DRAFT 101500

Report of
Federal Bridge Certification Authority
Initiative and Demonstration

Electronic Messaging Association
Challenge 2000

DRAFT 101500

DRAFT 101500

1.0 Introduction

This report describes the results of the EMA Challenge 2000, a demonstration of Public Key Infrastructure (PKI) interoperability using the Federal Bridge Certification Authority (FBCA). The report also provides an overview of Federal PKI efforts and how use of the FBCA is intended to support efficient, seamless interoperability of different agency PKI domains and ultimately, external PKI domains as well. A more detailed treatment of all Federal agency uses of public key technology can be found in *The Evolving Federal PKI*, published in June 2000 and available electronically at http://gits-sec.treas.gov. This report assumes that the reader has some understanding of PKI technology. Appendix 1 contains background information on that subject for those who desire it.

2.0 Background

The Federal PKI will support secure and authenticated unclassified transactions over open networks like the Internet, thus promoting e-commerce, e-government, and critical infrastructure protection. In particular, the Federal PKI will help Federal agencies conduct electronic transactions with other Federal agencies, with other levels of government (state, local and foreign), with trading partners in the private sector, and with the general public. .

Federal efforts to use public key cryptography generally begin with individual applications within agencies that provide immediate benefits of improved service delivery, efficiencies, and cost savings. Thus, the Federal PKI will not be a monolithic top down structure; it will be created largely from the bottom up. Agency efforts generally are paid for out of program funds, not funded as a centralized government PKI initiative. The challenge facing the Federal PKI is to meld the individual agency initiatives that use PKI products from a variety of commercial vendors, into an integrated PKI that is interoperable internally as well as with state and local governments, foreign governments, businesses and the general public.

3.0 Federal PKI Landscape

Within the Federal government, substantial efforts are already underway to deploy public key technology for intra- and interagency applications, especially those involving personnel matters, contracts, and financial transfers. These efforts include implementing agency public key infrastructures providing the full range of services needed to issue and manage digital certificates: Registration Authorities to identity-proof users, Certification Authorities to issue certificates, repositories to distribute certificates and certificate revocation lists, and key recovery agents to allow the recovery of encrypted data if the private encryption key is lost.

A wide range of PKI products and services exists supporting such enterprise-wide needs. As yet, these products do not universally support interoperability if different brands are employed between enterprises. Since the Federal PKI is developing from the bottom up, with agencies picking disparate products and services suited to their needs, a complex environment is emerging in which to achieve interagency interoperability.

Agencies generally justify the use of public key technology in terms of improved efficiency, reduced costs, and improved service delivery. In some cases, agencies will purchase and run their own PKI domain; in other cases, agencies may have a contractor fulfill that function; and in still others, agencies many only purchase PKI services. Thus, the landscape over which interoperability must be accomplished is complex and variegated.

"Metcalfe's Law," which states that a network becomes more valuable as it reaches more users, also applies to a PKI. It is apparent that there are great benefits to a system that propagates trust not just in the local environment, but throughout the entire Federal government, and further, that establishes a framework that can interoperate with trust domains throughout the nation, and the world. Trust in a PKI can propagate through certification paths. The main issue for the Federal PKI is this: Given that many, often quite different, systems that use certificates are now being implemented by Federal agencies, how do we create certification paths between them, in a consistent and coherent fashion, to allow reliable and broad propagation of trust?

3.1 Operational Concept

The FBCA will be the unifying element to link otherwise unconnected agency CAs into a systematic overall Federal PKI. The FBCA is not a root CA. It does not start or end certificate trust paths, it simply connects *trust domains* through cross certificate pairs to "Principal CAs" which are designated by each agency. Thus, the FBCA is a "Bridge of Trust." A Federal PKI Policy Authority (FPKIPA) will oversee FBCA operation and establish the requirements for an agency to cross certify with the FBCA. Ultimately, trust domains that are outside the government will be able to interoperate with agency PKI domains using the FBCA.

Initially, Federal agency CAs that operate in trust domains that meet the requirements established by the FPKIPA will be eligible to cross-certify with the FBCA. This will then connect them to the overall trust network of the Federal PKI, and provide relying parties and certificate holders in their trust domains with connectivity to the larger Federal PKI. This will be simpler and more effective than trying to manage an ad hoc collection of many peer-to-peer cross-certifications among agency CAs.

The FBCA will maintain a directory that contains certificates it has issued. The FBCA initially will issue Certification Authority Revocation Lists (CARLs). CARLs are Certificate Revocation Lists for certificates issued to CAs, which for the FBCA are the Principal CAs cross-certified with it. The total number of certificates issued by the FBCA will be modest, since the total number of agency Principal CAs is unlikely to be large. The number of FBCA certificates is likely to remain small even after the FBCA begins to cross-certify with PKI domains external to the Federal government.

The Federal PKI will support hierarchical, mesh and trust list PKI architectures – that is, agencies may use any of those architectures within their own PKI domain. Further, agencies will not be required to use the FBCA to interoperate within or outside the Federal government. Rather, they may go to the party with whom they want to interoperate and cross-certify directly. However, the FBCA simplifies interoperability among Federal agencies and ultimately with the private sector. Thus, the value of the FBCA is expected to grow as e-commerce and e-government activities expand.

3.2 Assurance Levels

The X.509 standard includes mechanisms for specifying policy information in a certificate in the certificatePolicies extension. X.509 does not impose any particular definition of policy, and so the meaning of policy information may vary between PKIs. In some PKIs, the policy information denotes privileges or intended application, which is useful in locally defined applications, but is unlikely to be meaningful in inter-agency applications. In some cases, however, the policy information denotes the level of assurance that is associated with a particular certificate, which is important for inter-agency applications.

The FBCA is designed to connect Federal agency PKIs to support a broad range of applications. As a result, FBCA certificates will convey assurance-level type policy information which will be meaningful to inter-agency applications. FBCA certificates will specify one or more of four different levels of assurance: Rudimentary, Basic, Medium, and High. A full description of each level can be found in the FBCA Certificate Policy available through http://gits-sec.treas.gov. These four levels are intended to meet the Federal government's requirements for trust establishment across security domains. In addition, this strategy is closely aligned with the certificate policy adopted by the Government of Canada (GOC), promoting ultimate interoperability between those PKIs.

Certificates issued by the FBCA will contain at least one of those assurance level policy Object Identifiers (OIDs) in the certificatePolicies extension. Certificates issued by agency CAs are likely to assert different policy OIDs reflecting CPs that are unique to each agency. As described below, the FPKIPA will map agency-specific levels of assurance to the levels of

assurance present in the FBCA CP; that mapping will be expressed in the policyMappings extension of the FBCA cross-certificates.

3.3 Trust Domains

In the Federal context, a trust domain is a portion of the Federal PKI that operates under the management of a single policy management authority or equivalent body; this will typically cover a single agency or a subordinate element of an agency. One or more CAs may exist within the trust domain. Each trust domain has a single Principal CA, but may have many subordinate CAs. Each trust domain has a domain directory and at least one CP.

3.4 Architecture Components

3.4.1 Federal Public Key Infrastructure Policy Authority (FPKIPA)

Any infrastructure which cuts across multiple agencies requires the cooperation of the affected agencies to make it work. The Federal PKI is no different. While agencies may run their own agency-specific PKI domains to serve their own agency-specific needs, interoperating with other agencies imposes unique requirements and obligations.

The model of governance reflects the fact that the Federal PKI has evolved from the bottom-up, from agencies adopting this technology to serve their specific needs rather than having its use prescribed for them. In 1996, the Federal PKI Steering Committee was formed under the Government Information Technology Services Board, co-chaired by OMB and the National Partnership for Reinventing Government (NPR). The Steering Committee, comprising over 50 members representing over two dozen agencies, has as its focus the promotion of interoperable PKI solutions, the development of common guidance, and the sharing of information so that agencies considering or deploying PKI solutions can benefit from those who have already done so. Participation in the Steering Committee is voluntary. Its activities are published at http://gits-sec.treas.gov.

Beginning in mid-1998, the Steering Committee developed a model for governance of the Federal PKI. This model is best described as "governance by the governed." In other words, those agencies employing public key technology would determine collaboratively how best to ensure they could interoperate efficiently and seamlessly. The model envisions the creation of a Federal PKI Policy Authority, which would have as its initial membership agency representatives to the Steering Committee. The FPKIPA would serve to establish the conditions under which an agency-specific PKI would interoperate with other agency-specific PKIs using the FBCA. That is, the FPKIPA would map the certificate policy of each agency to the FBCA certificate policy, thus allowing an agency to determine whether a certificate

from another agency contains the level of assurance or trust needed for a particular transaction. This model avoids each agency having to develop bilateral relationships and certificate policy mappings with every other agency; instead, that is done once with the FPKIPA.

In February 2000, the GITS Board announced that its activities will be merged with those of the Federal Chief Information Officers (CIO) Council, and the GITS Board will be disestablished. Accordingly, in June 2000, the FPKIPA was established under the auspices of the Federal CIO Council. The six charter members of the FPKIPA are GSA, the Office of Management and Budget, and the Departments of Defense, Treasury, Commerce, and Justice.

To summarize, the responsibilities of the FPKIPA include:

- Approving the Certificate Policy for the FBCA. This function will support interoperability between the FBCA and Federal agencies initially, and then with external parties as well
- Approving the Certification Practice Statement for the FBCA
- Determining the assurance level(s) at which an agency Principal CA may interoperate with the FPKI through the FBCA by comparing relevant CPs, CPSs, and other material submitted by the agency
- Defining which certificate policies and policy mappings to include in certificates issued by the FBCA to agency Principal CAs
- Providing additional support, advice, and assistance to Federal agencies in the management of their internal agency PKIs, when requested

3.4.2 Federal Bridge Certification Authority (FBCA)

Federal PKI interoperability could be accomplished in several different ways – through the use of CA trust lists, through hierarchical relationships, and through the use of a Validation Authority – but the model best suited to Federal agencies is represented by the FBCA. The FBCA acts as a non-hierarchical "hub." Agency CAs would receive permission from the FPKIPA to interoperate with the FBCA under terms that were mutually negotiated and accepted. Each CA that interoperates with the FBCA would be able to interoperate with every other using the certificates that the FBCA issues. It is useful to describe this process.

When a user (the "recipient") receives a digitally signed transaction from another user (the "sender"), the recipient's application software must do three things as it attempts to verify the signature on the transaction. First, the recipient's software must determine whether a trust relationship exists between the PKI domain that issued the certificate, and the PKI domain of the recipient; this can be done by establishing a so-called "trust path" of certificates between

those two domains. Second, the recipient must determine that the policy expressed in the sender's certificate meets the needs for the transaction at hand – i.e., does the certificate contain the requisite level of assurance? Finally, the recipient must determine that all of the certificates in the trust path are valid, that is, that they have neither expired nor been revoked.

If the sender and recipient were in the same domain, these three steps would be straightforward to execute because the transacting parties share the same trust anchor (Principal CA) and the same certificate policy. That is, the recipient understands the "quality" of the certificate offered for the transaction. If the recipient and sender are from different agencies, this process is more complicated. In addition to creating the certificate trust path using the FBCA, the recipient needs a mechanism to understand how the assurance level in the incoming certificate "maps" to the assurance levels understood in the recipient's domain. This is also done using the FBCA, since each certificate the FBCA issues contains a mapping between the levels of assurance honored by the FBCA, and those honored by the agency to which the certificate was issued. This mapping is established by the Policy Authority at the time that an agency applies to interoperate with the FBCA.

It must be emphasized that when an agency acts as a relying party (that is, when it is determining whether to accept a certificate issued by another agency), it is not required to use the FPKIPA mapping. It may employ whatever mapping it determines appropriate. This preserves agency autonomy. Moreover, the FBCA can be adjusted to accommodate a "trust list" approach, by having the FBCA digitally sign and post one or more such lists. This would permit a hybrid model that is likely to accommodate a broader spectrum of commercial products. (Note: This functionality does not exist in any current product).

Lead responsibility for designing, implementing and operating the FBCA resides with the Federal Technology Service of GSA. The Steering Committee, NSA, NIST, and Mitretek Systems provide technical and programmatic oversight. The FBCA is coming into existence in two phases. In the first phase, the FBCA has been implemented as a prototype, which went operational for testing purposes on February 8, 2000. The prototype was used for the EMA Challenge testing described in this report. The prototype has two CA products supplied by Cybertrust (now Baltimore Technologies) and Entrust, which are cross-certified within the FBCA membrane (which is the boundary between all of the CAs that form the FBCA, and the external CAs with which the FBCA interoperates). (Subsequent to the EMA Challenge, the Cybertrust CA was replaced with a Baltimore Unicert CA reflecting the fact that Baltimore acquired Cybertrust.)

The production version of the FBCA will be built using the same architecture, but including additional CA products within the membrane so that full interoperability is supported with any CA product or service an agency may select for its use. Indeed, this is the unequivocal

goal of the FBCA: whatever CA product or service an agency selects, they will be able to interoperate using the FBCA. Subject to approval of funding requested in the FY01 budget for this purpose, the production FBCA should be operational by late 2000.

3.4.3 Principal Certification Authority

A Principal CA is a CA within a trust domain that cross-certifies with the FBCA. Each trust domain has one principal CA. In the case of a domain with hierarchical certification paths it will be the root CA of that domain. In a mesh organized domain, the Principal CA may be any CA in the domain. However, it will normally be one operated by, or associated with, a Domain Policy Management Authority or equivalent body.

3.4.4 Agency Certification Authority

An agency CA is one that is subordinate to an agency Principal CA within an agency PKI hierarchy trust domain. If the agency trust domain is not hierarchical, then an agency CA is any CA within the domain other than the Principal CA.

3.4.5 Root Certification Authority

In a hierarchical trust domain, the Root CA is at the top of the hierarchy and is an "anchor" upon which all trust paths begin or end. In the hierarchical domain, certificate holders and relying parties are given the self-signed root CA certificate, by some authenticated, out-of-band means. For hierarchical trust domains, the root CA is also the Principal CA for that domain.

3.4.6 Subordinate CA

A subordinate CA is a CA in a hierarchical trust domain that receives a single certificate, that from its superior CA; it may also have subordinate CAs of its own to which it issues certificates.

3.4.7 FBCA Directory

The FBCA directory will be open to Internet access by anyone, and will make the following available:

- All certificates issued by any node of the FBCA
- All certificates issued to any node of the FBCA
- All cross certificate pairs containing certificates held or issued by the FBCA

- A CARL from each node of the FBCA covering the certificates issued by that node
- Other certificates and CRLs as determined by the FPKIPA

Directories are on-line facilities that provide certificates and certificate status information. Directories in the Federal PKI will provide information via X.500 DAP or LDAP; the FBCA directory will support both protocols. Directories that contain end-entity certificates and CRLs for end-entity certificates are established and run separately by each agency's individual trust domain.

3.4.8 Certificate Status

An important part of certification path processing is confirming that certificates have not been revoked or suspended. Certificates may be revoked for a number of reasons including changes in the names of individuals, reorganizations that change organizational names, the subject has left the organization or changed their job, any attributes bound to the subject in the certificate may have changed, or because of a known or suspected key compromise. Two standardized mechanisms are available for determining current status, CRLs and OCSP responders.

Users of the Federal PKI will rely on the FBCA CARL (which is a CRL for certificates issued to CAs, such as those issued by the FBCA to agency Principal CAs) to determine the status of certificates issued by any node of the FBCA. If the FBCA directory contains CRLs published by other CAs, that directory may also serve as a "one stop" mechanism for validating the current status of any certificates issued by CAs in the Federal PKI. FBCA issued certificates are expected to be more stable and of longer validity than end-entity certificates, so CARLs should be fairly small. The FBCA repository is expected to be a key resource for creating and validating certification paths, and will have high availability requirements, medium bandwidth requirements, and low storage requirements. The contents of the FBCA directory may be shadowed or replicated in other directories, however, and doing so will make the FBCA model highly resistant to denial of service attacks. This point is discussed further below.

3.4.9 X.509 Certificate Policy Processing

The certificatePolicies extension in Federal PKI certificates will be used to identify the policy that applies to a certificate. The certificatePolicies extension will contain the OID corresponding to an assurance level policy stating the highest level of trust supported by the certificate. The certificatePolicies extension will also contain the OIDs of all the lower assurance levels that the certificate also satisfies. For example, a certificate issued under a Medium assurance policy will also contain the policy identifiers for the Basic and

Rudimentary assurance policies, because the medium assurance certificate should meet the requirements of the basic and rudimentary assurance policies. The certificate would then be acceptable to a relying party who specified *rudimentary* assurance, *basic* assurance, or *medium* assurance, but not to a relying party who specified only *high* assurance. The certificatePolicies extension may also contain other specific policy identifiers that apply to the certificate. (Note that processing of the certificatePolicies extension was not demonstrated for the EMA Challenge 2000 but will be included in further testing. In order for agencies to make constructive use of the FBCA, applications will generally need to process this extension.

4.0 FBCA Demonstration at the EMA Challenge

4.1 Background

For the EMA Challenge, the prototype FBCA was used to support an interoperability test with six disparate PKI test domains, with digital signatures on S/MIME e-mail as the application. The interoperability test is described in detail below.

4.2 Objective of Demonstration

The objective of the demonstration was to show the ability of secure e-mail applications to interoperate across disparate PKIs by creating and validating trust paths using certificates issued by the FBCA. The test application employed digital signatures, but the concepts demonstrated are applicable to circumstances requiring encryption. That functionality, in addition to certificate policy mapping, is being tested in follow-on efforts. Specific goals included:

- Demonstrating that COTS S/MIME products, either out of the box or with specific modifications, can discover (create) certificate trust paths over complex topologies and of substantial length, and then process those trust paths.
- Demonstrating that five different PKI CA products (Entrust, Cybertrust, Motorola, Spyrus, and CygnaCom (prior to its acquisition by Entrust)) can interoperate (cross-certify) to provide a framework for certificate trust path creation and validation.
- Demonstrating the directory functionality required to achieve trust path creation and validation using four different X.500 directory products.

All of these functionalities must be demonstrated in order for the FBCA concept to be workable. The demonstration successfully did so, as is discussed further below.

4.3 Test Setup and Architecture

The architecture for the EMA Challenge included the following:

FBCA: Entrust CA and Cybertrust CA, cross-certified within the membrane; PeerLogic X.500 Directory System; firewall and Internet connectivity for directory

Domain CAs:

- DOD Bridge Demonstration Cygnacom CA, cross certified with the Entrust node of the FBCA. Under the CygnaCom CA, there were three separate PKI domains, each cross certified with the CygnaCom CA
  - one using three hierarchically arranged Spyrus CAs
  - one using three hierarchically arranged Motorola CAs
  - one using four meshed Entrust CAs
- FTS/GSA Cybertrust CA, cross certified with the Cybertrust node of the FBCA
- Georgia Tech Research Institute CA, cross-certified with the Entrust node of the FBCA
- One NIST Entrust CA, cross-certified with the Entrust node of the FBCA
- Second NIST Entrust CA, cross-certified with the Cybertrust node of the FBCA
- Canadian Government Entrust CA, cross-certified with the Entrust node of the FBCA
- National Aeronautics and Space Administration (NASA) Entrust CA, cross-certified with the Entrust node of the FBCA

Client Components:

- S/MIME e-mail clients (Eudora enabled with Entrust and other plug-ins; Microsoft Outlook enabled with Entrust plug-ins)
- Certificate path discovery, validation, and S/MIME v.3 libraries developed by CygnaCom and J.G. Vandyke

Directory Components:

- PeerLogic I500 Directory for the FBCA itself, as well as the NIST and GTRI domains
- Nexor Directory in the Canadian Government PKI domain, chained to the FBCA directory
- Chromatix Directory for the DoD domain
- Control Data Systems Directory for NASA

4.3.1 PKI Overview

Figure 1, PKI Overview, illustrates the PKI architecture of the participating EMA Challenge CA domains. The DoD BCA, two NIST CAs, Canadian Government CA, NASA CA, Georgia Tech Research Institute CA, and GSA/FTS CA are all cross-certified with the FBCA. There is an additional CA depicted, which was not cross-certified with the FBCA. Messages to/from a client in this CA domain were used to demonstrate the lack of interoperability across CA domains, when cross-certification with the FBCA is not in place.

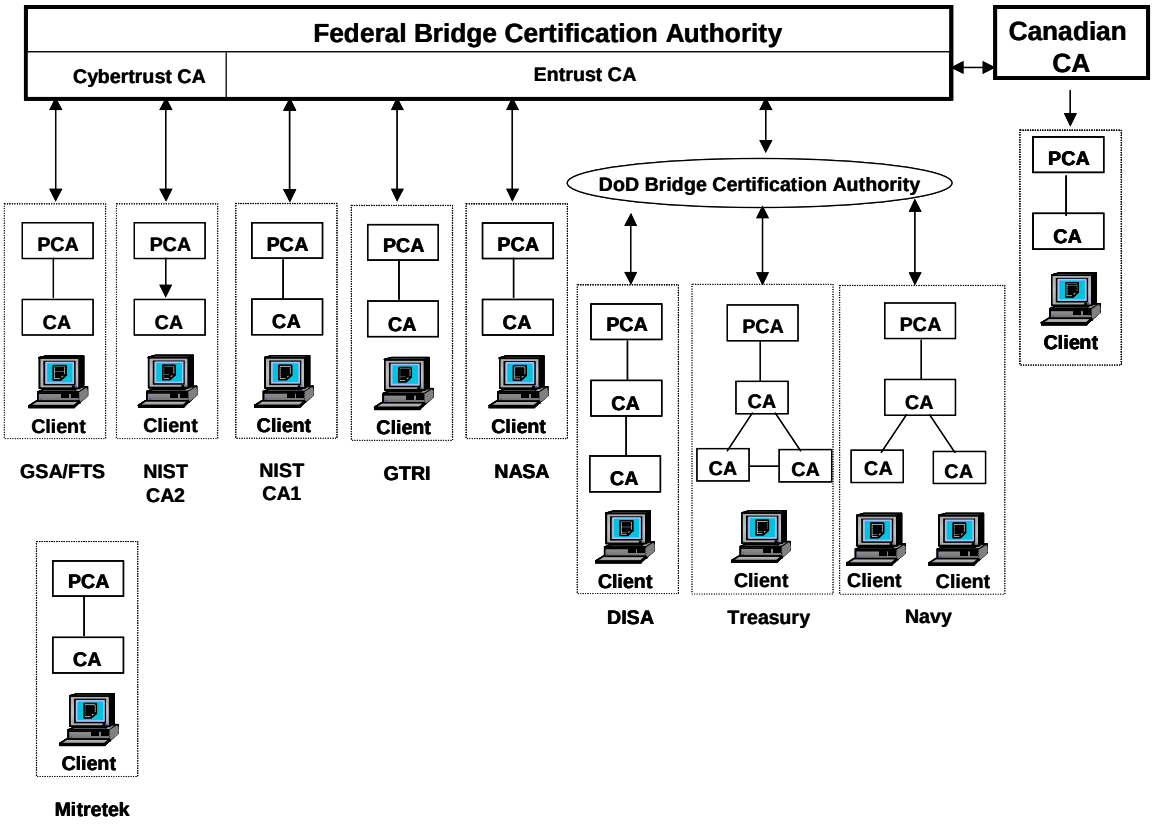

Figure 1. PKI Overview

4.3.2 Directory Chaining Schema Overview

Figure 2, Directory Chaining Schema Overview, illustrates the directory chaining architecture and directory tree schema information used to allow the client software to build the certificate validation trust paths.

The participating directories were added to the DSA (directory system agent) registry in the FBCA Peerlogic directory. The IP address, port, and DSP transport selector (if applicable) information was contained in these listings. The directory was configured for two way polarity, available association, initially started for chaining, and trusted for authentication. Then the directories were chained using cross-references. Some of the directories needed more than one cross-reference because of their schema. Twelve cross references were created. One directory needed three cross-references alone.

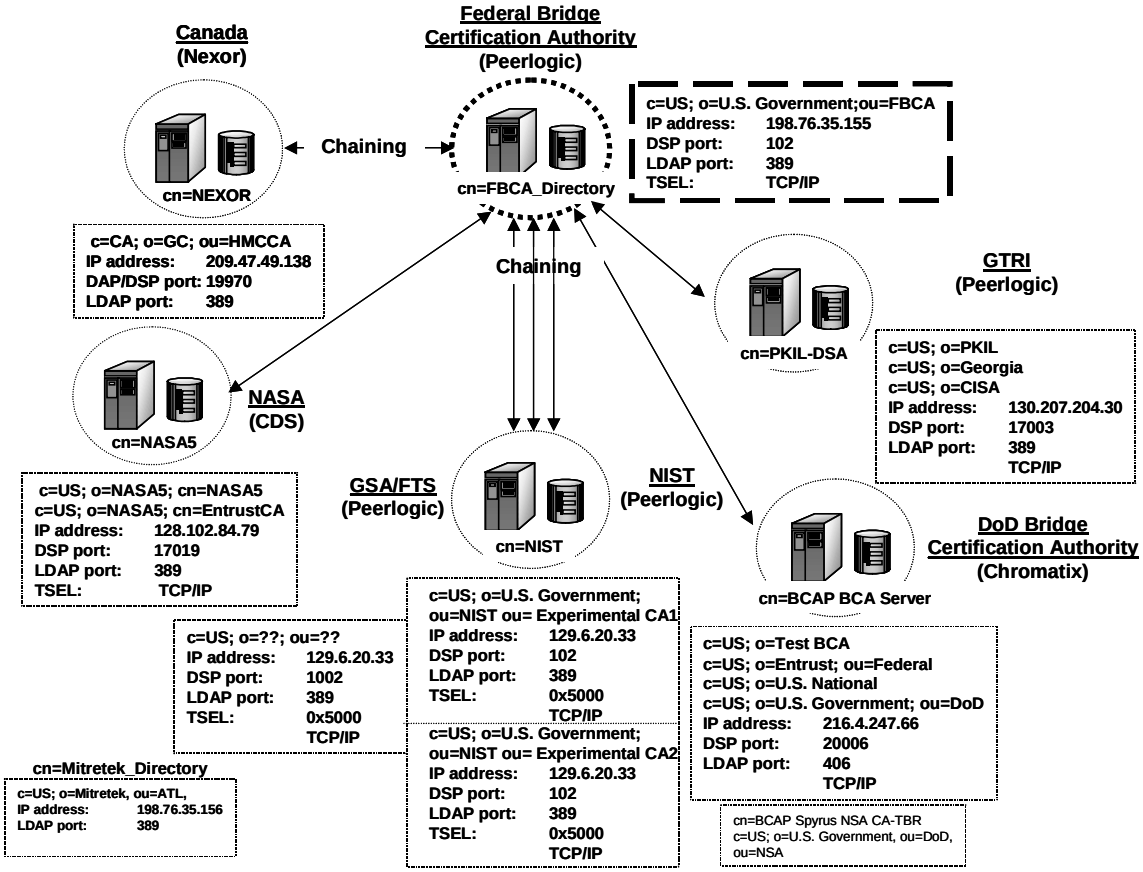

Figure 2. Directory Chaining Schema Overview

4.3.3 Network Overview

Figure 3, Network Overview, illustrates the network connectivity for the EMA Challenge. The laptop computers hosted two clients from each participating CA domain. One of the users has a valid certificate and the other has a revoked certificate (refer to Tables 2-4 for client number information). As cited earlier, one of the domains deliberately was not cross-certified with the FBCA.

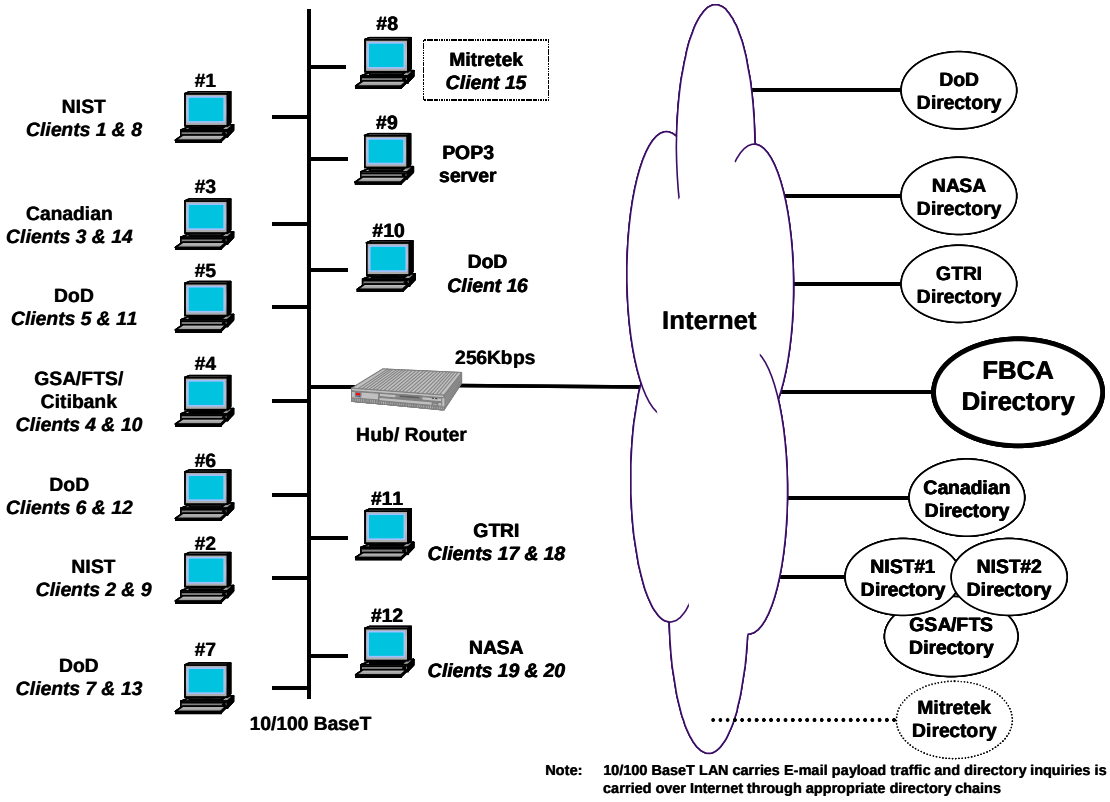

Figure 3. Network Overview

4.4 Demonstration Scenarios

The scenarios tested during the demonstration included:

- Signature verification using all valid trust paths that traverse the FBCA, in both directions

- Signature rejection based on selected end-entity subscriber revocations
- Signature rejection based on an end-entity subscriber of an Agency CA that is not cross-certified with the FBCA

4.5 Results of Testing

Table 1 describes the results of the testing performed during the month prior to the EMA Challenge April 6-8, 2000, and further testing done subsequent to the Challenge. Domains were considered to be interoperable if clients from one domain were able to establish and validate certification paths for clients in the other domain.

The original goal of two interoperable domains connected using the FBCA, was easily surpassed. At the time of the demonstration, five domains interoperated through the FBCA. These domains were NIST CA #1, DoD, Georgia Tech Research Institute, Government of Canada and NASA. The DoD domain was particularly complex, with its own single-product bridge CA connecting two DOD hierarchical domains, and one DoD mesh domain. One additional domain (NIST CA #2) had been established, but had very limited interoperability at the time of the demonstration owing to a certificate configuration problem discussed below. Only one portion of the DoD domain interoperated with NIST CA #2, and only in a single direction.

Because of a lack of time to complete preparation of a plug-in for the commercial e-mail client software, interoperability using the GSA domain could not be accomplished. It should be emphasized that this was not due to any technical problem, but simply to a lack of time.

Soon after the EMA Challenge conference, a problem was discovered with the certificates issued by one of the FBCA nodes which explained why some domains were unable to interoperate. Specifically, the Cybertrust node was inserting a “non-critical basic constraints” extension identifying the subject as an end entity. This was an inadvertent error unrelated to the product – the basic constraints extension should have identified the subject as a CA.

The architecture was also tested with revoked end-entity certificates, and with certificates issued by a CA that was not cross-certified with the FBCA, in order to test for false positive results. Successful interoperability between domains was only declared if there were no false positives, or false negatives.

DRAFT 101500

From To	NIST CA#1	NIST CA#2	DOD Entrust	DOD Spyrus	DOD Mot.	Canada	GTRI	NASA	GSA
NIST CA#1	NA	DEB							CUD
NIST CA#2	DEB	NA	DEB	DEB	DEB	DEB	DEB	DEB	CUD
DOD Entrust		DEB	NA	**NA**	**NA**				CUD
DOD Spyrus			**NA**	NA	**NA**				CUD
DOD Mot.			**NA**	**NA**	NA				CUD
Canada		DEB		DEB	DEB	NA			CUD
GTRI		DEB					NA		CUD
NASA		DEB		DEB	DEB			NA	CUD
GSA	CUD	CUD	CUD	CUD	CUD	CUD	CUD	CUD	CUD

Table 1. Results of Messages to and from Each Domain

Where CUD: Client Under Development

DEB: Debug

Green: Tested successfully

NA: Not applicable (the trust paths were properly processed but did not traverse the FBCA)

As of the time of the EMA Challenge conference, the paths shown as “DEB” were not working owing to the error cited earlier with the Cybertrust CA. Once that problem was

corrected subsequent to the Challenge conference, these trust paths were also demonstrated to work.

4.6 Lessons Learned

The EMA Challenge, like any grand experiment, had its share of surprises and lessons-learned. Here is a list of the most significant ones:

- The FBCA concept is sound, and the EMA Challenge demonstrated that the concept works. The FBCA concept is a viable tool for connecting PKI domains into a larger PKI. In this demonstration, PKI domains were implemented with three different PKI architectures: a single CA, a hierarchical PKI, or a mesh PKI. Clients from each of the PKI architectures could establish the validity of certificates issued in any other PKI architecture. That is, a client from a hierarchical domain could establish certification paths for end-entities in a mesh or a single CA PKI domain, and so on.

- Cross certification of different CA products is nontrivial but can be achieved with a few supplementary tools. To complete cross certification, CA operators employed two custom tools. The first tool created a cross certificate pair given two CA certificates. The second tool added, deleted or replaced specific values in an LDAP directory entry. These tools were required to address voids in the COTS products.

- Directory chaining can pose significant demands for an agency seeking to use the FBCA. Directory chaining was required for this demonstration; the client software depended upon the X.500 directory to find certificates and CRLs in remote directories. (Note: this is not a requirement for use of the production FBCA; the production FBCA directory will support both chaining and LDAP with referrals.) The directory architecture required a single chaining agreement for each agency directory – a chaining agreement with the FBCA director. The single chaining agreement was not expected to present serious problems. However, in practice, effecting the chaining agreement was often more difficult than cross certification. Four main problems were encountered: overlapping directory information trees (DITs), schema problems, heterogeneous products, and knowledge gaps. In the first case, directories refused to chain because they had been defined with overlapping DITs. For example, both the NIST and FBCA directories were set up for C=US. While their contents did not overlap, the directories could not be chained until the DIT overlap was corrected.

- The X.500 directory system did not present a performance bottleneck. The directory architecture selected for this demonstration (described above) conceded performance to gain management convenience. This architecture required just five chaining agreements.

In theory, chaining each directory to the other five directories would result in maximum performance and would preclude a successful Denial of Service attack against the central directory. The more complex architecture was rejected based on the difficulty of establishing fifteen chaining agreements. Nonetheless, the results were encouraging. In the worst case, path validation was completed in fewer than 30 seconds. This was achieved despite performing approximately forty LDAP retrievals (client queries used LDAP). While 30 seconds is a long time to wait for every signature validation, a trust path, once created, is cached, permitting future signature validations between the same transacting parties to be done in under one second. This point is discussed further below.

- Client ability to develop and process trust paths using an X.500 directory is straightforward to implement. Two different client implementations were used to find certification paths. These implementations were radically different. One started with a trust point and worked "forward" until it completed the path. The other started with the end-entity certificate and worked "backwards" until the path was discovered. Both strategies worked. Which of the two is optimal will only be determined through further testing, and it is unlikely that a single one will be optimal for all settings.

- The path validation procedure described in X.509 is ambiguous regarding processing of non-critical extensions. The designers of the two client implementations made two radically different assumptions regarding non-critical extensions. One ignored the basicConstraints and keyUsage extensions when they were marked non-critical. The other processed any extension it recognized, regardless of criticality. Each interpretation can be justified by citing portions of the 1997 X.509 standard. (A defect report has been submitted in an attempt to clarify this issue. The proposed resolution to the defect report is to change the text to clearly state that an implementation must process any extension that it recognizes, regardless of criticality.)

- The details are important to successful PKI deployment, but are easy to overlook. For example, as cited earlier, two CA certificates were issued with an inappropriate basic constraints extension. This extension should have been marked critical, and should have indicated the certificate subject was a CA. Instead, the extension was marked non-critical and indicated the subject was an end entity. This problem was not discovered until a detailed review was done of the certificate contents.

4.7 Plans for Further Efforts

The EMA Challenge 2000 effort was simply a beginning. It built upon the success that DOD achieved in their Bridge Demonstration PKI effort. Further efforts are planned in four areas:

validating paths for key management certificates, testing additional functionality, developing a production FBCA, and encouraging enhanced PKI support in COTS products.

The EMA Challenge focused only on digital signatures. In theory, path validation procedures are identical for end entity certificates that contain key management keys. This will be confirmed by demonstrating encryption based on key management certificates.

The FBCA design depends upon the X.509 certificate policies and policy mappings extensions to convey policy information. The FBCA design depends upon the policy constraints and name constraints extensions as well. These features were not employed in the EMA Challenge, but will be explored in preparing for use of the production FBCA. In addition, the FBCA will need to deal with mixing signature algorithms in certificates.

While a great deal has been learned from a prototype and in a test environment, many issues will only emerge in a production setting, particularly those issues that relate to scaling and administration. Thus, the learning process will continue as the production FBCA is deployed.

Finally, a PKI is not implemented for its own sake. A PKI exists solely to support security services in applications. Today, applications that are PKI-ready are the exception, not the norm. With a few notable exceptions tested in the EMA Challenge, applications that are PKI-ready generally are not able to take advantage of the features offered by the FBCA. Vendors must be encouraged to add or enhance the PKI functionality required to be a full participant in the interoperable PKI that the FBCA provides.

5.0 Implications of Test Results for Use of the Production FBCA

5.1 Security Issues

Three security concerns have been expressed about the FBCA concept: (a) if a miscreant can compromise the FBCA, that individual can create a trust relationship with a bogus CA and thus fool legitimate CAs that are cross-certified with the FBCA into trusting the bogus CA; (b) since all of the FBCA-issued certificates and CARLs will appear in the FBCA directory, a denial of service attack against that directory could have serious impact; and (c) the FBCA supports transitive trust relationships which could expose relying parties to undesired consequences if the trust chains are too long or complex, or if the parties in the chain do not behave in accordance with the agreements they reached with the FPKIPA. Fortunately, each of these concerns either lacks a technical basis, or can be ameliorated through careful design of the certificates issued by or to the FBCA.

Compromising the FBCA is extremely difficult and would require collusion among insiders who actually operate the FBCA. This is because the FBCA operates off-line (without networking), in a multi-person controlled environment. Creation of cross-certificates is done manually with floppy disks as the medium for exchange of certificate requests and responses – each of which is a digitally signed object. The FBCA nodes will be shut down most of the time; the only times they will be powered up will be to issue a new cross-certificate – which will be a very infrequent event – or to issue a CARL, which is expected to be a weekly event. The private signing keys for each FBCA node as well as other aspects of the CA equipment will be under multi-person control. Thus, taking all of these factors into consideration, the opportunity for compromise is extremely remote given the nature of the FBCA itself.

Attacking the FBCA directory is also likely to be unproductive. This is because all agencies who have their principal CAs cross-certified with the FBCA will be required to replicate the cross-certificates and CARL contents of the FBCA directory in their local agency directories, and if X.500 chaining is employed, to have more than one chaining arrangement established so that there is no single "master" X.500 directory. All of this should not be burdensome because the FBCA directory's contents are modest in size. Local replication means that an attack against the FBCA directory would fail to stop the agency from being able to obtain and use all of the FBCA certificates and CARLs – which are the only things needed to create and validate trust paths. If an attack against the FBCA directory were persistent such that CARLs held locally were no longer within their validity period (note that this would require a persistent attack over many days), backup plans would call for the FBCA operational authority to create and send updated CARLs to each agency separately. This too would not be very burdensome because such updates could be sent on floppy disks or as e-mail attachments without fear of modification since they are digitally signed objects.

It should be noted that even if the FBCA functionality were completely eliminated owing to an attack or natural disaster, there would be no effect upon agency PKIs continuing to be able to perform their functions within their agencies. Only interagency interoperability would be affected until the FBCA functionality were reestablished.

The issue of undesirable transitive trust is one which can be addressed through the use of constraints placed into the certificates issued to or by the FBCA. For example, if an agency does not wish to accept any certificates from one or more other agencies, it can place into the certificate it issues to the FBCA an "excluded subtree" in the nameConstraints extension field. This has the effect of causing client software that is attempting to validate a trust path containing a certificate from one or more of the excluded agencies to fail in that effort. Additionally, constraints can also be imposed on path length and other elements. In short, the X.509 standard provides ample capability for an agency to limit its exposure; at the same

time, it also places a burden on the agency to ensure that the client software it purchases for creating and processing certificate trust paths is fully compliant with X.509 and the Federal Certificate Profile, which requires applications to recognize (and properly process per X.509) these extensions.

5.2 Quality of Service

The FBCA model must support real-time transactions which do not impose undue burdens on the transacting parties. Waiting 30 seconds for every digital signature on an e-mail message to validate would be unacceptable. Fortunately, however, that is unnecessary. Rather, the user need only wait 30 seconds for such validation the first time he or she deals with an individual. Thereafter, the user need only wait less than a second for every subsequent transaction with that
individual. As the EMA Challenge testing demonstrated, even in complex PKI topologies, the time required for signature validation can be very short as long as trust paths are cached so that repetitive dealings with an individual become straightforward. Indeed, once a trust path is cached, any individuals who use the trust anchor can expect a fast response. This point is discussed further below.

5.3 Flexibility

One of the hallmarks of the FBCA approach is its fundamental flexibility – that is, its ability to connect disparate PKI domains, for client software to use multiple directory protocols (DAP, LDAP) to access information required for trust path creation, and for client or other software to evaluate certificate status and perform policy mapping. This can be done for e-mail (S/MIME) and web-based applications. A central question, however, is how well does the model work in connecting PKI domains with minimal if any changes required within those domains?

Ideally, an agency would like to be able to connect its PKI domain without having to make any changes. Realistically, that is plausible if the agency has adhered to open standards, has employed a common Federal directory schema, has used the Federal certificate profile in issuing end-entity and subordinate CA certificates, and has followed the efforts of the FPKI Steering Committee so as to keep abreast of impending changes to any of those elements. If the agency has done none of these, then it is less likely to be able to fit its PKI domain seamlessly into the interoperable space supported by the FBCA.

Since agencies are already standing up PKI domains, it is useful to discuss briefly the two most important elements that agencies need to contemplate now in order to facilitate ultimate

interoperability using the FBCA. These elements are informed by the results of the EMA Challenge.

The first element is directory interoperability or, at least, harmonization. While the clients tested during the Challenge employed LDAP calls to the directories within their own domains to obtain certificates and CRLs/CARLs, X.500 chaining was employed between all of the directories so as to avoid the need for the client software to perform the additional queries that would be necessary were LDAP referrals being employed. Thus, a useful hybrid LDAP/X.500 approach was employed, where the client software did not need to implement DAP. This approach can also make use of LDAP with referrals once that capability matures. Regardless of whether LDAP with referrals or X.500 chaining is used for directory interoperability, it is important that consistent directory schema are employed, that the Directory Information Tree (DIT) for each agency contain certificate and CRL information in common locations, and that naming conventions used for CAs honor a consistent mechanism. On the last point, it is interesting to note that in issuing certificates, agencies may choose to employ either an X.500 distinguished naming convention, or the Domain Component naming scheme used ubiquitously in the Internet, in populating the subject field or subjectAltName extension. Client software should be able using either naming convention to fashion the necessary directory calls to obtain certificates, CRLs, and other information necessary to create and process trust paths.

To help agencies accomplish directory interoperability, the Technical Working Group of the Federal PKI Steering Committee is developing a "directory profile" which will be made available to any agency seeking to interoperate with the FBCA. Agencies meeting the provisions of the directory profile can have substantial confidence that their directories will work within the FBCA scheme.

The second element entails ensuring that Federal agency certificates and CARLs/CRLs follow minimal interoperability standards. For example, poorly constructed certificate or CARL/CRL extensions could cripple an agency's ability to have its certificates accepted by other agencies.

5.4 Optimization

The FBCA concept is susceptible to many optimization mechanisms which are expected to greatly improve the performance of client software in creating, validating and processing certificate trust paths. Several examples are discussed below.

As configured for the EMA challenge, the FBCA is optimized for user-based applications. Most of the tests were performed in a cache-less environment, so every message validation

required construction of the complete path. This scenario should be experienced only infrequently. Even so, the delays experienced by users were not excessive.

Users may experience greater delays as the Federal PKI becomes more complex and directories operate under high load. When trying all possible scenarios with an empty cache, one of the client systems performed forty directory retrieval operations to develop a seven certificate path. While this took just a few seconds, a PKI with 200 CAs (rather than twenty) obviously is a more challenging environment. This risk can be mitigated by improving path development algorithms and developing smart certificate and CRL caching mechanisms.

Network administrators in the federal government may also wish to employ PKI-enabled services to secure network protocols and the network infrastructure itself. The performance requirements for these applications are more stringent than those imposed by users, and may require optimizations that were not tested in the FBCA EMA Challenge.

For some applications, a wheel and spoke directory structure may create an unacceptable bottleneck. Many requests for directory attributes will exploit two (or more) chaining agreements. This can be reduced with chaining agreements between agency directories, rather than relying solely on the FBCA directory. It may be even more efficient to use the Domain Name System (DNS), LDAP referrals, or pre-established certification paths to accelerate PKI services for network components.

5.5 Certificate Path Service

The EMA Challenge tested certificate trust path creation and validation using libraries accessed by the e-mail client software. In principle, these functions, as well as certificate trust path processing (for policy mapping), could be accomplished in a centralized fashion using a server. Specifically, the e-mail client, when presented with a certificate issued outside its domain, could provide the certificate to a central server that has as its sole purpose the creation, validation and processing of certificate trust paths. That server could also cache trust paths on a user-by-user basis. Doing this reduces the complexity of the client software and allows more efficient caching of trust paths between domains where they are shared by multiple users. Of course, such a service does not exist today, but if demand is sufficient, there is no conceptual reason it could not be developed.

DRAFT 101500

Appendix I: Introduction to PKI Technology

Public key technology provides a mechanism to authenticate users strongly over closed or open networks, ensure the integrity of data transmitted over those networks, achieve technical non-repudiation for transactions, and allow strong encryption of information for privacy/confidentiality or security purposes. Strongly authenticating users is a critical element in securing any infrastructure; if you cannot be certain with whom you are dealing, there is substantial potential for mischief. Ensuring the integrity of data from end-user to end-user makes it more difficult for data substitution attacks aimed at servers or hosts to succeed. Technical non-repudiation binds a user to a transaction in a fashion that provides important forensic evidence in the event of a later problem. Encryption protects private information from being divulged even over open networks.

Public key technology differs from systems using "shared secrets" or symmetric cryptography. In the latter, users are authenticated based on a password, Personal Identification Number (PIN), or other information shared between the user and the remote host or server, or between two or more users. A single key, again shared between two parties, provides communications privacy. The sender to encrypt and the recipient to decrypt transmissions use the shared key in an algorithm (agreed too beforehand by the transacting parties).

Symmetric cryptography has several inherent limitations that become acute when the transacting parties have no prior relationship. First, each pair of transacting party's needs a unique shared secret key – or else impersonation or eavesdropping becomes a problem. This means that the approach does not scale well – each user must have as many keys as people with whom he or she must deal. Second, once one party generates a secret key, that key must be transported securely to the trading partner, which can cause immense logistics problems and delays. Finally, because the individual must share the key with a trading partner, non-repudiation is lost. What this all means is that symmetric cryptography, by itself, is not conducive to e-commerce or e-government.

The limitations of symmetric cryptography are overcome using public key technology, which is also called "asymmetric cryptography." In a typical Public Key Infrastructure (PKI), two key-pairs are generated by or for each user, one key-pair for digital signatures and authentication, and the other key-pair for encryption. Each key-pair comprises two keys (very large numbers, typically 150 to 300 digits in length) which are mathematically linked in a very subtle way. For each key-pair, one key is kept private, and the other is made public.

Each public key is made public in the form of a digital certificate where a trusted party (called a Certification Authority, which may be within or external to the agency)

cryptographically binds the public key to the person's identity by digitally signing the certificate. The digital signature on the certificate ensures that any unauthorized alteration of either the identity or the public key will be detected.

> The mathematical algorithm used for generating the keys, and the size (length) of the keys, can be selected to provide virtually complete assurance that the private key cannot be deduced from the public one. In the case of a commonly used algorithm called "RSA," this can be done because information available at the time of key pair generation (where the private key *is* deduced from the public one using that information) typically is kept secure along with the private key

Because public key technology uses two keys, one of which is kept secret and the other made public, there is no "shared secret" between the transacting parties, and thus no opportunity for one party to compromise the interests of both by losing control over the "shared secret." There is also no need to manage large numbers of symmetric keys (since each set of transacting parties would need a unique symmetric key). The user makes the digital certificate available to whomever he or she wishes to conduct business with.

As long as the user keeps his or her private key private, a malefactor will have great difficulty attempting to impersonate the user or obtain private communications simply by attacking the remote computer or server – because there are no "shared secrets" used for these purposes. This is a critical point, because many attacks focus on large data bases of shared secrets – passwords, PINs, and the like – held at hosts or servers which, by their nature, must be available for access by multiple users and applications in order to provide the functionality for which they were designed. If the data base can be successfully compromised using dictionary or other attacks which rely upon finding one or a few commonly used passwords from a long list (even where the passwords are encrypted), a user's account or interests can be compromised without the user's knowledge and even if the user did nothing wrong. With public key technology, the user normally must do something wrong to be at risk: he or she must compromise the private key in some fashion.

In a common form of digital signature associated with e-mail, when the user wishes to sign a document digitally, he or she applies the private signing key to a hash of the document being signed which transforms the hash into a new, different value. The user then sends that signed hash along with the original document to the recipient. The hash is like a unique fingerprint of the document, expressed in the form of a large number. The recipient, in turn, takes the signed hash, applies the sender's public key which transforms the signed hash into the original unsigned hash, and then creates a fresh hash of the original document as sent. The two hashes must be identical for the digital signature to validate. The e-mail client software performs all of these functions – the user does not have to go through each step manually.

To describe an analogous situation using fingerprints, consider a case where the message sender wishes to send an emissary whom the recipient can trust. The sender takes the emissary's fingerprints (the "hash"), then seals the fingerprints in an envelope on which the sender signs his or her name manually so that it would be apparent if the envelope had been opened by anyone else (the envelope and content now constitute the "signed hash"). The emissary then carries the envelope and presents himself or herself to the recipient. The recipient takes the fingerprints of the emissary as he or she arrived; takes the envelope, verifies the written signature on it (converts the "signed hash" to the original hash), then opens the envelope and compares the fingerprints inside the envelope to those just taken from the emissary. If they are identical, the emissary is deemed to be the person sent by the sender. While this analogy is not perfect, it illustrates the concept in a human setting.

The action of digitally signing and then validating the signature to authenticate the sender provides data integrity for the document because any change to the document after the original hash is generated and signed would cause the signature to fail to validate. This affords *technical* non-repudiation – the user cannot later deny that his or her private signing key was used to make the digital signature. Of course, it is still necessary to demonstrate that the user had control of the private signing key to establish *legal* non-repudiation.

A sender can encrypt a document so that only the intended recipient can decrypt it. To do this, the sender generates a one-time symmetric encryption key (called a "session key") and uses that to encrypt the document. The sender then takes the public key of the recipient, encrypts the symmetric session key with that public key, and sends the encrypted session key plus the encrypted document to the recipient. The recipient, in turn, applies his or her private key to decrypt the symmetric session key, then uses that to decrypt the document. This combination of symmetric and asymmetric cryptography is done for reasons of computational efficiency, since the former can be done much faster on a computer than the latter. This is especially important for large files. Again, the e-mail software performs these functions automatically – the user does not have to go through each step manually.

Good security practice requires that the key-pair used for encryption should be different from the key-pair used for digital signatures. Why is this necessary? Because it is wise to have a copy of the private key used for decrypting information in the event the original copy is destroyed (otherwise there is no way to decrypt information encrypted using the corresponding public key). However, a copy should never be made of the private key used to make digital signatures. Thus, two key-pairs are needed.