FEDERAL PUBLIC KEY INFRASTRUCTURE

DIRECTORY PROFILE

Version 2.0 (Draft)

March 14, 2002

[conract title and number]

Prepared By:

Booz | Allen | Hamilton
900 Elkridge Landing Road
Linthicum, Maryland 21090

Table of Contents

1.0 INTRODUCTION

This profile defines the requirements for the initial operational Federal Public Key Infrastructure (FPKI) directory system. The FPKI will use the Federal Bridge Certification Authority (FBCA) that cross-certifies with agency Principal Certification Authorities (CAs) to provide trust paths between the agencies. A directory server within the FBCA will handle X.500 chained operations with agency border directories, and will also service referrals from agency Lightweight Directory Access Protocol (LDAP) directory servers. These operations are explained in detail within later sections of this document. The Border CA concept is described in [1].

The FPKI builds upon the Federal Bridge Certification Authority (FBCA) prototype that was successfully demonstrated during the Electronic Messaging Association (EMA) Challenge in April 2000. This prototype supported S/MIME messaging among several disparate Public Key Infrastructure (PKI) domains using several different CA products, X.500 directory products, and S/MIME e-mail clients. This demonstration illustrated interoperability on several levels – between CAs, between directories, and between e-mail clients. Each client created, and then processed a certificate trust path between the domain of the recipient and the domain of the sender in order to validate the signer's digital signature on the e-mail. Trust paths up to seven certificates were constructed and validated. Directories were chained using the X.500 Directory Services Protocol (DSP), while LDAP)was employed by the e-mail client to access its local directory [2].

This profile addresses the minimum required directory schema, naming conventions, directory protocols supported, security considerations, alternatives to consider, and issues to bear in mind in order to adapt to this evolving technology. Familiarity with PKI technology, concepts and general terms of the directory service is assumed.

The draft is based on several sections of the following documents:

- The Evolving Federal Public Key Infrastructure [2],
- Governmentwide Directory Support 2 Technical Series, the Updated US Gold Schema document [3],
- The Bridge CA Demonstration Repository Requirements Draft 4/8/1999 [4], and
- NSA Bridge Certification Authority Demonstration Phase II - Directory Requirements and Architecture, 7/3/2000 [5].

2.0 SCHEMA REQUIREMENTS

This section addresses the minimum schema requirements for agency directories to interoperate with the FPKI directory. The schema is limited to just the objects needed to support the PKI. At a minimum, the directories are required to store and disseminate the following PKI related attributes:

- Certification Authority Certificates
- Certificate Revocation Lists
- Authority Revocation Lists
- Cross Certificates
- End-entity certificates
- RFC822MailUser

In the Internet X.509v3 Public Key Infrastructure LDAPv2 Schema [6], these attributes are:

- cACertificate
- certificateRevocationList
- authorityRevocationList
- crossCertificatePair
- userCertificate
- rfc822Mailbox

This schema is used in some commercial CA products.

Some agencies may wish to make other information available externally to support their PKI applications. However, this profile does not address or impose requirements on application-specific data in agency directories.

The *cACertificate* and *crossCertificatePair* attributes require special attention when accessing the directory to build the certificate path. Neither the Public Key Infrastructure (X.509) – PKIX – specification nor the X.509 standards explicitly provide an algorithm to construct a certificate path. The PKIX LDAP-V2-schema provides guidance on what can be stored in the specific attributes. The draft states the following about the *cACertificate* attribute and the *crossCertificatePair* attribute:

> The *cACertificate* attribute of a CA's directory entry shall be used to store self-issued certificates (if any) and certificates issued to this CA by CAs in the same realm as this CA.

The forward elements of the *crossCertificatePair* attribute of a CA's directory entry shall be used to store all, except self-issued certificates issued to this CA. Optionally, the reverse elements of the *crossCertificatePair* attribute of a CA's directory entry may contain a subset of certificates issued by this CA to other CAs. When both the forward and the reverse elements are present in a single attribute value, issuer name in one certificate shall match the subject name in the other and vice versa, and the subject public key in one certificate shall be capable of verifying the digital signature on the other certificate and vice versa.

When a reverse element is present, the forward element value and the reverse element value need not be stored in the same attribute value; in other words, they can be stored in either a single attribute value or two attribute values.

In the case of V3 certificates, none of the above CA certificates shall include a *basicConstraints* extension with the cA value set to FALSE.

A path development algorithm must consider that the CA's certificate must be stored in the *crossCertificatePair* attribute, but the algorithm may consult the *cACertificate* attribute first, for performance reasons.

The following sections define the attributes and object classes that are required for end entities and CAs.

2.1 END ENTITIES

2.1.1 Attributes

End entity (EE) directory entries shall contain, as a minimum, the following attributes:

userCertificate as defined in 1997 X.509v3 [7] (OID: 2.5.4.36),

attributeCertificate as defined in 1997 X.509v3 (OID: 2.5.4.58),

commonName as defined in 1997 X.521 [8] (OID: 2.5.4.3),

surname as defined in 1997 X.521 (OID: 2.5.4.4).

Note**:** The EE relative distinguished name (RDN) shall consist of the *commonName* attribute type and value. For example: cn=John Smith

2.1.2 Object Classes

EE entries shall be made up of the following object classes:

person as defined in 1997 X.521 (OID: 2.5.6.6).

pkiUser as defined in RFC 2587: LDAPv2 Schema (OID: 2.5.6.21) for non-Entrust EEs -- OR -- *entrustUser* as defined in "Entrust Directory Schema Requirements" version 1.0, dated August, 1998 (OID: 1.2.840.113533.7.67.0) for Entrust EEs.

securePkiUser as defined in Allied Communications Publication (ACP) 133 Edition B [9] (OID: 2.16.840.1.101.2.2.3.66). This auxiliary object class includes *attributeCertificate* and *supportedAlgorithms* as optional attribute types.

Optionally, EEs may include the following object classes:

organizationalPerson as defined in 1997 X.521 (OID: 2.5.6.7),

inetOrgPerson as defined in Internet Engineering Task Force (IETF) Request for Comment (RFC) 2798 [10] (OID: 2.16.840.1.113730.3.2.2).

2.2 CERTIFICATION AUTHORITIES

2.2.1 Attributes

CA entries in the directory, including Policy Creation Authorities (PCAs) and Policy Approving Authorities (PAAs), shall contain at a minimum the following attributes:

commonName OR *organizationalUnitName* as defined in 1997 X.509v3 (OIDs: 2.5.4.3 and 2.5.4.11 respectively).

cACertificate as defined in 1997 X.509v3 (OID: 2.5.4.37). As per the LDAPv2 Schema (RFC 2587), the cACertificate attribute shall be populated as follows:

> "The *cACertificate* attribute of a CA's directory entry shall be used to store self-issued certificates (if any) and certificates issued to this CA by CAs in the same realm as this CA."

certificateRevocationList as defined in 1997 X.509v3 (OID: 2.5.4.39)

crossCertificatePair as defined in 1997 X.509v3 (OID: 2.5.4.40). As per the LDAPv2 Schema (RFC 2587), the *crossCertificatePair* shall be populated as follows:

> "The forward elements of the *crossCertificatePair* attribute of a CA's directory entry shall be used to store all, except self-issued certificates issued to this CA. Optionally, the reverse elements of the *crossCertificatePair* attribute of a CA's directory entry may contain a subset of certificates issued by this CA to other CAs. When both the forward and the reverse elements are present in a single attribute value, issuer name in one certificate shall match the subject name in the other and vice versa, and the subject public key in one certificate shall be capable of verifying the digital signature on the other certificate and vice versa.
>
> "When a reverse element is present, the forward element value and the reverse element value need not be stored in the same attribute value; in other words, they can be stored in either a single attribute value or two attribute values."

CAs entries in the directory may optionally contain the *authorityRevocationList* attribute as defined in 1997 X.509v3 (OID: 2.5.4.38).

> Note: The CA RDN shall consist of either the *commonName* attribute type and value or the *organizationalUnitName* attribute type and value. For example: *cn=NSA CA* -- OR -- *ou=ECA1*

2.2.2 Object Classes

CA entries shall be made up of the following object classes:

pkiCA as defined in RFC 2587: LDAPv2 Schema (OID: 2.5.6.22) for non-Entrust CAs -- OR -- *entrustCA* as defined in "Entrust Directory Schema Requirements" version 1.0, dated August, 1998 (OID: 1.2.840.113533.7.67.1) for Entrust CAs.

The base object class of CAs shall be one (or more) of the following:

person as defined in 1997 X.521 (OID: 2.5.6.6)

organizationalPerson as defined in 1997 X.521 (OID: 2.5.6.7)

inetOrgPerson as defined in IETF RFC 2798 (OID: 2.16.840.1.113730.3.2.2)

organizationalUnit as defined in 1997 X.521 (OID: 2.5.6.5)

3.0 NAMESPACE CONTROL AND DIRECTORY TREE STRUCTURE

Public key infrastructure certificates and related objects are defined by the X.509 specification, which is a portion of the overall X.500 information model. These PKI objects are made available to relying parties by a directory service. The FBCA directory service acts as a bridge between various agency directory services, allowing relying parties to retrieve the

The X.500 information model is used by both X.500- and LDAP-based directory servers, and forms the basis for interoperability between directory services. Objects within the federal directory services are located using the object's Distinguished Name (DN), which specifies both the Relative Distinguished Name (RDN) of the object and it's location within the overall federal directory.

The federal directory service can be thought of as a tree – a logical hierarchical structure composed of all the various directory services operated by federal agency directory services, and made possible by general agreement on naming schemes and directory structures. An agency's "namespace" refers to the individual directory, or subtree, that is controlled by a specific agency.

3.1 AGENCY DIRECTORY SERVICE REQUIREMENTS

Agencies are not required to conform to any specific directory protocol internally. But, in order to interoperate with the FBCA, an agency's directory service must conform to the following requirements:

- The agency must register their directory service as in Section 3.1.1 with the FBCA in order to establish interoperability.
- The agency's PKI information must conform to the X.500 information model and X.509.
- The agency's directory service must support 1993 X.500 chained operations, 1993 X.500 referrals, or LDAP v3 referrals.
- The agency's PKI information must conform to one of the namespace strategies stated in Sections 3.2, 3.3, and 3.4, below.

The agency may choose to employ a Border Directory Server Agent (DSA) to provide for protocol conversion, enforce security, and restrict access to internal directory services. Alternate approaches are discussed in Appendix B, along with relevant security implications and considerations.

3.1.1 Registration

In order to support connectivity between the FBCA and agency directory services, each agency participating in the FPKI must register their directory service or Border DSA with the FBCA Operational Authority (OA). Appendix A contains a worksheet to aid you in collecting this information prior to registration.

The following information must be provided:

- Name and address of agency desiring to interoperate with the FBCA directory
- Name, address and contact information for that agency's directory administrator
- Distinguished Name, Network Address, and Host Name of directory service
- Naming Context (namespace) provided by this directory server (see Sections 3.2, 3.3, and 3.4)
- Protocols supported (X.500 and/or LDAP)

3.2 X.500 DIRECTORY SERVICES

If the agency chooses to use an X.500-based directory service, its directory must conform to the name space as defined for the Federal Government [3] (Figure 3-1). This namespace contains the U.S. Government level of the global X.500 Directory Information Tree (DIT) and all governmental agencies and departments. In X.500 terms, this namespace includes directory servers with the naming context of:

c=us; o=U.S. Government

The U.S. Government is registered as an *organization* (o) object in the Global DIT, directly subordinate to the *country=us* object (the national U.S. country level object). Agencies and departments occupy *organizationalUnit* (ou) objects immediately beneath the o=*U.S. Government* entry. Agency and department names in the Federal Government namespace must conform to agency and department names as stated in the Federal Government Manual. This publication cites official names for agencies and departments (*organizationalUnit*s) of the Federal Government. For instance, Transportation and Treasury would be:

c=us; o=U.S. Government; ou=Department of Transportation

c=us; o=U.S. Government; ou=Department of the Treasury

The Federal Aviation Administration and Internal Revenue Service have been assigned the following directory naming contexts based on their official names and parent agencies:

c=us; o=U.S. Government; ou=Department of Transportation; ou=Federal Aviation Administration

c=us; o=U.S. Government; ou=Department of the Treasury; ou=Internal Revenue Service

Each agency or department is free to define and manage the namespaces for organizational units within that agency.

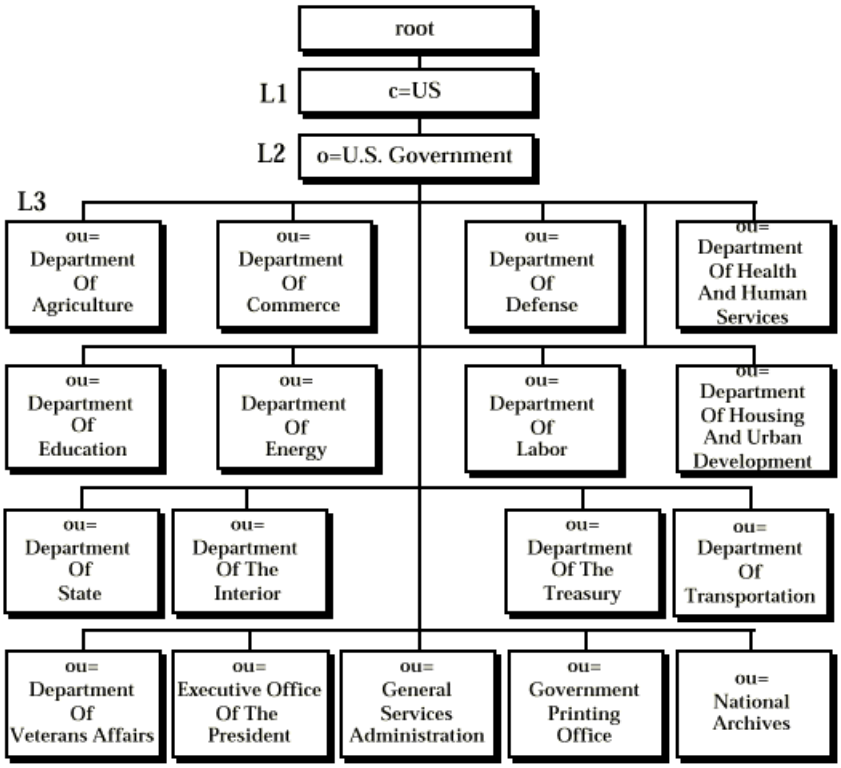


Figure 3-1 Federal Government Top Level Directory Naming

Abbreviations are allowed, but must be negotiated with the FBCA Registrar (as in Section 3.1.1 above) to ensure uniqueness within the U.S. Government namespace. Potential conflicts on abbreviations may occur, and will be solved as follows.

If the agency or department has a current domain registration within the Internet Domain Naming System (DNS) underneath *.gov*, they may use this as an abbreviation within the Federal Government directory. For example, the Department of Transportation is registered as *dot.gov*, and thus can use:

c=us; o=U.S. Government; ou=DOT

Department of the Treasury, however, is registered as *treas.gov*, and thus could use:

c=us; o=U.S. Government; ou=TREAS

Any number of *organizationalUnitNames* may be registered to aid in directory searches. For example, the Department of Transportation directory entry might have the following names (if authorized by the FBCA Registrar):

c=us; o=U.S. Government; ou=Department of Transportation (e.g.; the "official" name)

c=us; o=U.S. Government; ou=Transportation

c=us; o=U.S. Government; ou=DOT

The actual directory entries would be found underneath the name specified in the Federal Government Manual. All other DNs would actually be aliases pointing to the "official" name.

3.2.1 DNs versus RDNs

Each object stored in the directory is identified by a Relative Distinguished Name (RDN). The RDN is the value that uniquely identifies an entry within the current node, or container, of the directory. For instance, if the Relative Distinguished Name is a person's full name (e.g. their *commonName* or *cn*), each directory entry within a specific level will have a unique RDN. At a given level of the directory, there can exist only one, single entry with a RDN of *cn=John Smith*.

Since the directory contains several levels (nodes or containers), there might exist multiple nodes with a RDN of *cn=John Smith* throughout the directory tree. But within each individual directory node or container, there can exist only one such entry.

An entry in the directory is specified by its full Distinguished Name (DN), which is composed of all the RDNs starting with the top of the tree and moving downward to the specific entry. In the original X.500 syntax, the RDNs that composed a full DN were separated from each other by an @ sign, and listed beginning at the top of the tree. The full X.500 DN would look like:

c=US@o=U.S. Government@ou=General Services Administration@cn=John Smith

LDAP typically reverses the order and uses commas rather than @ signs, like this:

cn=John Smith, ou=General Services Administration, o=U.S. Government, c=US

Functionally, both DNs are the same. The DN describes the path through the directory tree, which contains the following objects:

- A *country* object with a RDN of *c=us*
- An *organization* object with a RDN of *o=U.S. Government*, which is subordinate to the *c=us* object.

- An *organizationalUnit* object with a RDN of *ou=General Services Administration*, which is subordinate to the *o=U.S. Government* object.
- The targeted person entry with a RDN of *cn=John Smith*

3.2.2 Advantages and Disadvantages of X.500

The X.500 naming scheme is well understood. It is supported in current PKI products, which have been successfully demonstrated in the PKI FBCA and the EMA challenge demonstrations. The drawback of this naming scheme is that it is little used by anyone other than for PKI. Most users do not understand nor care about the finer distinctions of the Federal structure. Hence, distinguished names with organizational structure embedded in them are generally difficult for users to comprehend or remember.

In addition, the more structure that is embedded in names, the more certificates that would need to be revoked when structures change. And the more structure that is built into the names, the more the name space needs to be administered. Many agencies have adopted a very "flat" namespace, where all the organization's users are listed directly underneath the agency object or within a single subtree, regardless of location or organizational structure.

Another recurring debate, which occurs with X.500-based systems, lies in the directory tree structure within the agency. There are three basic approaches:

- Put all the directory entries into a single, flat namespace (usually requires a single DSA serving the entire agency).
- Divide the tree to mirror organizational structure (may create problems if the directory servers are located in multiple geographic locations).
- Divide the tree to mirror geographical or network infrastructure (presents issues related to interactive searching and use).

The Federal Bridge CA has no preference and issues no guidance as to the tree structure of internal agency directory services. This area is clearly outside the scope of this document.

3.3 INTERNET DOMAIN NAME BASED NAMING

With the global acceptance of Internet and technologies such as the Domain Name System (DNS) and RFC822-based e-mail, many portions of the government have ignored older technologies such as X.500 and have implemented Internet-based infrastructures. These infrastructures are used primarily for e-mail and web-based delivery of services and information.

The Internet DNS provides a hierarchical naming and locating system based on domain name components. For instance, the Internal Revenue Service is registered as *irs.treas.gov*. The U.S. Federal government "owns" the *gov* "top-level domain", and is responsible for assigning and administering domain component names underneath that domain. Department of the Treasury (Treasury) has registered the domain component of "*treas*", underneath *gov*. Therefore, any e-mail user at the Department of the Treasury would have an e-mail address something like *user@treas.gov*, and the main Treasury web page would be found at *www.treas.gov*.

The Internal Revenue Service has been assigned the domain component of *"irs"* by Treasury, such that a user within IRS should have an email address of *user@irs.treas.gov*. However, IRS has also registered directly underneath *.gov*, meaning that most IRS personnel use email addresses like *user@irs.gov* and the main IRS web page is found at *www.irs.gov*.

This DNS-style of naming was originally developed to support hierarchical management and searching of computer system names (e.g. "hostnames"). Each computer attached to the Internet has an Internet Protocol (IP) address, which consists of four numbers between 0 and 255, separated by periods. These addresses look something like 192.248.32.14. Clearly, this is hard for users to comprehend, much less remember. Who wants to address an email message to john.smith@192.248.32.14? (Actually, this address *will* work on many Internet-connected systems). DNS maps this numeric IP address into a human-readable system name, called a Fully Qualified Domain Name (FQDN). This allows a user to send email to *john.smith@company.com* instead of trying to remember the IP address. The computer looks up *company.com*, finds the numeric address, and makes the connection. In this sense, IP addresses are like telephone numbers, and DNS is like a giant, worldwide electronic phone book.

X.500 is a completely separate directory system from DNS. However, a proposed Internet Standard as described in RFC 2247 [11] and RFC 2377 [12] provides a method of representing Domain Name System domain components using the X.500 information model. This allows both X.500 and LDAP-based directory services to store information in a structure familiar to Internet-literate users.

RFC 2247 defines an attribute, *domainComponent (dc)*, which can be used to store a domain component such as "*gov"*. It also defines two objects, *domain* and *dcObject*. The *dcObject* object can be added to existing objects so that they can contain a *dc* attribute. The *domain* object allows the addition of new entries that contain a *dc* attribute.

Using *domain* objects, it is possible to accurately represent the DNS "tree" within an X.500 or LDAP directory service (Figure 3-2). The user specified by the email address *john.smith@irs.treas.gov* would be represented by the X.500 DN:

dc=gov; dc=treas; dc=irs; pn=john.smith

LDAP allows a relaxed form of DN in reverse order separated by commas, which looks like:

pn=john.smith, dc=irs, dc=treas, dc=gov

The information in the directory is the same either way. Searching based on this DNS-style naming can be very intuitive to users who are familiar with Internet email addresses. The Federal Bridge CA will allow agencies to choose to implement naming in this fashion, instead of (or in addition to) the X.500-style Federal Government naming set forth in Section 3.1.

Additionally, the *dcObject* object can be used to add the *dc* attribute to other X.500 objects. Therefore it can allow for construction of DNs which look very much like X.500, but which are actually composed of *DomainComponent* attributes. This sort of DN would look like:

dc=us; dc=U.S. Government; dc=treas; dc=irs; pn=john.smith (or)

pn=john.smith, dc=irs, dc=treas, dc=U.S. Government, dc=us

The Federal Bridge CA will *not* support this style naming. Its similarity to pure X.500 naming can cause significant confusion. Since it doesn't map to the Internet-style e-mail addresses, it is not intuitive to use and therefore provides no discernable benefit. As DNS evolves in the future, country-based naming may come into use. If so, this decision will be revisited at that time.

3.3.1 Drawbacks of DNS-Style Naming

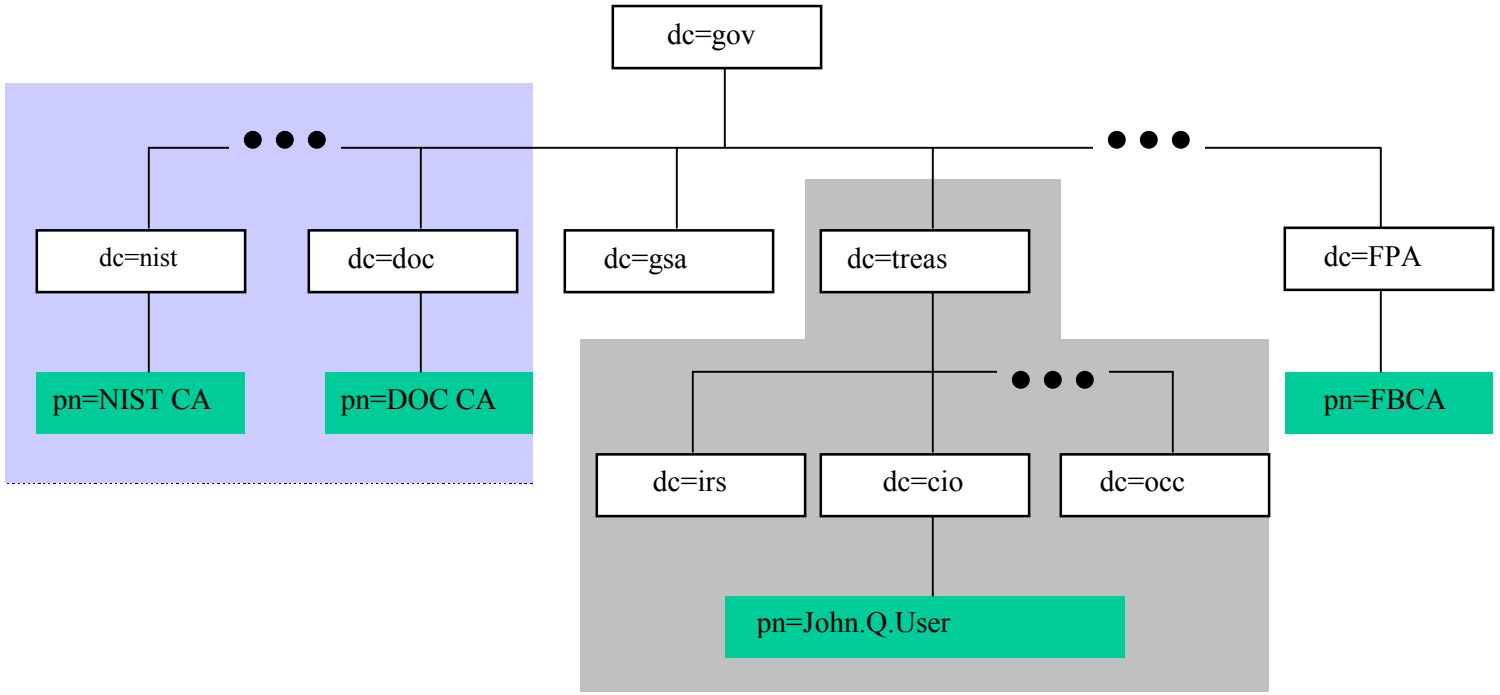
RFC 2247, the document that proposes this style of addressing, is a proposed Internet Standard. It therefore is fairly stable and not subject to major changes. However, it may not be widely implemented in applications and commercial software products yet.

The *.gov* domain is owned by the U.S. Government. Registration of government agencies and operation of the government-level DNS is outsourced to a vendor. There is little guidance relating to the creation of DNS-style domain information for government agencies. This leads to several confusing situations:

Internet domain name components are typically short and cryptic. Many times, all users appear directly underneath the organization with no clue as to organizational structure or geographic location. Also, many agencies have registered domain names that don't reflect the actual Federal departmental structure. This may be because of grandfathering (e.g., an agency got the name before any official policy was established), or because the public is neither interested in nor knowledgeable about the government's departmental structure, and would simply be confused by domain names that reflect actual structure. Examples include:

faa.gov	rather than	*faa.dot.gov*
nist.gov	rather than	*nist.doc.gov*
cg.mil	rather than	*cg.dot.gov*

Figure 3-2. Domain Component Naming DIT



It may be fairly clear that the FAA should be a part of the Transportation Department, but does the public generally know that NIST is a part of the Commerce Department, or that the Coast Guard, a uniformed service, is actually under the Department of Transportation rather than the Department of Defense?

Another potential problem can be confusion between the government and the private sector because of the Top Level Domain Names. The U.S. Government only has authority over domain names ending with *.gov*. Sites such as *www.irs.com* and *www.fbi.com* play off of this confusion for purposes of social satire, political commentary, and worse.

And lastly, there is no automatic synchronization between X.500 and the DNS. When a domain component is registered in the DNS, it will require a second action to have it manually entered into the X.500 directory. This presents the potential for the X.500 or LDAP-based directory to get out of synchronization with the current state of the DNS. Within government, the changes are infrequent enough that this may be a manageable problem.

3.4 COMBINED DOMAIN COMPONENT NAMES WITH X.500 NAMES

Recently the Higher Education community, in a part of the Higher Ed, Internet II effort [13], has taken a slightly different approach to the use of domain component names, and asked the FPKI directory profile support this option. This community advocates combining domain component names with traditional X.500 names in the subjectName field of a certificate to enforce name uniqueness. This requires no new registration or management, and it may faciliate directory service discovery via DNS SRV records [14]. No rule in X.500 prohibits this, recent changes to the FBCA CP will also allow for this flexibility. New infrastructures are being designed in the Internet2/EDUCAUSE arenas to meet the needs of academia and a myriad of applications [13]. Allowing this flexibility will facilitate interoperability between institutions of higher education and the federal government, and foster the use of the FBCA model outside the US government.

The directory working group has discussed this proposal extensively and tentatively agreed to support this option as a reasonable basis for interoperable naming. The FBCA would stand up a directory server with 2 (or 3) roots for [o=US Government, c= US], [dc=gov], and, possibly, [dc=mil]. Agencies would be encouraged to include the combined name form in entity certificates and could choose whether to use [o=US Government, C= US] (Figure 3.3) or [dc=gov] (Figure 3.4) as the most significant part of their name. It would also be acceptable to use only one name form or the other (Figure 3.1 and Figure 3.2).

Using this scheme, some equivalent examples would be:

cn=John Smith, dc=irs, dc=treas, dc=gov, ou=Department of Treasury, o=U.S. Government, c=US

cn=John Smith, dc=irs, ou=Internal Revenue Service, dc=treas, dc=gov, ou=Department of Treasury, o=U.S. Government, c=US

cn=John Smith, ou=Internal Revenue Service, dc=irs, dc=treas, dc=gov, ou=Department of Treasury, o=U.S. Government, c=US

Or, starting with the ".gov" domain name:

cn=John Smith, ou=Internal Revenue Service, o=U.S. Government, dc=irs, dc=treas, dc=gov

cn=John Smith, ou=Internal Revenue Service, o=U.S. Government, c=US, dc=irs dc=treas, dc=gov

Figure 3-3. Combined Domain Naming with X.500 Names

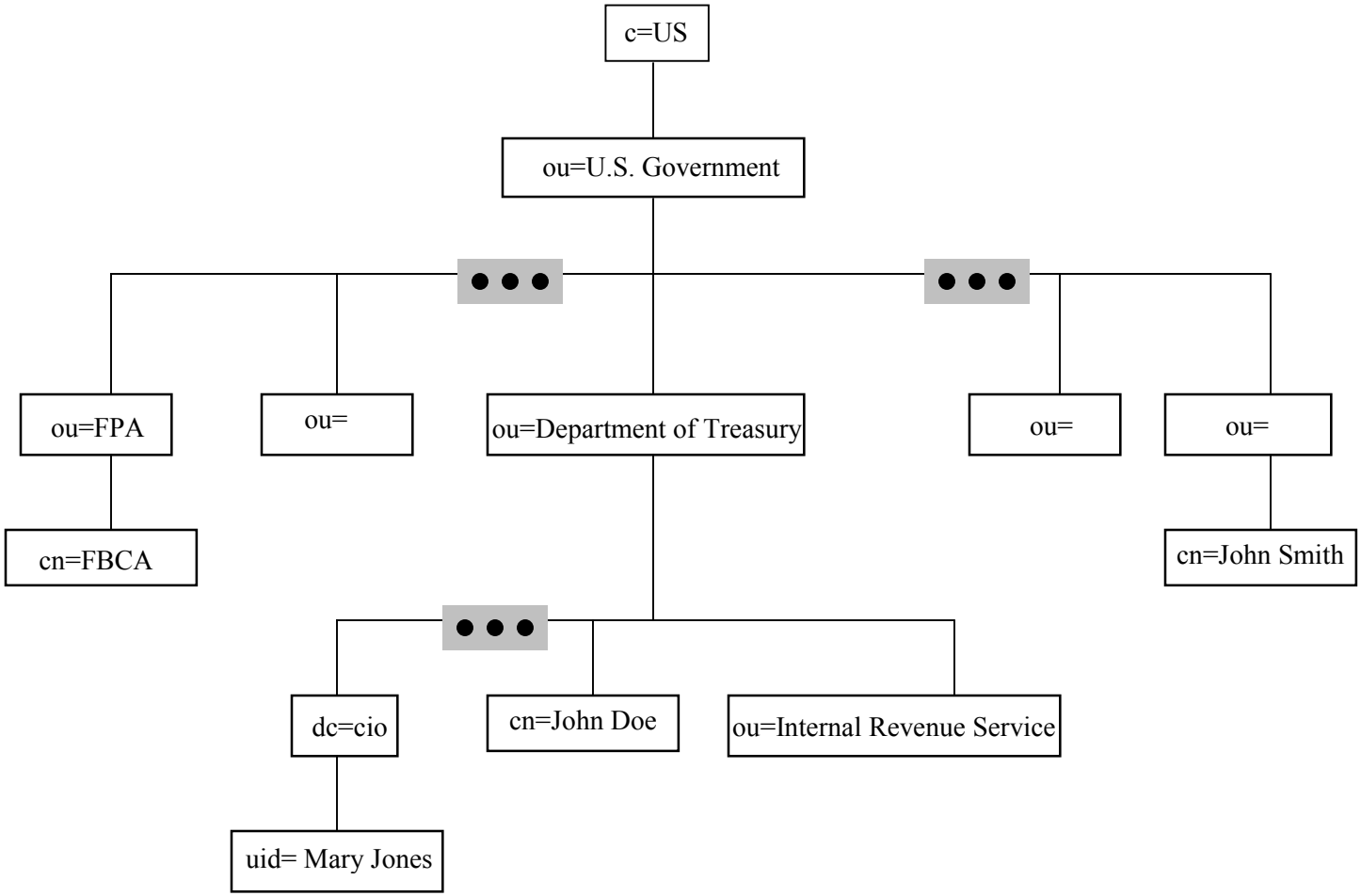
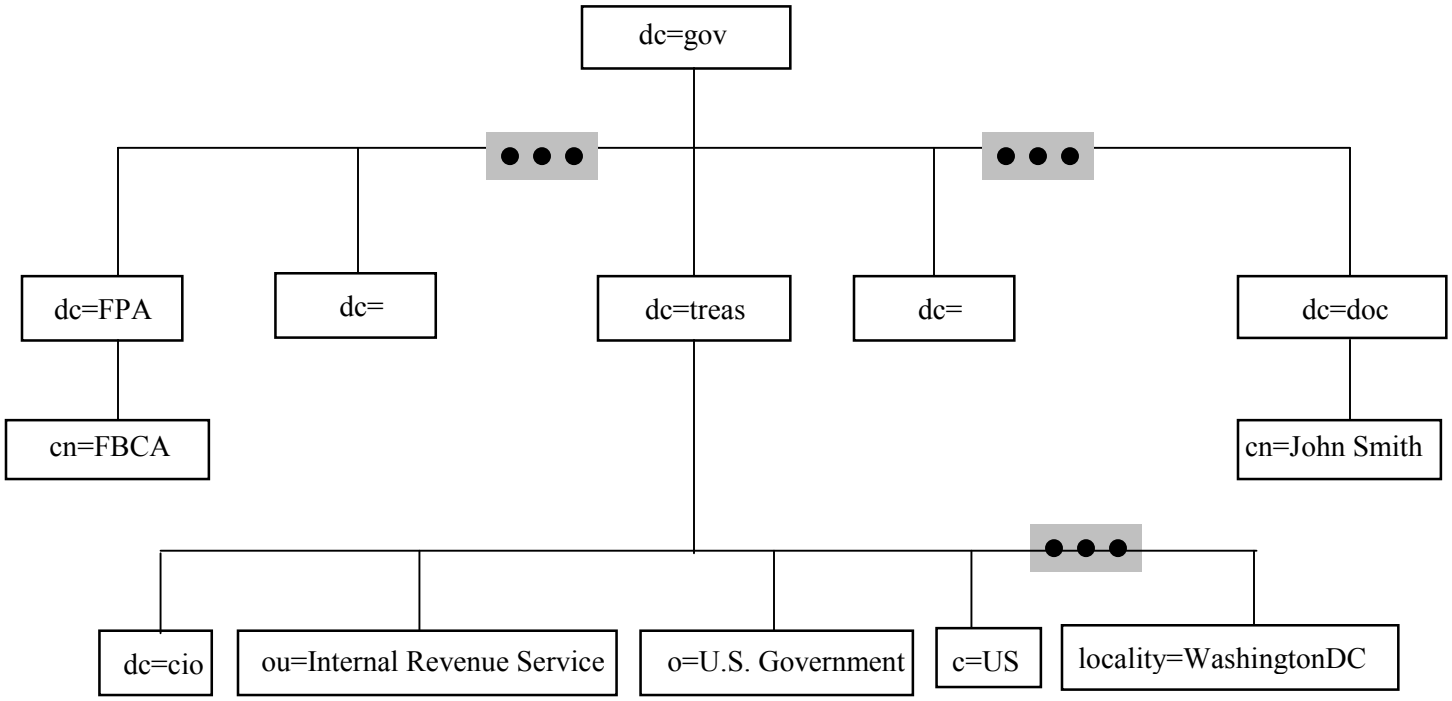


Figure 3-4. Combined Domain Component and X.500 Naming (alternate)



Several issues have been raised regarding this combined naming scheme. One is, do X.500 DSA products "object" to seeing the "c=" attribute subordinate to the "dc=". Are there other features of this naming scheme that "break" some directory products? What are the rules, if any, for formulating the combined names? For example all the names above start (on the right) with either the "c=US" or "dc=gov" attribute and end (on the left) with the common name. This makes sense intuitively, but does it make any difference to the processing of the name? These issues are yet to be explored.

3.5 THE U.S. GOVERNMENT DIRECTORY SERVER

In order to promote interoperability between various agency and department directory services, the Federal Bridge CA program will operate a Directory Server that supports both the Federal Bridge CA, and the U.S. Government level of the X.500 DIT.

In support of the U.S. Government level, the FBCA program will provide the following services:

- Registration of directory services for agencies that wish to (a) participate in the Bridge CA program, and/or (b) interoperate with other government directory services.
- The DSA will provide knowledge references to all registered directory services, and also to international government and the private sector, as required in order to promote Electronic Government initiatives.
- Coordination with E-gov and international interoperability initiatives.

The DSA will support the traditional X.500 DIT for the U.S. Government (Figure 3.1), the "de-facto" Internet DNS directory structure (Figure 3.2), as well as the hybrid DITs as illustrated in Figure 3-3 and Figure 3.4. It will be able to bridge among these namespaces, promoting interoperability among agencies that have implemented traditional X.500 naming, those that rely upon the DNS structure, and those supported both.

4.0 DIRECTORY PROTOCOLS

Two broad categories of directory servers are currently in use: "X.500 DSAs", and "LDAP servers." Both use the same X.500 directory information model and the LDAP v2 or v3 client directory access protocol. An X.500 DSA also supports Directory Service Protocol (DSP) chaining of directory servers. An LDAP server typically supports the LDAPv3 [15] client interface and LDAPv3 referrals. If chaining between LDAP servers is offered, it is almost certainly a proprietary implementation.

The FBCA will maintain an X.500 DSA, holding the roots for *c=US, o=U.S. Government, dc=gov,* and possibly, *dc=mil*. This FBCA DSA will be available for chaining to agency X.500 DSAs.

Although this profile does not preclude chaining internal directory servers to the FBCA directory server, most agencies will choose to operate with the Federal PKI through a border directory server located outside the agency firewall, as described in Appendix B.

For agencies that use X.500 DSAs for their directory service, or their border directory, it is not necessary to specify the client to directory server access protocol. Typically, it will be some version of LDAP, but the older X.500 Directory Access Protocol (DAP) is also acceptable. All that is required is that agency clients are compatible with agency servers. Agency servers will obtain needed external certificates and CRLs for their clients via DSP chaining, and this is transparent to the clients. Each agency border directory will be chained to the FBCA directory, via DSP chaining.

Agencies that choose to use LDAP servers internally may make external agency certificates available to clients in several ways:

- The agency may stand up an X.500 DSA as a border directory and chain it to the FBCA DSA;
- Alternatively, if agency clients support LDAP v3 with referrals, then the LDAP servers may refer clients to the FBCA DSA for external certificates (or may make direct referrals to the border directories of other agencies).

Agencies that choose to use LDAP servers internally may make internal agency certificates and CRLs available externally by:

- Standing up an X.500 DSA chained to the FBCA DSA and posting externally available certificates and CRLs to it. This may be achieved by purchasing directory services from a 3rd party supplier. This is the preferred or recommended method of interoperating with other agencies through the FBCA DSA;
- Alternatively, if no X.500 border DSA is set up, users may include a certificate list beginning with the certificate issued by the FBCA to their agency PCA and ending with the user's signature certificate in the header of signed S/MIME messages. This does not directly support encryption, but it allows an external relying party (who interoperates through the FBCA) to validate S/MIME signatures.

As the Federal PKI develops, the FBCA directory may incorporate an meta-directory capability, to transparently resolve the queries of X.500 DSAs for information contained in LDAP servers. This capability, however, will not be a part of the initial FBCA directory.

In principle, the choice to use X.500 style or Domain component names is independent of the choice to use X.500 DSAs or LDAP servers. In practice, it appears likely that those who choose to use domain component names will probably choose to use LDAP servers. It is possible to chain through the FBCA DSA from an agency that uses Domain Component names to one that uses X.500 style names. The

FBCA directory shall hold the root for both *c=US; o=U.S. Government* and *dc=gov*, and support chaining of both name types.

4.1 AUTHENTICATION REQUIREMENTS

Directories are required to support simple authentication for LDAP and DSP communications.

4.1.1 Client Authentication

FPKI directory clients that read the FPKI directory (read, list, search directory operations) require no authentication (i.e. anonymous bind to the directory is acceptable). This profile does not address directory access control requirements to update FPKI directory servers. Agencies must ensure that only authorized parties can update FPKI directory information.

4.1.2 Server Authentication

Initially, the FPKI directory service will not require authentication between agency servers and the FBCA directory server for DSP chained operations. The FBCA directory server is protected by a firewall that will be configured to allow only DSP operations between the FBCA directory and specified agency directories. Since the entries contained within the FBCA directory is public information, these firewalls will offer sufficient protection. The identity of LDAP clients querying the FBCA's LDAP directory server will be anonymous.

Future enhancements to the FPKI directory structure may allow for strong credential-based authentication between servers. However, the state of technology is such that this is capability is not possible at the present time given the fact that different vendors' products do not provide for seamless interoperability of security functionality.

4.2 DISCLAIMER

The FBCA directory service is being provided to promote full interoperability between government agencies, in support of the Federal Bridge CA. Every attempt will be made to ensure that information contained in the directory service is correct (as provided by the individual agencies), and that this information is protected from unauthorized access and modification. However, each agency or department must consider the possible consequences of unintended disclosure of information provided due to error or attack. It is the responsibility of each agency or department to establish their own policy and security posture with regard to directory-based information, and to implement whatever protocols and protection that they deem sufficient to protect critical systems, including their internal directory services.

APPENDIX A – NAME REGISTRATION WORKSHEET

If your agency is deploying an X.500 directory service and desires to use the X.500-style naming, you should register your directory with the FBCA Operational Authority (OA). To register, complete the following information and forward it to:

Name (FBCA Contact Information goes here)
Street Address
City, State Zip
Phone Number:
Fax Number:
Email address:

Agency Information	
Agency Name	
Mailing Address	
City, State & Zip	
Main Phone #	
Main Fax #	
Directory Administrator Information	
Administrator Name	
Mailing Address	
City, State & Zip	
Telephone #	
Fax #	
Email Address	
Alternate Contact	
Mailing Address	
City, State & Zip	
Telephone #	
Fax #	
Email Address	
Directory Service Information	
Type of Service	[ ] X.500 [ ] LDAP v2 [ ] LDAP v3 [ ] Other_____
Server Host Name	
Server IP Address	
Directory Port #	
DN of DSA entry (if applicable)	

Naming Context(s) and Protocols Supported	
Does this directory support chained operations using X.500 DSP?	[ ] Yes [ ] No
Does this directory support user access via X.500 DAP?	[ ] Yes [ ] No
Does this directory support user access via LDAP v2?	[ ] Yes [ ] No
Does this directory support user access via LDAP v3?	[ ] Yes [ ] No
Does this directory allow access from other Federal Agencies?	[ ] Yes [ ] No
Does this directory allow access from anonymous / untrusted users?	[ ] Yes [ ] No
What are the naming contexts supported by this directory server?	*e.g. ou=Bureau of XYZ, ou=Department of ABC, ou=U.S. Government, c=us*
Naming Context #1	
Naming Context #2	
Naming Context #3	
Naming Context #4	

APPENDIX B – ESTABLISHING AN AGENCY DIRECTORY SERVICE

Thus far, this document presumes that your agency already has an established organizational directory service that can be connected to the FBCA. If this is not the case, an official directory service must be established that can serve as the connection point between your agency, the FBCA, and other agencies. The general steps involved in setting up such a directory service are:

1. Decide which directory technology and/or product that your agency is going to support. It is highly suggested that you use either X.500 or an LDAP directory server. Proprietary directory services such as Active Directory or NDS may already be in use within your agency. If so, it should be possible to connect them to the FPKI directory service, but it may require that your agency implement a border DSA. Integrating directories other than X.500 and LDAP are outside the scope of this document.

2. Decide upon the naming convention that your agency is going to support, whether full X.500-based naming or Internet-style domain naming. A full discussion of these options can be found in Section 3. The FBCA supports both styles for agency naming contexts.

3. Register your directory service with the FBCA OA. Instructions for doing so can be found in Appendix A.

4. Plan the directory architecture – how many servers, where located, alternate / fallback service.

5. Plan the security architecture for the directory. The following sections of this appendix discuss security related threats and mitigation strategies that can affect the directory architecture.

6. Begin the process for registering for connection to the FBCA, as outlined in Appendix D of this document.

B.1 SECURITY CONSIDERATONS

All of the information contained within the FBCA directory server is considered public information. However, because agency directory services support operational requirements, they often need to contain information of a sensitive or For Official Use Only (FOUO) nature that should not be revealed to persons outside of the agency. Therefore, an agency may have issues with the security implications of chaining operations between their directory service and a directory service such as the FBCA, which supports anonymous access by a large number of unknown users.

Agencies are normally faced with conflicting goals with regard to an agency directory service. On one hand, they want their directory service to contain many different kinds of information and be readily available to all agency users who need access to that information. On the other hand, they generally want to identify a very small portion of their overall directory information as public information (phone numbers and email addresses, PKI certificates and CRLs, etc.), and they want to restrict access so that non-agency users cannot gain access to the rest of the information in their directory. At the same time, they want to protect their directory service from attack, denial of service, and unauthorized disclosure of information. And, they want to use the directory service to obtain needed information from other agency directory services.

Therefore, as agencies begin to allow connectivity with other directory services and access by non-agency users, they find that additional security capabilities must be added in order to provide accessibility and connectivity while ensuring survivability and availability, and protecting sensitive information from unauthorized disclosure.

B.2 LDAP VS. X.500

LDAP has a different security model than X.500. In LDAP, the client authenticates to the local server and this serves as proof of identity. The server uses that identity as the basis for all subsequent operations during that session. LDAP can use Secure Socket Layers (SSL) or Transport Layer Secuirty (TLS) in order to protect from unauthorized disclosure by encrypting the data flowing between the server and client. If this information were not encrypted, passwords and other directory information could be intercepted capturing the information flowing between the client and server using network "sniffers".

LDAP is a client-server protocol that allows user applications to retrieve and update directory-based information. It was originally based on a subset of the ITU X.500 recommendations, but has always been an Internet proposed standard. LDAP version 3 diverges from the pure X.500 in a few specific details, but still follows the X.500 "information model". Almost all directory-aware clients use the LDAP protocol to access directory services, and nearly every X.500 directory vendor provides an intrinsic LDAP server within their product. More accurately, this server is an LDAP-to-DAP gateway, converting the user's LDAP requests to X.500 query operations, converting X.500 responses to a series of LDAP responses, and sending the responses back to the client. Therefore, the client application doesn't know or care whether the directory being accessed is X.500-based, LDAP-based, or an Oracle database.

LDAP-based directory services (e.g. non-X.500) are becoming quite scaleable and robust, and are being implemented by the majority of federal agencies. It is fairly straight-forward to set up a large LDAP directory to serve an agency's user population. Unfortunately, having that directory server interoperate with another organization is not so simple, even if the other organization has implemented LDAP.

LDAP directory servers tend to be isolated islands of information. Users within the organization cannot easily access directory information within other agencies, whether LDAP-based or X.500. And, users outside of the organization cannot access information maintained in the organizational directory. While this may be an inconvenience when the directory information consists of email addresses and phone numbers, it is a severe problem when the information to be retrieved includes public key certificates and certificate revocation lists. Relying parties outside of the organization must be able to retrieve these objects in order to validate digital signatures, or to obtain encryption certificates.

Pure X.500 directory servers can also require that the client authenticate to the local X.500 DSA. In addition, each directory request carries the identity of the requestor. If the local DSA doesn't hold the requested information it will chain the operation onward. When the performing DSA receives the request, it can prove the identity of the requestor. In X.500, requests can be digitally signed, thereby providing non-repudiable proof of identity of the requestor. The DSA that performs the requested directory operation can check this digital signature in order to prove the requestor's identity, and can use that identity when enforcing access controls that may apply to the requested information.

This incompatibility presents a couple of difficulties related to trust when creating hybrid X.500-LDAP directory services:

- LDAP has become the universal client-to-directory access protocol, and LDAP clients cannot create signed directory requests. LDAP servers base their trust on the fact that credentialed authentication may have been performed between the client and server at initial bind. However, this trust in the client's identity is only held by the LDAP server and cannot be provided to another server as part of a chained operation. This is likely to change in future versions of LDAP, but it is not possible with LDAP v3 and earlier implementations.
- If the client is using LDAP to connect to an X.500 DSA, any chained requests forwarded from that DSA can contain the user's identity (DN). However, the requests cannot be signed because the DSA doesn't hold the user's private key (which is required to create a digital signature). Therefore, the performing DSA (the one that eventually does the requested operation) cannot trust

the user's identity. The security policy and at the performing DSA would normally treat such requests as untrusted, or anonymous.

This discontinuity in security generally leads agencies to implement additional security technology and techniques to protect the agency directory from unauthorized use and attack. These can include techniques such as compartmentalization, selective replication, border directories, and proxy servers – all of which are described in some detail below.

B.2.1 PKI Users With LDAP-Based Directories

In order to validate a digital signature, a PKI-aware client must construct a trust path, and must check to see that the certificate has not been revoked. The certificates and CRLs needed to perform these tasks are generally obtained via LDAP from a directory service. If the issuing authority and client are both within the same agency and served by the same directory service, this means that the client can simply issue repeated requests to the organizational directory until it has all the objects it needs.

However, if the signature was created by someone in a different agency, the PKI-aware client must construct a trust path that includes the Bridge CA, and must check the revocation status of a certificate that was issued by a completely different organization. Not only will the client need access to information in their own agency's directory service, but they will need information that is found in the Bridge Directory and in the issuing agency's directory service.

If the agency uses an X.500 directory infrastructure, this is relatively straight forward. The user simply queries their directory server using LDAP. The query is converted to X.500 DSP and chained to whatever X.500 directory holds the required information. The response is chained back to the agency directory, converted to LDAP, and sent to the client. The client is not even aware that the query and responses were automatically chained through multiple directory servers in order to satisfy their query.

But, if the agency uses a pure LDAP server, all it can do is return a referral. LDAP v3 implements a "referral" capability similar to that provided by X.500. If an LDAP server does not contain the information requested, it can return a referral pointing to another LDAP server that might be better able to respond to the query. The LDAP client can then choose to disconnect from the current LDAP server and try the one referenced in the referral instead.

However, many current LDAP clients are not yet able to follow LDAP referrals. Even if they were, every federal agency directory server would have to be configured with referral information about all the other agencies. This is sometimes referred to as the N*(N-1) problem. If you only have two agencies, two referrals are required. Three directories require six referrals. Four directories require twelve referrals. One hundred directories would require that 9,900 referrals be maintained. Some estimates place the number of directory services within federal agencies at over 1,000 – requiring nearly a million referrals, posing a bit of an issue with regard to scalability!

The initial operating capability of the FBCA directory service doesn't allow direct access by LDAP clients. When available, client requests could be converted to X.500 DSP and chained by the FBCA directory to other agencies. From that point, the client would not have to handle any further referrals unless the issuing agency was also using an LDAP-based directory service. In this manner, the FBCA directory service would become the defacto standard directory server to which the majority of agency referrals could point.

B.2.2 X.500 Access to Agency LDAP Directories

Certificates and revocation information must be obtained from the issuing agency in order to validate a digital signature. If the relying party's organization uses an X.500 directory and the issuing agency uses

an LDAP-based directory service, there is no way to chain the X.500 DSP queries to the agency's LDAP-based directory service. One possible approach would be for the agency to provide directory information such as certificates and CRLs to the FBCA, which would "publish" them, adding them to the FBCA directory base so other agencies could find them.

However, this will require that updated information be provided by the agency on a regular (probably daily) basis. Each agency providing this kind of information will have to convert their data to a standardized format (probably LDIF) so that it can be posted into the FBCA Directory.

Future enhancements of the FBCA directory service are planned that will better facilitate interoperability between LDAP and X.500-based directories within the FPKI directory environment.

B.3 TYPES OF THREATS

An agency's directory service should be designed such that it is resistant to common types of attacks. The most common types of threats are noted below.

B.3.1 Loss of Service

An agency directory should be available to the users and applications that rely upon it. Not only must it support the agency's own users, but it will be needed in order to obtain validate PKI-based digital signatures. The two basic issues to be addressed are availability and survivability.

Availability means that the directory service must be able to handle the expected usage load, and that the agency network infrastructure can reliably connect users to the directory. Strategies for ensuring availability include monitoring the directory service's performance and ensuring that network infrastructures are sufficiently robust, with fallback or failover capability.

Survivability means that the directory service is resistant to intentional attack or systems failure. Strategies include protecting agency systems with firewalls, compartmentalization (segregated networks for infrastructure components and servers), active monitoring, and distribution/replication of directory information across multiple systems.

B.3.2 Unauthorized Disclosure

As noted earlier, much of the information contained in agency directory services might be considered sensitive and therefore not suitable for access by unknown persons. Methods of gaining unauthorized access to directory information can include social engineering (usually by tricking support personnel to grant access to an untrusted party), bird-dogging (accessing an authorized user's terminal when they aren't aware), snooping (watching the data move across the network itself), spoofing (providing false identification and credentials to the directory service itself), and directly accessing the data held by the directory (usually by hacking into the network and gaining access to the directory server itself).

Social engineering and bird-dogging must be addressed by training both users and support staff. Snooping can be mitigated to some degree by using SSL or TLS to encrypt data flowing between LDAP clients and servers. If the agency directory service is X.500-based, communication between DSAs can be protected using link encryption or virtual private network technology in order to prevent snooping. Spoofing is a more difficult problem to prevent, and requires establishing a method whereby a user's identity can be proven by some sort of credentials (such as a PKI private key) before being allowed to access the directory service. X.500 (and some LDAP) directories can implement access controls that restrict access to information based on the user's identity.

B.3.3 Unauthorized Modification

FPKI information should only be modified by authorized parties within each agency. Each agency is responsible for ensuring that unauthorized modifications of the information in the agency directory do not occur – especially PKI-related information that will be relied upon by people outside that agency.

If FPKI information is to be extracted from an agency directory and provided to the FBCA, the agency must ensure that only public information is included in the extract. In addition, a method of securing the information (such as a digital signature) must be agreed upon between the agency and the FBCA OA. A digital signature would provide proof that the extracted information had not changed in transit.

Obviously, directory information can also be modified by unauthorized access to the computer system the directory is running on. This sort of data alteration may not be detected immediately, if the information is cached for performance reasons or accessed infrequently.

B.4 PROTECTION STRATEGIES

The following strategies can be employed to protect an agency directory service. Many strategies can help mitigate multiple threats. Often, multiple strategies will be employed in conjunction with each other in order to create stronger protection architectures.

B.4.1 Publication to Bridge CA

The following strategies can be employed to protect an agency directory service. Many strategies can help mitigate multiple threats. Often, multiple strategies will be employed in conjunction with each other in order to create stronger protection architectures.

B.4.1 Publication to Bridge CA

It is quite likely that some agencies will not allow unrestricted access to their directories, and are unable or unwilling to put up a border or "sacrificial" directory service. These agencies may ask the Bridge CA to publish this information for them. In this type of arrangement the agency will provide a file – probably in Lightweight Directory Interchange Format (LDIF) format – containing to the information to the Bridge Operating Authority on a regular (probably daily) basis. Automated scripts would extract this information and post it to the Bridge directory server.

This capability must be negotiated on a case-by-case basis with the FBCA OA.

B.4.2 Authentication and Access Controls

The LDAP v3 core standard provides for no access control capability. However, most vendor products offer some sort of access control – usually a subset or variant of the X.500-style Access Control Information (ACI) functionality. When selecting an LDAP server, you should ensure that you understand the method by which access control is implemented in the product you are considering.

Most directory servers support either anonymous access (no authentication) or simple authentication (passwords). Simple authentication provides only limited assurance of the user's identity because passwords can be guessed or intercepted by network snooping. SSL or TLS encryption should always be used when simple authentication is used.

Strong authentication uses credentials such as a PKI digital signature in order to establish the identity of the user. Both X.500 and many LDAP products can perform strong authentication of users, but most vendors' implementations are not compatible with each other.

Access controls are based on who you are (your identity) and what you are permitted to do (access rights). X.500 style access controls are based on a user's identity as expressed by the full distinguished name in the directory request. LDAP access controls are usually based on the user's identity or computer address, provided when the user first binds to the directory server. Using access controls, an agency can allow external users to view public information while restricting access to sensitive information such that only agency users can view it. Since access controls are enforced based on the user's identity, strong assurance (e.g. credentials such as digital signatures) are the only way to be assured that the user's identity has been proven.

Access controls are important to restrict unauthorized access to directory information, but they may not provide sufficient protection For instance, your agency may base access control decisions on the user's Distinguished Name in the directory request, and allowing organizational users to list and read all the information in the directory. If a bad player can create a bad directory request with a name that you trust, they can gain access to directory information. That request can come from anywhere on the Internet, so most organizations believe that it's a good idea to protect their directory from outside access by use of other techniques such as firewalls and border DSAs.

Most LDAP servers implement an inherited access control model. When access controls are implemented on a container, any objects further down in the directory tree will typically inherit the higher-level access controls. As an example, if you apply a policy that any anonymous user can read objects in the top level of the directory, all objects within this entire subtree will normally inherit this access control. It can be over-ridden further down the directory tree if needed. For instance, you might want to severely restrict access to a lower level of the directory. An access control statement applied to that container would over-ride the inherited access control definition set higher in the directory tree. Some LDAP servers ship with default access controls already defined, while others require you to define all access control information.

Replication brings another set of problems with regard to inherited access controls. Since inherited access controls flow downward from higher levels of the directory, some of the applicable access control information may not exist in the portion of the directory that is being replicated. The answer is generally to add additional replication agreements such that access control information is replicated in addition to the directory data.

B.4.3 Compartmentalization

Using X.500 it is possible to segment the agency directory into segments while keeping these segments connected into a logical agency DIT. The portions of the directory can be deployed on different network segments, separated by intelligent routers. For instance, if part of the directory tree contained public information, the directory server that held that portion might be located in the same network segment that holds the agency web server. If the directory server holding sensitive information is only to be accessible by agency users, it could be placed in the organizational network and routers configured such that directory traffic could not pass between the agency network and the Internet.

In the same manner, the "master" copy of the directory information could be held on a protected directory server located within a very secure network environment. Directory data can be updated by directory administrators operating with this secure network. When complete, the directory data would be replicated outward to a servers operating in the lower assurance network. The main disadvantage to this architecture is that directory updates cannot be accomplished by users. Requests for any needed changes must be sent to the directory administrators, the updates accomplished, and the modified information replicated outward to the production directory servers.

B.4.4 Encryption

If your agency is worried about interception of directory information traveling across the network, various forms of encryption can be employed. Not only can a bad actor discover information held by the directory, but they can also intercept information such as passwords, server network addresses, and chaining history (in X.500 requests) can be intercepted.

The original LDAP v3 specification lacks a definition of security services. To fill this gap, the Simple Authentication and Security Layer (SASL) was defined by John Myers in RFC 2222. SASL is a method for adding authentication support to connection-based protocols such as LDAP. In a SASL-protected session, the client issues an authentication command that includes a SASL mechanism name. Every SASL mechanism name must be registered with the Internet Assigned Numbers Authority (IANA), whose web site can be found at *http://www.iana.org*; RFC 2222 gives instructions for registering new authentication mechanisms with IANA. If the server supports the requested SASL mechanism, it initiates an *authentication protocol exchange* - a series of server challenges and client responses specific to that particular security mechanism. During this authentication protocol exchange, the client transmits the user's identity and negotiates for the use of a mechanism-specific security "layer".

The transmitted authorization identity may actually be different from the client's identity, to permit agents such as proxy servers to authenticate using their own credentials, followed by requesting access privileges belonging to the identity for which they are proxying. Once a security layer is requested / negotiated, it is used to protect all subsequent data sent between client and server.

Secure Socket Layers (SSL) and Transport Layer Security (TLS) are methods of encrypting information flowing between computer systems. SSL is the older of the two and is used extensively in securing access to sites on the World Wide Web. SSL encrypts the data carried within the "packets" flowing between the two computers. Any stream of information flowing across the Internet is actually busted into little chunks, called packets. These packets flow independently from the sending computer to the receiving computer. The receiving computer stores up the packets and re-assembles the data stream – all without the user's knowledge. The information carried in these packets for LDAP is text, and can easily be viewed by hardware and software tools known as "sniffers" – hence the term "packet-sniffing". SSL encrypts the information contained within the packets so that only the receiver can decode it.

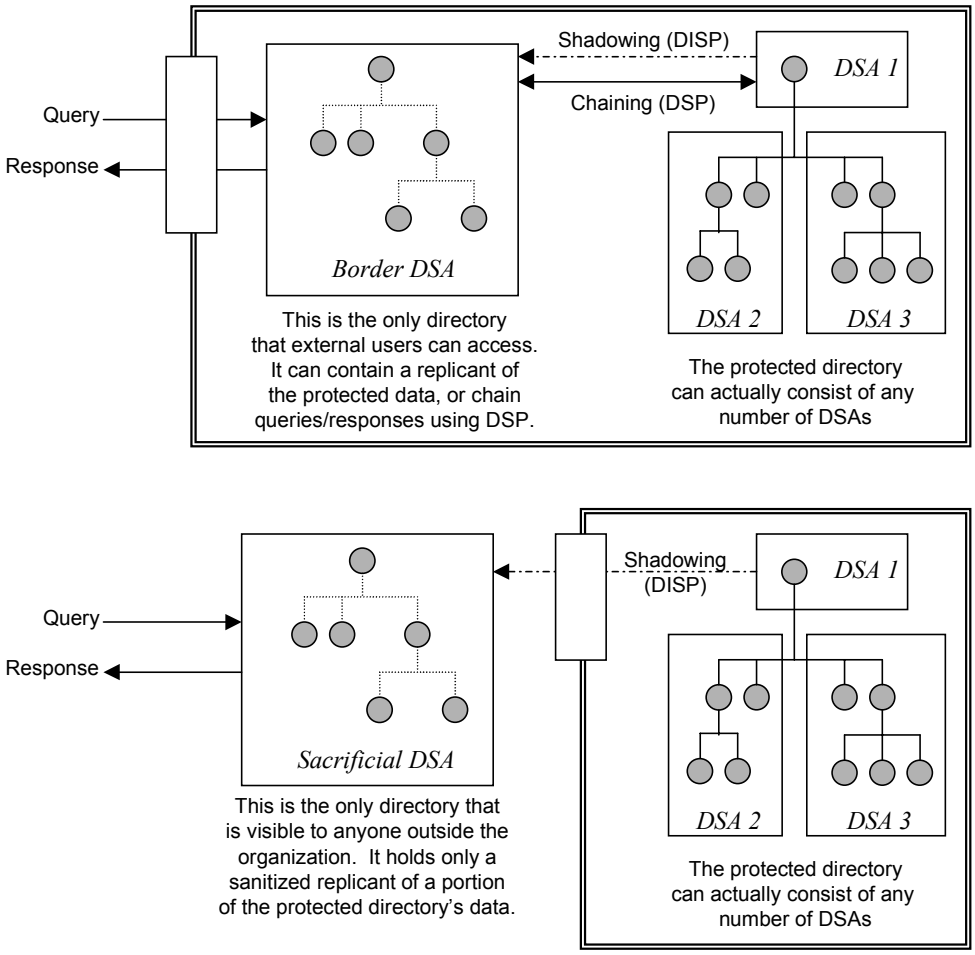
TLS is a relatively recent standard, and as of this writing has not been implemented in a great many products. It provides similar functionality to SSL, but at the transport layer rather than the packet layer. In other words, the data stream itself is encrypted in TLS before being broken into packets, whereas SSL breaks up the data stream first and then encrypts each packet.

Both SSL and TLS allow mutual authentication using strong authentication, and can use X.509-based certificates issued by various commercial PKI systems.

B.4.5 Border and "Sacrificial" DSAs

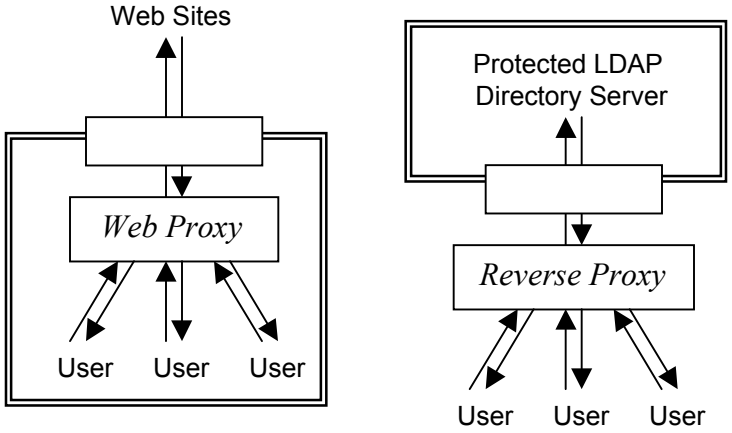
A Border DSA is an application level firewall that typically sits just inside the corporate firewall, or possibly in a DMZ (demilitarized zone) network segment with your email and web servers. External directory requests are received by this DSA, which either returns information based on the DIT that it holds (usually a subset of your overall directory information), or perhaps chains the query inward. You can use just about any X.500 product to create an effective Border DSA for civilian government agencies. Simply replicate information into the Border DSA, and do not configure it to chain queries inward. It will enforce access control information on the replicated entries, and return any applicable information from the replicant that it holds. The Border DSA should not master any part of the DIT.

Sacrificial DSAs are nearly the same as Border DSAs, except for two regards. First, they almost always exist outside the corporate firewall, albeit perhaps within a DMZ. Second, they usually receive a refresh of the replicant data that they hold on a regular basis, sometimes by a proprietary method (such as via FTP or an LDIF update). A Sacrificial DSA will normally assume that all requests are anonymous, and will not hold any information of a sensitive nature. If the Sacrificial DSA is attacked, it holds no sensitive information that would be of use to the attacker. Because it's data replication is one-way and it doesn't support chaining, the Sacrificial DSA provides no additional information that could be used to compromise the corporate directory service.



B.4.6 LDAP Reverse Proxies

LDAP presents significant challenges when securing the corporate network, because it does not (yet) implement any sort of chaining. Users from all over the world may expect to be able to directly contact your LDAP-based directory. If that directory exists within your corporate network, you will have to open your firewalls to allow this access. Most firewalls are not able to act as an application-level LDAP gateway. The only thing you could do to restrict access is to deny connectivity to all systems inside your organization except the LDAP server, and restrict access such that only LDAP operations and results can be passed between the LDAP server and users across the Internet. However, this configuration is still extremely worrisome to most security administrators, especially since most external LDAP access will be anonymous.



Traditional Proxy **Reverse Proxy**

A new technology, the *reverse proxy server,* addresses a great deal of LDAP's security concern in this type of environment. For a traditional service like Web access, a proxy server will sit inside the corporate firewall. Users are not allowed to access the Internet, but they *can* connect to the proxy server. The proxy server is allowed to connect to Web servers outside the corporate networks *on behalf of* the user. Responses come back to the proxy server, which then forwards them to the appropriate user. A reverse proxy places the proxy server backwards, outside the corporate firewall. In the event of a reverse LDAP proxy like the iPlanet Directory Access Router (IDAR), any user from the Internet can connect to the IDAR. A single hole through the corporate firewall allows the IDAR to connect to the corporate LDAP server to send LDAP operation requests and receive results.

APPENDIX C – CONNECTING TO THE FBCA DIRECTORY

This section briefly describes the steps that an organization must complete in order to connect to: the Federal Bridge Certification Authority (FBCA or "Bridge").

C.1 OVERVIEW

The Federal Bridge CA (FBCA) is operated by the Federal Bridge CA Operational Authority (FBCA OA) under the guidance and oversight of the Federal PKI Policy Authority (FPKI PA). The FBCA facilitates trust between your organization and other organizations by providing a certification path between your PKI and other government PKIs. This path is created by issuing cross-certificates between your organization and the FBCA, and connecting your organization's directory service to the FBCA's directory. This allows your PKI-aware applications to verify digital signatures created by certificates issued by other organizations, and to verify that those certificates are still valid (and haven't been revoked). Obviously, your organization must already have an operational PKI in order to make use of the Bridge.

C.2 WHERE TO FIND ADDITIONAL INFORMATION AND ASSISTANCE

Information on the Bridge, including the most current version of this Getting Started guide, will be found at *http://www.cio.gov/fbca*. Information on the FPKI Policy Authority will be found at *http://www.cio.gov/fpkipa*. Information on the FPKI Steering Committee will be found at *http://www.cio.gov/fpkisc*. Information on Internet standards can be found at *http://www.ietf.org*. A glossary of common security terms used in this and related documents can be found in RFC 2828 [1] (below).

C.3 DOCUMENTS

You will need the following documents, all of which are available on-line.

[1] Shirey, R. RFC 2828: Internet Security Glossary (May 2000). [Online] http://www.ietf.org/rfc/rfc2828.txt

[2] Federal PKI Policy Authority. X.509 Certificate Policy for the Federal Bridge Certification Authority (14 Jun 2001). [Online] http://www.cio.gov/fpkisc/ documents/fbca_cp_06-14-01.pdf

[3] Chokhani, S. RFC 2527: Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework (March 1999). [Online] http://www.ietf.org/ rfc/rfc2527.txt

[4] Federal PKI Policy Authority. Application for Interoperability with the Federal Bridge Certification Authority (TBD). [Online] http://www.cio.gov/fbca/docs/Application.doc.

[5] Federal PKI Steering Committee. Federal PKI Directory Profile (TBD). [Online] http://www.cio.gov/fbca/docs/DirectoryProfile.doc.

C.4 HOW TO GET CONNECTED TO THE FEDERAL BRIDGE CA

Your organization must complete the following steps before your PKI can be interconnected to the FBCA.

1. Create an Agency Certificate Policy (CP) – Your organization's formal Certificate Policy (CP) must be included in your *Application for Interoperability* [4] (see "Documents", above) with the Bridge. If you have an existing CP, you should review it for equivalence with the Bridge CP [2]. Your application must describe how the assurance levels in your CP map to those provided by the

Bridge. General guidance for creating a Certificate Policy and Certification Practices Statements can be found in RFC 2527 [3].

2. Create an Agency Certification Practice Statement (CPS) – Your CPS must provide specific details of how your PKI implementation meets the requirements stated in your CP. The CPS must be included as supporting documentation in your *Application for Interoperability* [4] with the Bridge.

3. Have Your PKI and CPS Audited for Compliance with Your CP – A third party must perform an independent audit of your organization's PKI implementation, to ensure that it meets the requirements set forth in your CP. The resulting Compliance Audit report must be included in your organization's *Application for Interoperability* [4] with the Bridge.

3. Submit Your Application for Interoperability to the FPKI PA – You must submit an *Application for Interoperability* [4] to the Bridge. This application must include your CP, CPS, Compliance Audit, and contact information for a A discussion of the information to be provided as part of the application follows in a later section of this document.

   After being reviewed by the FPKI PA for completeness, the application will be referred to the Federal Certificate Policy Working Group (FCPWG). The FCPWG will evaluate the application, work with your organization to resolve any Certificate Policy issues that might prevent a favorable recommendation for cross-certification, and return their recommendation to the FPKI PA. The FPKI PA will then approve the application and issue an authorization for your organization's PKI to be cross-certified with the FBCA at a specific Certificate Policy level.

4. Negotiate a Memorandum of Agreement (MOA) with FPKI PA – After your *Application for Interoperability* [4] has been submitted to the FPKI PA, you can negotiate a Memorandum of Agreement (MOA) between your organization and the FPKI PA. This MOA can be drafted during the time that the FCPWG is reviewing your application, but it cannot be signed and put into effect until the FPKI PA approves your application.

5. Perform Interoperability Testing with the Prototype Bridge CA – Once the application has been submitted, you can also begin to perform interoperability testing. The FBCA OA maintains a prototype FBCA facility to support interoperability testing, so that initial testing can be performed without adverse impact to the production FBCA. Your PKI cannot be cross-certified with the production FBCA until you receive approval from the FPKI PA. Your directory system may not establish interoperability with the operational BCA Directory System until interoperability testing with the FBCA prototype facility has been completed, and the FBCA Operational Authority authorizes the connection.

6. Conduct Live Test with the Production Bridge CA – After you receive approval, create an MOA with the FPKI PA, and perform initial connectivity testing, you can connect to the production BCA Directory Service and set up cross-certification between your Principal CA and the Bridge CA. Once your agency is connected, you will conduct some initial functional tests to ensure that everything works as expected. Then you can begin working with all the other organizations connected to the Bridge!

C.5 FILLING OUT THE APPLICATION

The *Application for Interoperability* [4] can be obtained from the FBCA website. After completing the application, you will submit it to the FPKI PA in written form or in Microsoft Word format to the Federal PKI Policy Authority. The information you must provide includes:

1. Organizational Information – You must include Organization Name and Address; Name, Title, Address and Contact Information for Designated Agent and Secondary Contact(s).

2. Certificate Policy – You must attach a copy of your organization's Certificate Policy (CP) to the application. Your agency CP must be in RFC 2527 [3] format. If your CP was not developed to this format, you must convert it to this format before submitting it to the FBCA Policy Authority.
3. Compliance Audit – You must describe how the organization PKI, the Principal CA, and any other CA that has a trust relationship with the organization PKI, is audited – including the frequency and the identity of the organization who performs the audit. You must attach a copy of your organization's latest PKI Compliance Audit, documenting your organization PKI's compliance with your CP. This audit must demonstrate that all aspects of the agency PKI Certificate Policy are being complied with, and must be conducted by an independent third party.
4. Certificate Policy Mapping – You must describe the mapping that your agency proposes between the certificate levels covered under your CP, and those set forth in the FBCA CP. You must explain the basis for the proposed mapping by comparing the two CPs and providing any other relevant information or justification.
5. PKI Information – You must provide information regarding the PKI that will be cross-certified with the FBCA. This information must include:
    a) You must provide information about the PKI system implemented within your organization, including:
        - PKI product being used,
        - Version implemented,
        - Signature algorithms supported, and
        - Encryption algorithms supported.
    b) You must identify at the Principal Certification Authority (CA) to be cross-certified with the FBCA. Information to be provided about this CA will include:
        - Distinguished Name (DN) of the Principal CA that will cross-certify with the FBCA,
        - The X.500 Name Space in which the PKI operates; and
        - Contact information for the manager of the Principal CA,
    c) If any CA with a trust relationship to the Principal CA provides certificates that assert object identifiers not covered in the organization's CP, you must identify those OIDs and provide a copy of the relevant CP under which those OIDs are defined.
6. Directory Information – Currently, the initial configuration of the Bridge supports interconnection to X.500-based directory services using the Directory Services Protocol (DSP). Your directory must support X.500 DSP in order for your PKI to interoperate with the Bridge. You must provide information regarding your organization's directory service, including:
    a) A statement regarding the level of conformance of your agency directory with the Federal PKI Directory Profile [5],
    b) DSA Distinguished Name, product, version, network address, and confirmation that the DSA supports1993 X.500 DSP;
    c) The naming context supported by this DSA, e.g., the X.500 “prefix” that identifies your organization’s directory information tree (DIT). For instance, Treasury has an X.500 naming context of: *c=us, o=U.S. Government, ou=Department of the Treasury*;
    d) Information about any secondary DSAs that also support that naming context;

e) The knowledge references that must be established between the BCA's directory and your agency's directory – e.g. cross, superior, or subordinate references – as noted in the Federal PKI Directory Profile.

f) Contact information for your directory and host system administrators.

Important note: If your organization's directory is *not* X.500 compliant, it will not be able to interoperate with the BCA Directory Service using the Directory Services Protocol (DSP). If so, you must utilize a Border DSA that has the ability to service directory requests from your users using whatever protocol is supported within your organization (such as LDAP or NDS) and can communicate with the FBCA using X.500 DSP.

C.6 TESTING WITH THE PROTOTYPE BRIDGE

After application has been made to the FPKI PA, you can begin interoperability testing with the prototype Bridge CA. This testing can proceed in parallel with the FPKI PA's approval of the application. The purpose of testing with the prototype Bridge CA is to identify and solve any technical or connectivity issues prior to connecting to the production Bridge CA. The steps involved include:

1. Establish and test network connectivity to the prototype FBCA facility. When this is accomplished, you will be able to *ping* the prototype BCA Directory service host computer from your primary directory server, and vice versa. Your technical people will work with technical staff from the FBCA OA to accomplish this task.
2. Next, establish and test connectivity between your agency's X.500 directory service (or Border DSA) and the prototype BCA Directory. This will involve configuring knowledge references between the prototype Bridge's directory and your directory service, and testing to ensure that directory queries and responses can flow between the directories using the DSP protocol. Your technical people will work with technical staff from the FBCA OA to accomplish this task.
3. Exchange cross-certificates between your Principal CA and the prototype Bridge CA, and install the Bridge CA's cross-certificate in your agency's directory.
4. Perform end-to-end testing to ensure that your PKI-aware application is able to verify digital signatures created by an application in the FBCA OA, and vice versa.

C.7 CONNECTING TO THE PRODUCTION BRIDGE

You can establish interoperability with the production Bridge CA after the FPKI PA has approved your agency for connection to the Federal Bridge CA, and you have executed a MOA with the FBCA OA. To connect to the production Bridge CA, you will perform essentially the same steps as when you set up interoperability with the prototype Bridge CA.

APPENDIX D – REFERENCES

[1] Burr, W., "Public Key Infrastructure (PKI) Technical Specifications: Part A-Technical Concept of Operations", September 1998

[2] The Evolving Federal Public Key Infrastructure, Federal Public Key Infrastructure Steering Committee, Federal Chief Information Officers Council, gits-sec.treas.gov.

[3] Governmentwide Directory Support 2 Technical Series, the Updated US Gold Schema document, 7/14/1997, by Booz Allen & Hamilton.

[4] The Bridge CA Demonstration Repository Requirements Draft 4/8/1999 by Chromatix, Inc.

[5] NSA Bridge Certification Authority Demonstration Phase II - Directory Requirements and architecture, 7/3/2000, by Entigrity Solutions.

[6] Boeyen, S., Howes, T., and P. Richard, "Internet X.509 Public Key Infrastructure LDAPv2 Schema", RFC2587, June 1999.

[7] ITU-T Recommendation X.509 (1997) | ISO/IEC 9594-8: 1997, “Information technology - Open Systems Interconnection - The Directory: Authentication framework”, June 1997.

[8] ITU-T Recommendation X.521 (1997) | ISO/IEC 9594-7: 1997, “Information technology - Open Systems Interconnection - The Directory: Selected object classes”.

[9] Common Directory Services and Procedures, ACP (Allied Communication Publication) 133 Edition B, March 2000.

[10] M. Smith, "Definition of the inetOrgPerson LDAP Object Class", RFC 2798, April 2000.

[11] Kille, S., Wahl, M., Grimstad, A., Huber, R., and S. Sataluri, "Using Domains in LDAP Distinguished Names", RFC 2247, January 1998.

[12] Grimstad, A., Sataluri, S., and M. Wahl, "Naming Plan for Internet Directory-Enabled Applications", RFC 2377, September 1998.

[13] The Middleware Architecture Committee for Education (MACE) http://middleware.internet2.edu/MACE/

[14] Armijo, M., Esibov, L., Leach, P., and R. Morgan, “Discovering LDAP Services with DNS”, Work in Progress.

[15] Wahl, M., Howes, T., and S. Kille, "Lightweight Directory Access Protocol (v3)", RFC 2251, December 1997.

APPENDIX E – ACRONYMS

ACP	Allied Communications Publication
c	country
CA	Certification Authority
cn	commonName
dc	domainComponent
DIT	Directory Information Tree
DN	Distinguished Name
DNS	Domain Naming System
DSA	Directory Service Agent
DSP	Directory Services Protocol
EE	End Entity
EMA	Electronic Messaging Association
FBCA	Federal Bridge Certification Authority
FOUO	For Official Use Only
FPKI	Federal Public Key Infrastructure
FQDN	Fully Qualified Domain Name
IETF	Internet Engineering Task Force
LDAP	Lightweight Directory Access Protocol
o	organization
OA	Operational Authority
OID	Object IDentifier
ou	organizationalUnit
PAA	Policy Approving Authority
PCA	Policy Creation Authority
PKI	Public Key Infrastructure
PKIX	Public Key Infrastructure (X.509)
RDN	Relative Distinguished Name
RFC	Request For Comment
S/MIME	Secure Multipart Internet Messaging Extensions
SSL	Secure Socket Layer
TLS	Transport Layer Security
uid	userID
X.500	ITU specification for directory services