes that determination is strictly up to Alice. PGP makes no statement and has no protocol about how one user determines whether they trust another user or not. In any case, encryption and signatures based on public keys can only be used when the appropriate public key is on the user's keyring.

PGP Key revocation

The convention for key revocation in PGP is for the owner to issue a key revocation certificate, signed by the owner. This certificate has the same form as a normal signature certificate but includes an indicator that the purpose of this certificate is to revoke the use of this public key. The corresponding private key must be used to sign a certificate that revokes a public key. The owner should then try to disseminate this information as quickly and as widely as possible to enable correspondents to update their public-key rings.

S/MIME

S/MIME(Secure/Multi-purpose Internet Mail Extension) is based upon technology from RSA Data Security. S/MIME is emerging as the standard for commercial and organizational use, while PGP remains the choice of personal e-mail security. S/MIME is recommended for use at GMU.

S/MIME is very similar to PGP in its functionality. Both offer the ability to sign and/or encrypt message formats and message preparation. S/MIME provides the following functions: enveloped data, signed data, clear-signed data and signed and enveloped data.

For S/MIME DSS is the preferred algorithm for digital signature, Diffie-Hellman is the preferred algorithm for encryption/decryption and for the hash function used in creating the digital signature 160-bitSHA-1 is the preferred. For message encryption, three-key triple DES is recommended.

S/MIME uses public-key certificates that conform to version 3 of X.509. The key management scheme used here is a hybrid between a strict X.509 certification authority and PGP's web of trust. That is, the responsibility is local for maintaining the certificates needed to verify incoming signatures and to encrypt outgoing messages. On the other hand, certification authorities sign the certificates.  

---

8 Cryptography and Network Security, Principles and Practice, William Stallings, Chapter 12, pg 391
**Comparison of Algorithms**

RSA vs Diffie Hillman (DH)

The advantage of RSA has over DH:

If a file is encrypted with RSA, it will be smaller than if the file was encrypted with DH.

Advantages of DH over RSA:

The key advantage of DH vs RSA is that, no copyrights or patents cover DH. This means that anyone who wants to create a security package has full access to the algorithm used in DH. However, to use RSA, one must first obtain licensing from RSA Labs for commercial products. Even after obtaining a license, not all libraries are released.

The next advantage of DH over RSA is that, in RSA, someone could use a guess and check method to try to find the private key. By encrypting common phrases with the available public key, it is not likely the private key could be found, but it is possible. RSA offers less security for the same amount of bits. For example, DH in the PGP software package, uses certain session keys in its encryption. It records mouse movements and key strokes and uses them in the key generation process. Without these keys, someone could physically be at the receiver’s computer and coerce the receiver into later revealing his private key. In terms of efficiency, DH in generally much faster than RSA.

In conclusion, from the arguments presented in favor of each algorithm, DH is obviously the most desirable encryption algorithm. With its session keys and large advantage in efficiency, it easily makes up for the few downfalls.

**Chapter 11**

**Hardware and Software Platform Selection**

In order to implement the GMU CA, several different hardware and software choices exist. This study recommends the following platforms for the university:

**Software:**

CA – The Netscape Certificate Server may be used as the software for creating, signing and managing standard-based public-key certificates. This product will be utilized for creating the CA and the registration application.

LDAP Directory Server – Netscape Directory Server or iPlanet may be utilized as the LDAP Directory server. Alternately an LDAP directory server may be developed by the GMU.

**Hardware:**

CA Server – HP K200 class of machine or SunSparc Ultra 10 Server
LDAP Server – SunSparc Ultra 10 Server. This server must be utilize high availability architecture.
## Commercial CA products

The following vendors offer commercial products.

<table>
<thead>
<tr>
<th>Component</th>
<th>Vendor</th>
<th>Platforms</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDAP</td>
<td>Iplanet (Netscape)</td>
<td>Sun Solaris 2.6 (UltraSPARC)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sun Solaris 8 (UltraSPARC, 32-bit and 64-bit)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Microsoft Windows NT 4.0 Server (x86 only, Service Pack 6a)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Microsoft Windows 2000 Server (Service Pack 2)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Microsoft Windows 2000 Advanced Server (Service Pack 2)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Hewlett Packard HP-UX 11.0 (PA-RISC 1.1 or 2.0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IBM AIX 4.3.3 (32-bit, PowerPC)</td>
</tr>
<tr>
<td>LDAP</td>
<td>NDS – Novell</td>
<td>Linux, Netware 5.x, Windows NT/2000</td>
</tr>
<tr>
<td>LDAP</td>
<td>Microsoft – Active Directory</td>
<td>Windows NT/2000</td>
</tr>
<tr>
<td>CA</td>
<td>Netscape Certificate Server</td>
<td>Sun Solaris 2.6(UltraSparc)</td>
</tr>
<tr>
<td>CA</td>
<td>Verisign-OnSite Managed Trust Service</td>
<td>Windows NT/2000</td>
</tr>
<tr>
<td>CA</td>
<td>Entrust Authority Security Manager 6.0</td>
<td>Windows NT/2000</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HP UX 11</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AIX 4.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Solaris 7 &amp; 8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Oracle 8i &amp; Informix</td>
</tr>
</tbody>
</table>
**Conclusion**

A certificate authority is very useful to GMU. The CA will enable GMU parties that do not know each other to conduct transactions by ensuring information security through confidentiality, data integrity, authentication and non-repudiation. By adhering to industry standards such as X.509 v3, the GMU CA can work with external organizations as well.

The costs associated with implementing a CA are reduced due to the fact that most of the key components, except for the CA software and hardware are readily available. For example the network infrastructure, the e-mail/messaging infrastructures are already well established. The knowledge base to implement such a system exists at GMU. The administrative issues associated with organizing the CA such as personnel, policies are also mitigated by following the GMU organization structure. Further, reliance on CREN as the root certifying authority enables GMU to interoperate with other universities that have CREN as the root CA. This also reduces the cost by eliminating reliance on vendors.

The GMU CA will be the center piece of a system that authenticates the identity of GMU community members who wish to communicate with one another through electronic means.

**References**

A.J. Menezes, PC Van Oorschot and SA Van Stone, “Handbook of applied cryptology”.
http://www.cren.net/know/techtalk/trans/caservices_1.html#future
www.ietf.org
www.ietf.org/rfc/rfc2560.txt
http://www.sevenlocks.com/papers/paperscryptopk.htm
http://www.nwfusion.com/reviews/2000/0515rev1.html, Joel Snyder, Network Test Alliance
http://sec.isi.salford.ac.uk/download/smart.pdf
http://www.securecomputing.com/index.cfm?sKey=675
http://www.uk.pgp.net/pgpnet/secemail/q4/node1.html
http://solution.ecn.purdue.edu/~simmons/crypto.html#rsamath