Report: EDUCAUSE – NIH PKI Interoperability Pilot Project

Peter Alterman, Russel Weiser, Michael Gettes, Kenneth Stillson, Deborah Blanchard, James Fisher, Robert Brentrup, Eric Norman

Background

Under mandate to adopt broad electronic business methods by October, 2003, Federal Agencies are working hard to figure out ways to put their business on-line in a way that is secure. A leading contender to make e-government secure is and trustworthy public key cryptography. At the same time, far-sighted institutions of higher education have been busy deploying PKIs and issuing digital certificates to their faculties and staffs to enable secure, electronic business with the government and with each other. These institutions wish to use their locally issued digital credentials to do electronic business with the government securely. The NIH, in turn, wishes to be able to rely on business partner-issued digital credentials, thereby avoiding the cost and administrative burden of issuing and managing electronic credentials. NIH and EDUCAUSE jointly constructed a PKI interoperability pilot project that demonstrated the ability of the Federal Government to receive electronic forms signed with digital certificates issued by institutions of higher education.

Description of Project

In order to address this situation, NIH and EDUCAUSE conceived a research project that would demonstrate a simplified approach to submitting digitally signed electronic grant applications to NIH. Although the project used an electronic grant form, in reality any form could have been used; the point being that the project's approach is applicable to any electronic form or file. The explicit goals of the interoperability project were to:

- Receive grant applications as digital forms signed with two different, validated, digital certificates each (an NIH business process requirement);
- Use digital certificates issued by three (later changed to five) participating academic institutions;
- Demonstrate interoperability among different CA vendors' products, including PKI service providers.

A key consideration in the design was that NIH would be a relying party with respect to the digital credentials used to sign the electronic grant applications. This is important for several reasons. For privacy and resources reasons, NIH would like to avoid issuing digital credentials to individuals and institutions. Experience trying to maintain an up-to-date, accurate inventory of research faculty and staff has demonstrated to NIH the futility of a government-centric, centralized approach to issuing and maintaining credentials of faculty engaged in government-sponsored biomedical and biobehavioral research. On the other hand, academic institutions have a much easier time of keeping track of their faculty and graduate students – so long as they wish to continue to receive paychecks.

Many academic institutions are in the process of deploying PKIs and issuing digital certificates to faculty, staff and students to facilitate e-business on campus, and these schools have voiced a clear desire to use their locally issued digital credentials for doing business with the Federal government. Thus, the logical design plan was to encourage deployment of institutional PKIs.

To support the work of the project, NIH and EDUCAUSE contracted with Digital Signature Trust (DST) and Mitretek Systems to complete key portions of the work. Fundamental work resolving directory issues was done by Georgetown University.

NIH provided the participating institutions with a Microsoft Word Template version of the *PHS-398, Application for Research Grant* form, to be used as the model for this pilot. The form was made available for download at an NIH web site. (Although not selected by any participant, a PDF version of the PHS-398 was made available to all institutions for the pilot.) This was done to provide the institutions with an electronic document that could be manipulated locally (PI) common desktop software applications. Desktop signing of the Word templates was accomplished using Assured Office (now ProSigner) software, a Microsoft Office Suite

plug-in and standalone application developed by E-Lock (now Lexign). ProSigner, however, only works on the Microsoft Windows platform.

Phase One of the project incorporated the following assumptions and features:

- **A form that could be shared between the Principal Investigator and the Authorized Official of Record (AOR) at the research institution.** The PHS-398 is completed by PIs and the AORs, also known as Institutional Representatives (IR) in recognition of the fact that NIH funds institutions, not individuals. The form must allow for completion by multiple users, although only one of these users will submit the form to NIH.

- **A form that could be digitally signed with multiple digital signatures.** Both the PI and an IR sign the PHS 398. Both digital signatures need to be validated, that is, checked to verify they are good, when the form is submitted to NIH. The PI is typically part of a research operation of an organization. The institutional representative is an administrator, typically called the Authorized Official of Record (AOR) or IR. The two may be hundreds or thousands of miles apart. Bringing these people into a room at a single moment is often not feasible. Further, the AOR or IR may be handling numerous forms at a single time, related to many different investigators.

- **A form that could be completed with virtually no additional software requirements for the PI and IR/AOR.** In order to allow for maximum scalability, the team decided that the adopted solution should have as small a client footprint as possible, not only because of difficulties in downloading and installing products, but also because Information Technology (IT) departments are averse to installation of software that is not part of the standard configuration supported by the Institution's IT environment. This concern arises from added cost and support (which also translates to cost) requirements.

- **A form that could utilize commercial-off-the-shelf (COTS) digital signing products.** Based on our analysis of COTS digital signing software, the product that we recommended, E-Lock Web-Signer (now Lexign ProSigner), would sign not only portable document format (PDF) files, but also generally any other file type. Due to the number of users participating in this pilot, it was more cost effective to use the per-user-priced Assured Office (ProSigner) rather than the recommended Web-Signer, which is priced on a server basis.

  Research into the capabilities of Adobe Acrobat reader revealed that the reader software supported verification of signatures, but did not support digital signing or digital certificate validation natively. Additionally, Adobe Acrobat software, as distributed by the manufacturer, requires additional software plug-ins to be added to the desktop to allow it to function with PKI certificates that would be applicable to the project requirements. By using a COTS product that worked correctly with any file format, including Word templates, a separate plug-in for Adobe did not need to be created.

- **Form could be digitally signed and sent as an email attachment, requiring no changes to the NIH mail server.** In order to best meet the needs of the constituents of the pilot, e.g., the research institutions and NIH, the Word template needed to be completed, digitally signed, and emailed as an attachment to the NIH OER recipient. This allows for easier submission of the form, requiring no changes to the NIH email server or to current database or web servers. Furthermore, it greatly simplified the submission process for the institutions. The fact that their email systems logged the sending of the message as proof of date and time of submission was a serendipitous extra benefit.

PKI Bridges

To allow NIH to successfully validate the digital certificates affixed to the electronic grant applications, EDUCAUSE deployed a Higher Education Bridge Certification Authority (HEBCA) prototype structurally similar to the Federal Bridge Certification Authority (FBCA) prototype. With the support and approval of the Federal PKI Steering Committee, which included a generous grant, the two bridges were cross-to-certified and currently interoperate at the test level of assurance. Participating institutions' PKIs cross-certified with the Higher Ed Bridge while a proxy NIH CA cross-certified with the Federal Bridge. Thus, a trust path was created between NIH and the institutions

through the bridge-bridge infrastructure created to support the project.

Trust path discovery and validation for the bridge infrastructure model required use of specialized software. Mitretek Systems modified the Certificate Arbitration Module (CAM) originally created for the GSA Access Certificates for Electronic Services (ACES) program (an umbrella contract mechanism allowing the Government a acquire a broad range of PKI services) and added DAVE. The CAM/DAVE became the validation service used by Assured Office to validate the digital signatures affixed to the completed MS Word templates. How this worked will be explained further on in this paper.

Significant issues were encountered in attempting to link the different directories that supported the institutional PKIs. To resolve them successfully, the team found it necessary to use an Internet 2-supported "registry of directories," described below, developed by Michael Gettes of Georgetown University.

Interoperability

In addition to brokering trust among discrete PKIs, the Federal and Higher Education bridges also supported Certificate Authority (CA) product interoperability. The University of Alabama at Birmingham used the DST TrustID certificate service (RSA technology); the University of Wisconsin-Madison used the Netscape iPlanet CA and Dartmouth College used the Entrust CA. The University of California Office of the President and the University of Texas – Houston Health Science Center used the VeriSign On-Site CA service. (The latter has not yet been demonstrated to operate successfully in the pilot, but is expected to be operational shortly.)

By using interoperating bridges, the overall number of cross-certifications required within the community of interest was reduced. Policy mapping decisions were offloaded to the Bridge policy authorities. This model allowed disparate PKI communities to be "bridged" together. Its disadvantages were also evident: liability issues arose by offloading policy mapping functions to a Bridge policy authority; it was heavily dependent on a distributed directory system that was vulnerable to failure in a number of locations. Certificate path construction was complex, and there were disparities between the underlying directories, e.g., X.500 vs. LDAP. If proper certificate constraints were not used, then security issues were destined to erode the trust in the infrastructure. Depending on the policies of the Bridge Policy Authority, peer-to-peer cross-certification of CAs still could be required.

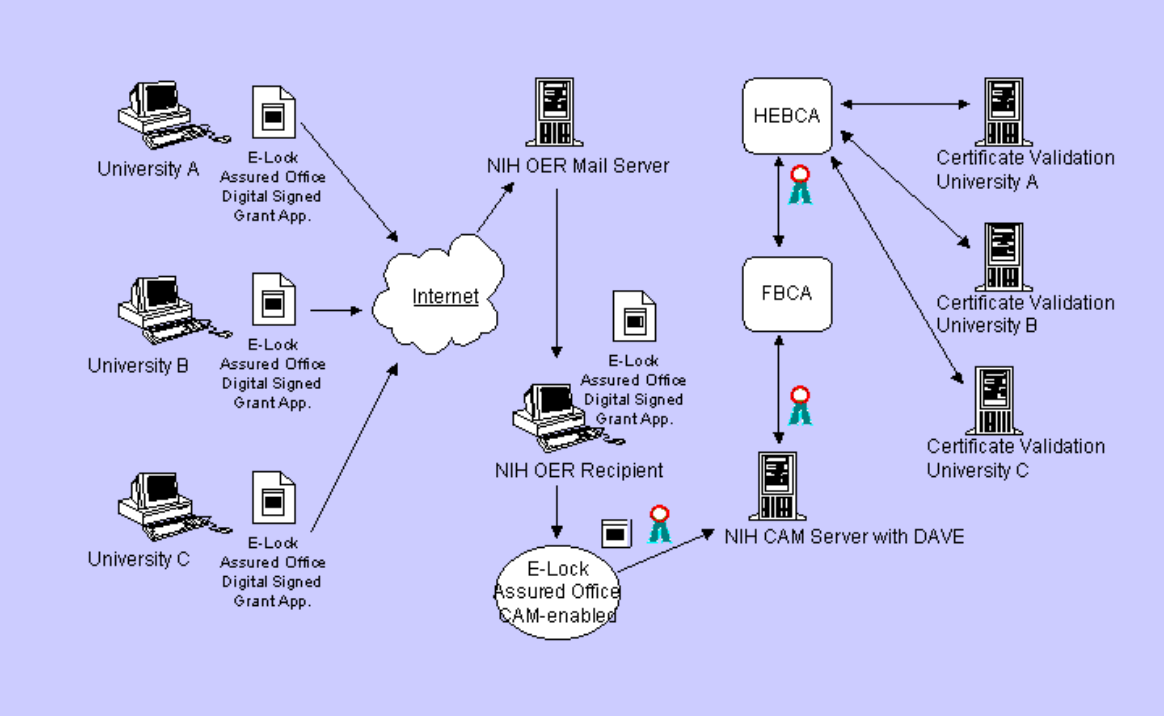
University CA Issues

As part of this project, university participants utilized their own CA software. The University of Wisconsin, for one, utilized the iPlanet CMS as its CA for university personnel certificates. This was one of the most challenging experiences – especially for the directory services. Their CA came integrated with the iPlanet LDAP directory in its default configuration, which assumed the CA would be used for an enterprise PKI in which users existed within the directory prior to obtaining the end entity certificate.

Because of this assumption, cross certifying with the HEBCA took some effort, specifically obtaining a PKCS#10 certificate request of the University of Wisconsin's root. This was found to be written as a file, instead of provided to the administrator. The publication of the cross certificate pair to the iPlanet directory had to be performed manually. The iPlanet software came with the *CertificationAuthority* object class and included *CrossCertificatePair* as one of the attributes. Using the *LDAPModify* command from the command line, the CrossCertificatePair could be published the directory

The Certificate Arbitration Module (CAM)

The CAM is an application-level router that efficiently and consistently routes certificates from relying party programs to the issuing certificate authorities (CAs) for validation. By interfacing directly with the CAM, a relying party application can interact seamlessly with multiple CAs. CAM is also flexible; it allows RSA-based certificates to be validated with the Certification Authority. The CAM runs as a separate process within the agency's security domain, allowing the agency to manage the resources and controls necessary to support the validation processing at the enterprise level. Applications interact with the CAM through a simple validation API that communicates over TCP/IP or by using a Microsoft ActiveX control.



Phase 2 of the NIH-EDUCAUSE Interoperability Pilot Project with FBCA and HEBCA

When a digital signature and the corresponding signer's certificate are presented to a PKI-aware application and the application does not recognize it, the application submits the certificate to the CAM. The CAM parses the certificate, verifies that it has not expired and checks to see that the certificate issuer trusted by the application. The CAM then either uses stored instructions or looks at the Authority Information Access (AIA) extension within the certificate to obtain the location of the OCSP validation service cited by the issuing CA. The CAM then builds an OCSP request, digitally signs it with a certificate issued to the CAM, and submits it to the OCSP server for validation.

When DAVE is incorporated, the issuing CA no longer needs to be known *a priori* (via configuration) and trusted by the CAM. Instead ,DAVE's trust anchor is known *a priori*, and DAVE performs the steps of trust path discovery and validation, the latter typically via Certificate Revocation Lists (CRLs).

The CAM *Validate Request* message contains three parameters: a message type, an Application ID string, and the DER-encoded certificate to validate. CAM then performs certificate validation on behalf of the application and returns a response message back to the application. The *Validation Response* message contains five parameters: message type, certificate status, an ACES profile check code (not used in this project), an ASCII representation of the parsed certificate, and the binary digitally-signed validation response message received by the CAM from the CA's validation service.

As the application-to-CAM communication utilizes TCP/IP, an Intranet (or Internet) connection must exist between the application and the CAM. The validation request response messages are transmitted in "Little Endian" byte order, so applications integrating with the CAM must take this into account and translate the messages if they are not running on a non-Intel platform. The NIH and many of the academic institutions used Intel platforms, so this was not an issue for them during the pilot project, but it was noted that a significant Macintosh users are part of the NIH client base.

The CAM receives the signed OCSP response from the issuing CA's Responder, verifies the signature, and parses the response to obtain the certificate status. The CAM logs the response (providing an audit trail) and packages the status along with additional information in the *Validation Response* message, as discussed above. While the

functionality of each CAM is limited to a single security domain, it is also ideal for a one-stop gateway or portal architecture.

Enabling applications to utilize the CAM for validation is a fairly straightforward task. Several key points must be taken into consideration, though (See CAM Communications Specifications - Version 3.1.0 at http://cam.mitretek.org/cam):

The original design requirements assumed that the CAM and the application are running in the same security domain, that is, the protocol between the application and the CAM itself were not currently authenticated;

- The CAM server runs on a Microsoft NT 4.0 or Windows 2000 platform;
- The CAM utilizes TCP/IP to transport the validation request, responses to and from the CAM;
- The CAM trust model, when not extended by DAVE, is that the CAM is authoritative; only certificates issued from a CA explicitly trusted by the CAM are validated, hence applications have no need for further validation.

CAM Implementation

To date, the CAM has been deployed successfully in a number of instances within the Federal Government. Although not in broad use today, this growth trend should continue over time. Examples:

1. The SSA is in the third year of its "Annual Wage Reporting" (AWR) pilot and the second year of utilizing the CAM as a signature validation service for electronic AWR filings. This year's pilot includes the use of a simplified signing control, "simple sign" to calculate the signature hash, sign the signature hash, and submit the filing to the SSA services. There, the signature is validated through the CAM validation server. Not only is SSA accepting signatures through the ACES program, it has added the State of Washington PKI as a trusted issuer within their CAM trust list;
2. FEMA utilizes the CAM validation service in several programs; first, to provide certificate-based access control to several critical databases available to emergency personal during disasters; second (deployed since the September 11th attacks), a government assistance program for local government agencies that are applying for FEMA assistance. This application allows electronic submission of grant applications as well as certificate-based access to check on the status of the application by the applicant;
3. NIST has developed an electronic grant application submission and review workflow to support its research grants program. This program utilizes both ACES and NIST-issued certificates and handles signature validation via the CAM;
4. NTIS has enabled its labor union wage reporting system, utilizing CAM for signature validation of union officials when union wage reports are filed with the NTIS servers. The reports are then accepted and the information fields verified and fed into the Agency's back-end workflow system;
5. The EPA ran a pilot, "CDX," that enabled digital signing of pollution reports by reporting agencies and businesses. The program has recently incorporated a full-blown reporting exchange that includes the digital signatures, submitted reports, and their validation at the point of acceptance.

Discovery And Validation Engine (DAVE)

DAVE is an open-source software package that provides X.509 certificate trust path discovery and validation services as a TCP/IP accessible Microsoft Windows NT/2000 service. DAVE may be used as an add-on to the CAM, extending CAM-enabled applications to hierarchical and cross-certified PKI domains.

Configuration settings for DAVE include:

- A certificate corresponding to the "trust anchor." All trust-paths end at this "most trusted CA;"
- An LDAP server name and port to use for retrieval of certificates and CRLs and/or ARLs.

The incoming request protocol used by DAVE is the same as that used by the CAM. Starting with CAM version 3.6a, the "CAM-linking" and "default CA" capabilities may be used to defer validation to DAVE for CAs not specifically listed on the CAM trust list. The outgoing request protocol for certificate path discovery and for CRL retrieval is LDAP, both for certificate path discovery and CRL retrieval. OCSP-based validation may be added at a later time. CAM already provides OCSP support, but only for directly trusted CAs, not ones located by path discovery.

DAVE applies multiple techniques to construct the certificate path. When the location of the issuer's certificate is given in the AIA field of the certificate in question, DAVE contacts that specified LDAP or X.500 directory directly. When explicit locations are not conveyed in the AIA field, or when a complete trust path has not yet been constructed, DAVE switches to a second technique, issuing LDAP "read" requests to its default LDAP server which, in turn, discovers and queries the correct directories. Such discovery is made by way of hierarchical CA certificates and cross-certificates. The explicit steps taken are: (1) read the issuer field from the certificate in question and call this the target domain name (DN), and (2) do an LDAP read for the target DN, asking for the return of both all *cACertificate* and *crossCertificatPair* attribute values.

This places two requirements on the directory infrastructure DAVE utilizes:

1. PKI objects (certificates, cross-certificates, and CRLs / ARLs) must be properly stored in a part of the Directory Information Tree (DIT) with a DN equal to the subject field of the object(s);
2. The LDAP server to which DAVE connects must know of and be able to retrieve any intermediate certificates or CRLs / ARLs along the constructible paths. This generally implies directory chaining agreements or an LDAP referral arrangement.

Internally, much of DAVE's functionality is provided by other open-source packages:

- The Certificate Management Library (CML) v2 provides path construction logic and certificate validation functions;
- Crypto++ provides cryptographic functions for signature verification;
- Netscape LDAP SDK DLL (in object form; no source available) provides referral-enabled LDAP client functions;
- S/MIME Freeware Library (SFL) provides MIME processing functions, and an abstraction for Crypto++;
- Certificate Arbitrator Module (CAM) code is taken from CAM for NT service abstraction and basic core library functions that provide thread safety, safe memory allocation, logging, etc.

DAVE Status

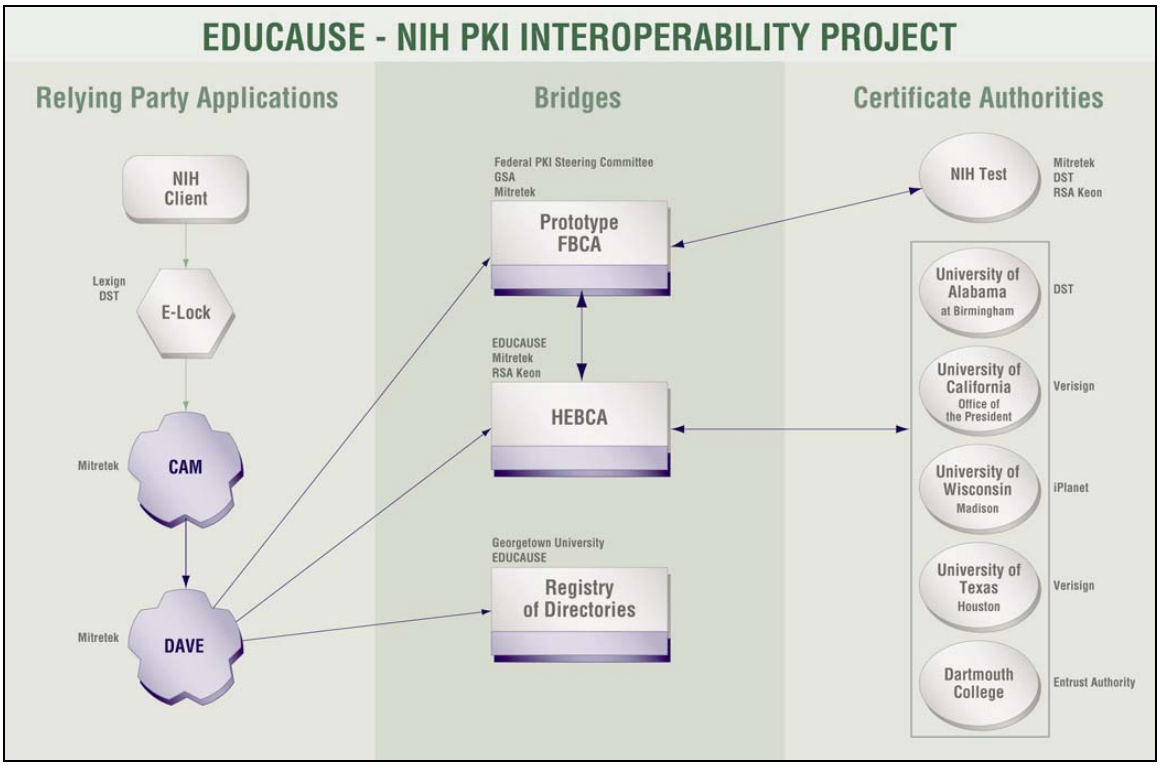
Initial development of DAVE is completed, and the source-code will be freely available shortly. DAVE has been tested in a number of trust topologies, with a variety of certificates issued by different CA product vendors.

Interoperability Pilot Test Environment

NIH is a participant in the Federal Bridge CA (FBCA) prototype and has a CA cross-certified with the FBCA prototype. The universities are participants in the Higher Education Bridge CA (HEBCA) prototype and their CAs are cross-certified with HEBCA. When a certificate is validated in this test environment, it demonstrates a trust path that traverses hierarchical and cross-certificate-based PKI domains, multiple bridges, multiple CA product vendors, and both LDAP networking mechanisms, directory chaining agreements for the FBCA, and an LDAP referral-based directory networking for the universities.

The pilot project test environment pictured above involved two users at three of the universities sending "dual signed" grant request forms using certificates issued by their respective CAs (DST/RSA, iPlanet, Entrust). The digitally signed forms were sent as attachments via standard e-mail to a user at the National Institutes of Health (NIH).

The NIH user received the e-mail message and used the CAM-enabled Lexign ProSigner application to validate the attached, signed form. ProSigner was configured to contact NIH's CAM, which contained a single-item trust list, deferring validation to DAVE. DAVE was configured with NIH's self-signed CA as it's trust anchor, and an LDAP meta-directory (referral-based) as its LDAP starting-point. On an initial run, this system was able to validate both signatures on the form within 20 seconds. On a second test run, when DAVE had automatically cached the certificates of the path, validation took place in under 5 seconds.



Pilot Project Description, highlighting positioning of CAM and DAVE in the trust discovery path

Directory Overview

Currently, the FBCA environment relies heavily on the use of X.500 directory standards to facilitate path discovery and path processing. This is partially due to the Federal Government's extensive experience with X.500 directories. Although the FBCA does utilized the LDAP v3 protocol as the primary protocol to the bridge directory, another X.500 based protocol is utilized to connect transparently to a distributed mesh of directories. Certificates that make up a full path may reside in external directories that are connected to the bridge directory transparently. The FBCA environment relies on the X.500 DSP protocol to chain automatically to the external distributed directories to retrieve the CA certificates, CRLs, and ARLs that are needed to perform path processing. The DSP protocol is managed through the use of 'Chaining Agreements' that manage authentication and retrieval of attributes and values that reside on these external directories. This environment has been tested in small scale by the FBCA with several directory and CA products.

Directory Issues

The FBCA model presents two fundamental challenges to the development of a HEBCA world. First, the FBCA was constructed under the assumption that X.500 directory services would be used for both the bridge and the agency directories, and the location for publishing certificates (including objects containing client, CA, CRL and ARL information) would be known *a priori*. Second, using directory request chaining to resolve requests for X.509 objects which the X.500 standard supports presents difficulties for LDAP implementations, since LDAP does not have a uniform mechanism for chaining requests and not all LDAP clients understand LDAP referrals. In the Higher Education computing environment, as in the marketplace, the use of X.500 directory servers is quite limited and LDAP is the predominant directory server technology employed for enterprise-class directory-enabled services. Since directory chaining is not one of the X.500 capabilities brought forward into the LDAP specification, the project team developed techniques for getting around these limitations.

Fundamental to the Federal BCA model is the notion that a request for an object associated with a *SubjectName* (Subject or Signer) is performed directly and not by issuing search requests. An application simply calls the "getDN" function and the directory infrastructure resolves the DN for the application.

It is also important to note that without an *AIA* extension in the certificate, the issues related to chaining and locating objects become significant. Very little software makes use of *AIA*, however, DAVE and CAM both use the *AIA* extension if it is present. If an HTTP URL form is present, DAVE will bypass directory lookups and use HTTP directly. If an LDAP URI form is presented to DAVE, the module directly queries the given LDAP server for the given DN; if it is a DN-only form, DAVE queries the default LDAP server using the DN from the *AIA* field, not the DN from the issuer/subject fields. The same logic applies for CDP fields when getting CRLs.

Chaining

This paper does not attempt to describe all aspects of chaining per the X.500 specification, but simply makes note of some of the reasons for choosing the X.500 chaining methodology and presents challenges for an LDAP equivalent methodology.

What typically transpires in the BCA model is that an application receives a form or document with an affixed certificate. To validate that certificate, the CRL associated with the issuer of the certificate must be queried to see if the received certificate is still valid. The application (or an associated certificate-handling module) extracts the *Issuer Subject Name* from the certificate and requests the DN that is the Issuer *SubjectName* from a locally-defined and -configured directory service. In the X.500 context, the DSA has the responsibility for performing any name mapping and for chasing down the DSA that houses the object associated with the DN. Since this involves accessing other directories, the authentication credentials are appropriately passed to other directories for proper access control to required information. This places the burden of translation and location on the DSA, and the application has to know little of the "magic behind the curtain." This "magic" is commonly referred to as "knowledge references" and there are various types to describe and implement different behaviors. One reference describes a chaining agreement between two DSAs. Another reference describes a referral, which is returned to the application to be handled as the application sees fit. From an application perspective, this is a reasonable mechanism.

In the LDAP world, however, chaining doesn't exist formally. It is relatively easy to implement a simplified version of chaining using LDAP, but there is no standard defined for the activity. In the pilot project, the application has to chase the DSA associated with an issuer DN. While applications usually call library functions, this model potentially increases the complexity for the applications, depending upon which LDAP libraries are used. In the case of the open-source *OpenLDAP* implementation, a derivative of the University of Michigan SLAPD implementation, the libraries handle referral chasing rather well. Nevertheless, for both referral and chaining, there is still work that must be done at the DSA to define knowledge references (and, of course, to test those references). Thus, in LDAP-based models, applications must know more about the process of certificate validation, calling library functions and performing the work, but this type of activity is commonplace for LDAP-enabled applications. If handled properly, the X.500 model and the LDAP model are equally transparent to the application.

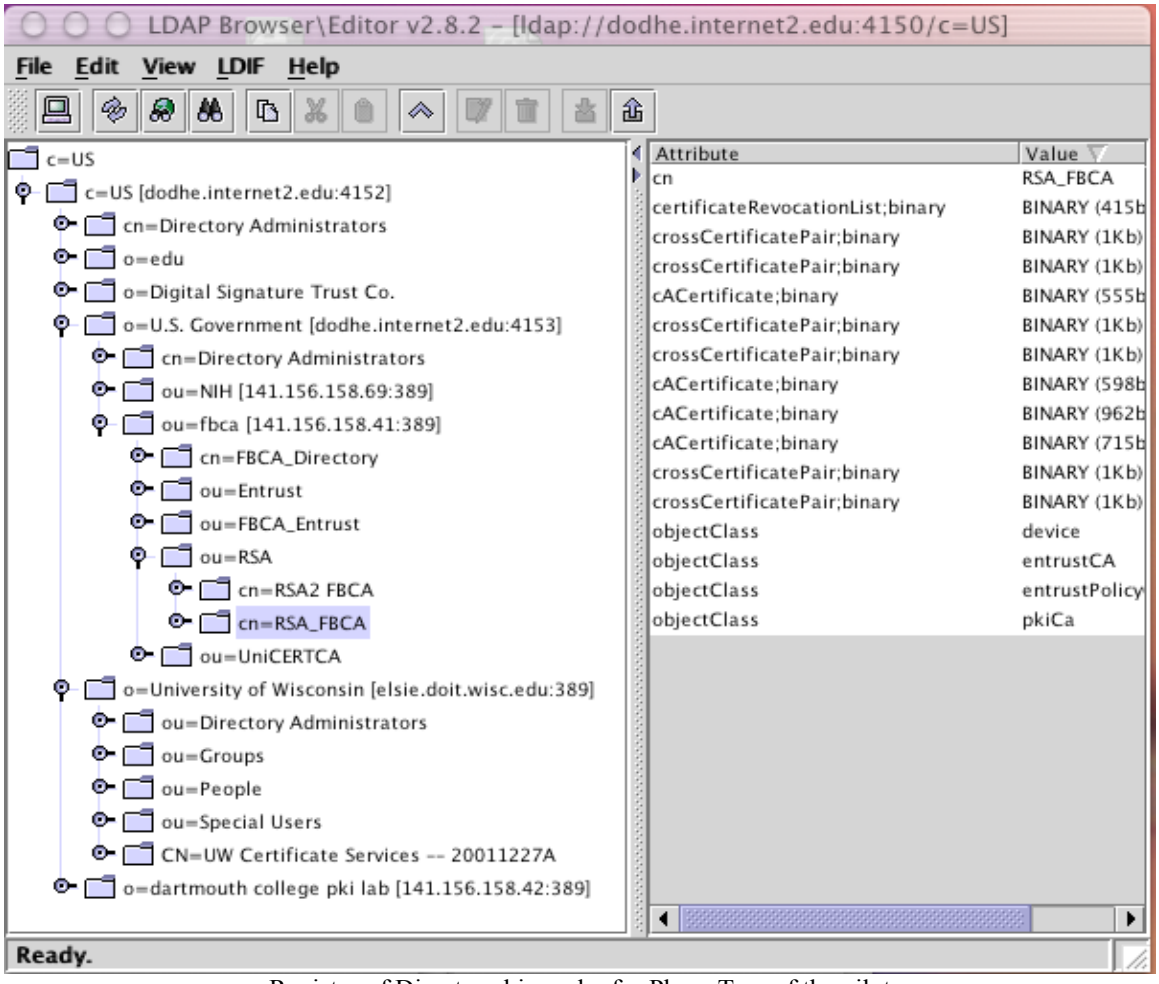
One important lesson from the FBCA work is that chaining agreements between different vendors of X.500 DSAs is quite problematic - to the point that a workaround was required for successful demonstration of the project proof of concept. Not every institution has the same Certificate Authority product or directory service product ,and if they do have the same products they might be different versions that are incompatible. This last situation particularly caused problems at the Dartmouth College PKI Lab, both with the CA and the directory (which had to be upgraded to the latest version, and even then had numerous directory chaining issues though it was an X.500 directory). Finally, the DSP protocol is time-dependent and hence two directories that are tied by chaining agreements require time synchronization in order to operate correctly.

Resolving Objects via LDAP: Registry of Directories

Given that LDAP has no inherent chaining capability, a knowledge reference service was developed that the LDAP-enabled, BCA-aware applications utilized. This service is a Registry of Directories (RoD). The RoD is an LDAP directory

utilized to provide "smart referrals" for CAs which are cross certified with the HEBCA, but which do not have X.500 directories that support the DSP chaining protocol. The RoD provides DN entries for the organization CA and an LDAP-based URI referral to the organization's directory, where the CA certificate, CRLs and ARLs actually reside. This allows DAVE to access the directory of the institution quickly and to retrieve the CA certificate, CRLs and ARLs in order to perform the path development and processing needed to bridge a trusted path with generic LDAP read and LDAP search operations. This is not much different from the FBCA concept, except that multiple directories are accessed via LDAP by the path processing software as opposed to being accessed by a single bridge directory, which then chains to the distributed directories of the participating CAs. The advantage of this is the simplicity of management of the RoD, as opposed to establishing separate chaining agreements across numerous distributed directories. This is particularly important given the sheer number of institutions, and the diversity of their infrastructures and needs.

The project created the RoD on a test system (dodhe.internet2.edu) using different ports to simulate a federated administration model of this registry. Our first implementation required the application be configured with the top of the registry service defined - or pointed to - any DSA associated with the RoD service. Each RoD DSA was configured with a superior reference, which implied that any DN requested that was not managed by the current DSA yielded a referral to the top of the RoD. The RoD figure below shows an expansion of the RoD hierarchy for this phase of the project. For each root, we configured a new RoD hierarchy. We defined two roots for this part of the project, one for c=US and one for dc=edu. Only the c=US branch is shown below, since this presents the FBCA test environment, as well as the HEBCA test environment.



Registry of Directory hierarchy for Phase Two of the pilot

Note the referrals shown in the above figure at:

c=US
o=U.S. Government
ou=NIH
ou=FBCA
o=University of Wisconsin
o-dartmouth college pki lab

The RSA_FBCA Certificate Authority was also selected in the above figure and shows the object contents to the right, revealing the CRL, *crossCertificatePair*, and *caCertificate* attributes which would be utilized in path validation and discovery. An application requesting the associated data with this object would, starting at the top, receive one referral for *c=US*, then another referral for *o=U.S. Government*, then one more referral for *ou=fbca*. The DN of this object is: *cn=RSA_FBCA, ou=fbca, o=U.S. Government, c=US.*
Referrals within the RoD service may exist at any level as appropriate for the administration of the namespace being referred. This offers flexibility to delegate administration out to the true owners of the namespace in the "global" DIT space.

Open Issues for the Registry of Directories

- Resource discovery seems to be a daunting and, as of yet, unsolved problem. Configuring client software (email clients, web servers and so on) with a local (or remote) DSA that is part of the RoD service is not desirable. Software should have a mechanism for locating the global service only if there is not a locally defined service. Using DNS SRV records and even poking at the DNS hierarchy within the local domain seem appropriate until an RoD Service SRV record is located. This will allow the starting point to be locally defined and will provide an escape route from the global hierarchy for special arrangements or alternative hierarchies depending on the commercial climate of namespace providers. DNS security is not an issue here since the objects being located will be digitally signed and will be, therefore, "self-secure" with respect to the certificate being validated.

- It is not clear which approach is better: getting an object or searching for an object. If certificates contain *AIA* extensions that lead directly to the object associated with the issuer, this is clearly the best approach. However, not all methodologies associated with *AIA* are understood by all software. If one has to locate the issuer object, then how is that accomplished? Do we search on the DN in question or simply get it? Currently, there is quite a bit of discussion within the IETF-PKIX community as to which approach is best, and even discussions regarding the representation of a certificate in a directory. Do we provide new attributes that represent the contents of certificates and search those attributes (since X.509 certificates are stored as binary blobs) or do we search using special filters and matching items which allow for searching inside the binary X.509 blobs? These questions are not yet resolved, but the FBCA model will likely have to incorporate some new set of techniques to work with new, PKI-aware applications developed in response to the results of the IETF deliberations.

- The referral URI used in the smart referrals of the RoD must be pre-escaped, meaning the URI definition rules must be adhered to such that space characters must be translated to the %20 in the URI.

- Utilizing the LDAP standard port definitions of 389 or 636 simplifies the setup, since the firewalls usually are already open for other LDAP services. The X.500 chaining agreement setup requires special ports to be opened, which can lead to time delays and further security concerns by IT staff.

- In the case of X.500 directory chaining, chaining agreements are required in both directories. This requires a coordinated effort and substantial amount of administrative time to initially setup, and test proper chaining. The LDAP referral method was found to be easily set up and tested without the need for tightly coordinated effort and without the number of schema restrictions of chaining.

- Directory availability and security are critical issues associated with the deployment of this type of PKI. There exists many issues and solutions to yield high levels of both availability and security. The Federal model advocates use of a "border directory" which is essentially a public view of data originating from internal directories or databases that likely reside behind a firewall. There are other issues associated with directory enabled applications which also require consideration which we will not attempt to discuss here. For more

information, refer to the Internet2 Middleware Initiative web site and the LDAP-Recipe at http://middleware.internet2.edu.

Border directories are specialized directories exposed to the world that contain a partial replica of proprietary information in the enterprise directory information tree of an institution or enterprise. This allows the border directory to supply public information to the bridge environment, thereby reducing the need for directory access controls and simplifying directory administration. The concept of the border directory is part of the FBCA architectural design to provide agency-based directories that expose only information needed for the FBCA to perform the path discovery and path processing. Institutions participating in the HEBCA will probably find this same concept to be a useful data security measure. Within the FBCA, the directories and border directories may be considered critical infrastructure systems and therefore require redundancy. This adds to the setup time and testing of the X.500 chaining agreements for both the bridge directory and the border directories. The HEBCA and the participating institutions could also be considered critical systems, but it is much easier to set up and test the smart referrals in the RoD than it is to ensure redundancy on all parts of the directory architecture.

- Firewalls and access controls to the directories within the institutions and the HEBCA will always need to be considered, although the referral mechanisms of the RoD simplify these issues because of LDAP's use of standard ports 389 or 636, as mentioned above.

- Anywhere that X.500 DSP is utilized, the administration of chaining agreements will require continuous checking, as well as synchronized time supplied, adding complexity to the infrastructure.

- Referral management will require institutional administrators to be aware of changes to the local directory tree that could affect RoD smart referrals. The LDAP Browser/Editor version 2.8.2 by Jarek Gawor was utilized for the creation of the smart referrals in the RoD as the native administration interface of the directory server was found to be cumbersome.

- Dartmouth College cross certified an Entrust Authority CA with the HEBCA. The Critical Path (previously PeerLogic) X.500 directory product was used with the Entrust Authority CA in this installation. The X.500 product needed to be upgraded to version 8A3 to resolve problems with directory chaining. The cross-certification exchange of certificates did not complete properly because of a still-unresolved incompatibility in the RSA product's response to the Entrust product. This issue was worked around by manually installing the cross certificates in the Dartmouth directory. A shadow DSA was created to avoid potential issues resulting from the manual certificate storage operation. Since additional hardware was not readily available to support the shadow DSA at Dartmouth, the team initially attempted to use a non-standard port for the shadow directory's LDAP connection. The Mitretek firewall, however, was only open for port 389 traffic. To work around this issue, the shadow directory was subsequently hosted on a server inside the Mitretek firewall. In addition, the update frequency for the CRL was extended to simplify synchronization with the shadow directory.

Desktop Service – Lexign ProSigner (E-Lock Assured Office)

ProSigner is a Public Key Enabled (PKE) application, allowing any PC-based documents to be digitally signed and encrypted. ProSigner is fully integrated with Microsoft Word, Microsoft Excel, and Adobe PDF enabling users to sign and encrypt documents quickly.

- Provides ease of use through a point-and-click tool bar that integrates with Microsoft Word, Excel, and Adobe Acrobat;
- Enables document encryption, so only specified people can view the content of a document;
- Provides centralized security including signing, encryption, verification, and certificate validation;
- Manages multiple signatures and creates an audit trail of documents as they flow through the signature cycle;
- Supports any X.509 digital certificate and works seamlessly with certificates issued by Digital Signature Trust, Entrust, RSA Security, VeriSign and others;
- Policy definition, enforcement and auditing insure simple workflow requirements.

Usage

ProSigner version 4.2 was utilized as a desktop service enabling the university partners to sign the Microsoft Word template PHS-398 form. To enable the signing, NIH translated its research grant workflow rules into Lexign signing policy that defines the two signatures be applied to the PHS-398 form.

The use of ProSigner, Microsoft Word and the PHS-398 allowed the researchers to fill out the electronic grant application form offline, wherever they were located. The researchers simply utilized Word to add the pertinent information into the PHS-398 document. Once all the information was completed, the researchers used the ProSigner controls in Microsoft Word to select their personal signing certificate to sign the application, then they attached it to an email to the institutional signing authority. The signing authority then reviewed the document, verified that it was signed by the researcher, and digitally signed it with his/her own signing certificate. Once both signatures were attached to the PHS-398, it was submitted to NIH simply by attaching it to an email and sending it to the OER email server.

The NIH recipient then opened the email and opened the attached PHS-398 with WORD and ProSigner. The NIH officer's ProSigner was configured to validate all certificates against a local CAM/DAVE validation service. When the PHS-398 was opened, signature validation was requested via the *Validate* API. If the certificate was within the trust list of the CAM, then standard ACES-level OCSP validation was performed. Since the certificates were issued from CAs not in the CAM trust list, validation was passed to DAVE and its configured default CA, the FBCA - HEBCA bridge infrastructure, to perform path discovery and path processing. When both certificates were verified through the CAM/DAVE service, the NIH officer then verified all the proper information was completed for the applications and disseminated it to referral and data entry.

As mentioned before, currently, ProSigner users must mange participating institutions' root certificates since the application still needs to see them in the Microsoft certificate store as trusted CA issuers in order to operate properly, even though it is CAM-aware.

Since Entrust software uses a proprietary client-side certificate store, it was necessary for Dartmouth's PKI Lab to use the Entrust-specific version of ProSigner to sign the sample NIH PHS-398 form with Entrust-generated certificates. Other pilot project participants used the Internet Explorer version. The now-current version of Entrust supports key/certificate export to the Microsoft Crypto-API, which should allow use of the IE version of ProSigner in the future. With these issues resolved, signatures and remote verification at NIH were successful.

Outstanding Desktop Application Issues To Be Resolved

The ProSigner version 4.2 utilized in the pilot project contained several problems that were worked around and should be fixed in later versions. Following is a brief list of these problems, followed by further explanation.

1. Explicit Trust in the CAM/DAVE validation without attempting to verify the CA within the local browser root store: ProSigner has been designed so that its certificate validation supported CRLs, OCSP, and CAM validation. In the case of CRL and OCSP based validations, the explicit validation of the CA required that the issuing CA root certificate was in the local browser root store and that the certificate being validated was valid within the validity period of the issuing CA's certificate.

2. The CAM response interpretation: Currently, the CAM validation API utilized by ProSigner returns several components to the *validate* API response message. Two of these parameters are important to the operation of the bridge-bridge model: the first is the CAM status code, which is the authoritative status of the certificate being validated and the second is the binary response message received by the CAM from the CA. Traditionally, this has been an OCSP response message from the issuing CAs validation service that may be used for long-term validation or archival proof of the certificate validation.

   The addition of DAVE means that an OCSP response message is not sufficient to contain the path information and its validation response to be stored with the document, allowing for the long-term interpretation of the document signatures. The addition of another signed binary response is an issue. Also, the signed

binary response from DAVE that encapsulates the path and validation information has not been standardized to provide a clear standard for developers to utilize. Although several IETF drafts provide options into which this information may be put, they are still subject to change. This is an area that needs further development. The first viable IETF Standard RFC to defined response information that includes path and validation information should be incorporated into DAVE. Of course, determining which IETF standard is viable can be problematic.

3. The CAM's application-to-CAM API has no security provisioning built into the *validate* API. This may be a limiting factor of the CAM's acceptance as a general validation service. An unexpected finding of the interoperability pilot project was the desire of researchers to use ProSigner and the CAM/DAVE validation service across institutional boundaries. This could allow a researcher to share critical research information securely utilizing ProSigner. The recipients then would need to verify the source of the signed documents electronically and would require that a public validation service such as CAM be deployed with new APIs providing security to the educational institutions.

4. Verification and Timestamp Issues. ProSigner stores audit information along with the signed document as signatures are verified. The timestamp of the verification is associated with the signature and with the document. However, if a document is signed and verified on 4/1/2002 at 12:01AM and then again on 4/15/2002 at11: 59PM, the timestamp is set to 4/15/2002 and not the original signing and validation date o 4/1/2002. Although not a direct issue for the pilot project, long-term audit information is highly important as proof of when a valid signature is applied to a document over time. It has been suggested that an initial validation timestamp and last validated timestamp should both be associated with digitally signed documents. This would facilitate creation of a minimal long-term archive of signed documents like the PHS-398.

5. When a document that has been signed and validated with the validation response stored with the document, then the document's signature hash is broken with a debugger, ProSigner does not report a invalid hash when using offline validation. This problem was reported as a defect to Lexign and should be fixed in the next release of ProSigner.

Policy Issues

The CAs that are part of the Interoperability Project issued certificates at the test level of assurance. To do business electronically, some form of policy needs to be created that addresses trust**.** Within the commercial X.509 PKI community, this is understood to require creation of a Certificate Policy (CP) in RFC2527 format that formulates the policies and procedures for issuing X.509 certificates at stated levels of assertion of identity and security. It also requires creation of a Certification Practices Statement (CPS) that describes in detail how the CA is to be operated to comply with the Certificate Policy. The degree to which a certificate user can trust the binding embodied in a certificate depends on several factors. These factors include the practices followed by the certification authority (CA) in authenticating the subject; the CA's operating policy, procedures, and security controls; the subject's obligations (for example, in protecting the private key); and the stated undertakings and legal obligations of the CA (for example, warranties and limitations on liability).

Beyond the strictly formal policy and procedures requirement, however, the organization issuing digital credentials needs to develop trust policies that address the questions implicit in establishing secure electronic business processes, for example: which credentials are good enough to satisfy trust requirements for a given transaction? What must be done to satisfy the business objectives, legal requirements, and culture of the organization issuing digital certificates?

Lessons Learned

Client Applications Client applications that rely on a Bridge CA have to know how to handle the certificate of each CA in the Bridge or to rely on the server-based certificate validation. Certificate repositories may not be accessible to the client applications. Client applications tend to not be able to handle complicated certificate hierarchies that may use cross certificates. Finally, client applications must be able to utilize the policy mappings of the different CAs in the bridge. This tends to be too much processing for client applications to handle.

Applications and Certificate Path Processing Server- based applications need to be able to handle the complexities involved to support certificate path processing and validation of the trust domains. In the implementation of the HEBCA, the CAM was enhanced to use an add-on discovery and validation engine (DAVE) module to facilitate certificate path processing and to validate the trust domains.

Trusted Servers Organizations are moving towards solutions that leverage trusted servers to do the hard work associated with certificate processing, rather than have the client do all the work. Hence solutions like CAM and OCSP or even plug-in modules such as DAVE are designed to perform discovery of a certificate path for processing to be used for validation.

Cross Certification In the Bridge approaches previously mentioned, cross certification can only be obtained with self-signed root certificates. Numerous commercial PKIs are designed such that subordinate CAs within the hierarchy are designated as the trust anchor for specific policies. This leads to the need to cross certify subordinate CAs with the bridge environment.

Directory Implementations In the Bridge approaches previously mentioned, X.500 directory and border directory implementations need to further embrace LDAP. As mentioned in the implementation of the HEBCA, a registry of directories and smart referrals were utilized to address interoperability across a diverse community of directory technologies.

Using a Bridge CA The cost for many agencies or institutions to operate and run their own PKI is more than these organizations can budget or afford. These organizations need to consider that ***in order to use a BCA, the agency or institution must have their own PKI***. An organization is oftentimes best served to utilize a trust model or PKI that is already in existence, such as ACES or a trusted third party (TTP).

Areas for improvement in the current application-to-CAM communications protocol: first, the lack of security within the protocol. Although not an original design requirement of the CAM, there are now use cases where the CAM and PKI implementation would benefit by the addition of authentication and confidentiality features to allow validation of the messages sent and received across the Internet. This would protect the transactions against denial of service (DOS) attacks and against replay attacks. Second, as noted above, the TCP/IP messages between the application and CAM utilize a nonstandard packet byte ordering, that is, Microsoft byte ordering instead of standard network byte ordering. Special attention should be paid to this when integrating applications to the CAM. The CAM source AA_TEST application, which is used for initial testing of a CAM installation, is a good starting point for integrators implementing the *validate* API.

Continuing Work

As more agencies and organizations adopt and participate in the BCA approach, more work needs to be done to ensure their success. Some of the immediate needs are identified below.

- Create a cookbook or document that identifies the minimal requirements and contents of the cross certificates and the directories; Given the lessons learned and discoveries made for all the components, a cookbook or document needs to be formally written that identifies the minimal requirements for certificates, directories and applications.
- Complete the cross-certification of Dartmouth by resolving the incompatibility with the RSA Keon CA product and Entrust;
- Continue to work with Verisign to complete the cross-certification of the University of California-Office of the President and University of Texas-Houston Health Science Center;
- Split the registry of directories to enhance performance across the infrastructure;
- Analyze and determine a more general solution for DAVE to perform directory discovery. It may be advantageous for DAVE to speak OCSP, for example;
- An investigation into multiple smart referrals to provide two different URIs to verify the enablement of redundancy for critical infrastructure cases. This would include teaching DAVE to try a secondary URI if the first did not return a response. If the AIA extension were mandated for any CA that wants to operate in a bridge environment, that would be a good beginning. Then, require an RoD entry for all participants of a bridge environment so the software would look at the AIA extension or the RoD to locate the issuer.

Summary/Conclusions

Given the disparate and many PKIs that are in use within the Federal Government and within other communities of interest, research institutions and Federal Government need to begin understanding how they can best leverage and work with the PKI environments that are underway. We need to come to an understanding and agreement that there will never be a single open PKI for everything. Rather, each major industry will determine its own solution, and the other industries that have a requirement to interoperate with other industries will need to figure out how to interoperate. An example is in the credit card world. The Federal Government did not define its own credit card standard. Rather, it evolved its payment processes to include the use American Express (AMEX) cards by Federal employees. The same is true for PKI. As an example, the higher education community will define its solution, and if the higher education community and the Federal Government want to interoperate, these two diverse communities will need to determine the best method of interoperability or continue to participate in the development of the infrastructure for each community.

Acknowledgements

Grateful appreciation for their participation in the pilot project and in the preparation of this manuscript is acknowledged to: Clair Goldsmith, University of Alabama at Birmingham; Jill Gemmill, University of Alabama at Birmingham; Keith Hazelton, University of Wisconsin-Madison; Eric Norman, University of Wisconsin-Madison
Robert Brentrup, Dartmouth College; Ed Feustel, Dartmouth College; Michael Gettes, Georgetown University; David Wasley, University of California Office of the President; Bill Weems, University of Texas – Houston Health Science Center; Mark Luker, EDUCAUSE; Steve Worona, EDUCAUSE; Deb Blanchard, Digital Signature Trust; Monette Respress, Mitretek Systems; Jim Fisher, Mitretek Systems; Ken Stillson, Mitretek Systems; Russ Weiser, Digital Signature Trust; Jack Kirivong, Lexign; Andrew Lehfeldt, RSA Security; Andrew Lins, Mitretek Systems; Cheryl Jenkins, Federal Bridge Certification Authority; Judy Spencer, Chair, Federal PKI Steering Committee.

References

LDAP-Recipe: A Recipe for Configuring and Operating LDAP Directories, Michael R Gettes, Georgetown University, February 2001 & April 2002

Bridge Validation Authority, Ambarish Malpani, ValiCert, Inc., December 2001

Planning for PKI, Best Practices Guide for Deploying Public Key Infrastructure, Russ Housley, Tim Polk, John Wiley and Sons, Inc., 2001

Federal Grant Streamlining Program, Department of Health and Human Services Response to RFI-4-02-HHS-OS, Digital Signature Trust, February 2002

Final Report – Phase 1, Prepared for National Institutes of Health (NIH) Office of Extramural Research (OER), Under Contract No. GS00T99ALD0006, Digital Signature Trust, February 2002

Report of Federal Bridge Certification Authority Initiative and Demonstration Electronic Messaging Association Challenge 2000, October 2000, Mitretek Systems

PKI, Implementing and Managing E-Security, Nash, Duane, Joseph, and Brink, RSA Press, McGraw Hill, 2001

Educause Review , "A "Bridge" for Trusted Electronic Commerce", Mark A. Luker, January/February, 2002, Volume 37, Number 1

The Evolving Federal Public Key Infrastructure, Federal Public Key Infrastructure Steering Committee and Federal Chief Information Officers Council, June, 2000

Internet2 Middleware Initiative Web Site, http://middleware.internet2.edu, Middleware Architecture Committee for Education (MACE), et. al.