Bridge Certification Authorities: Connecting B2B Public Key Infrastructures

William T. Polk and Nelson E. Hastings
National Institute of Standards and Technology

Businesses are deploying Public Key Infrastructures (PKIs) to support internal business processes, implement virtual private networks, and secure corporate assets. In addition, most businesses have industrial partnerships with other businesses for economic reasons. If these industrial alliances wish to exploit their internal security capabilities for business-to-business (B2B) electronic commerce applications, connection of their corporate PKIs will be required. However, corporate PKIs may implement different architectures, security policies, and cryptographic suites. A flexible mechanism is needed to link these corporate PKIs and translate these corporate relationships into the electronic world.

The Bridge Certification Authority (BCA) provides the means to leverage the capabilities of existing corporate PKIs. The following article describes different PKI architectures, difficulties in connecting the architectures, and how the BCA addresses these issues. In addition, the article describes the BCA concept, BCA deployment in the U.S. federal government, and how the BCA enables B2B electronic commerce.

Keywords: Public Key Infrastructure, Cryptography, Certification Authority, Bridge Certificate Authority, B2B, and Electronic Commerce

Author Contact Information:

Nelson E. Hastings
100 Bureau Drive – STOP 8930
Gaithersburg, MD 20899-8930
Phone: (301) 975-4634
Fax: (301) 948-1233
E-mail: nelson.hastings@nist.gov

W. Timothy Polk
100 Bureau Drive – STOP 8930
Gaithersburg, MD 20899-8930
Phone: (301) 975-3348
Fax: (301) 948-1233
E-mail: tim.polk@nist.gov

Bridge Certification Authorities: Connecting B2B Public Key Infrastructures

William T. Polk and Nelson E. Hastings
National Institute of Standards and Technology

Businesses are deploying Public Key Infrastructures (PKIs) to support internal business processes, implement virtual private networks, and secure corporate assets. In addition, most businesses have industrial partnerships with other businesses for economic reasons. If these industrial alliances wish to exploit their electronic commerce capability for business-to-business (B2B) applications, connection of their corporate PKIs will be required. However, corporate PKIs may implement different architectures, security policies, and cryptographic suites. A flexible mechanism is needed to link these corporate PKIs and translate the corporate relationship into the electronic world.

The Bridge Certification Authority (BCA) provides the means to leverage the capabilities of existing corporate PKIs. The following article describes different PKI architectures, difficulties in connecting the architectures, and how the BCA addresses these issues. In addition, the article describes the BCA concept, BCA deployment in the U.S. federal government, and how the BCA enables B2B electronic commerce.

Attributes of Traditional Public Key Infrastructure Architectures

A Public Key Infrastructure (PKI) is the key management environment for public key information of a public key cryptographic system. A public key cryptographic system is a cryptographic system where two mathematically related keys are used to encipher and decipher information. In a public key cryptographic system, one key is used to encipher or decipher the information and the other key is used to perform the reverse operation. One of the keys must be kept secret and is known as a private key, while the other key may be distributed to anyone and is called the public key. Within a PKI, a data structure called a certificate is used to bind a specific identity to a specific public key and information on how the public key can be used. The most widely used certificate specification is found in the International Telecommunications Union X.509 standard. Certification Authorities (CAs) are trusted entities that issue certificates to users within a PKI and provide status information about the certificates the CA has issued.

Traditionally, PKI architectures fall into one of three configurations: a single CA, a hierarchy of CAs, or a mesh of CAs. Each of the configurations is determined by fundamental attributes of the PKI: the number of CAs in the PKI, where users of the PKI place their trust (known as a user's trust point), and the trust relationships between CAs within a multi-CA PKI.

The most basic PKI architecture is one that contains a single CA that provides the PKI services (certificates, certificate status information, etc.) for all the users of the PKI. All the users of the PKI place their trust in the only CA of the architecture. Every certification path begins with the CA's public key. This results in a single user trust point. (see sidebar "When Should Alice Trust A Certificate?") This configuration is the simplest to deploy; only one CA must be established and all the users understand the

applications for which the certificates were issued. However, this configuration does not scale easily to support very large or diverse communities of users. The larger the community of users, the more difficult it becomes to support all of the necessary applications. A natural extension is to connect CAs that support different communities to create a larger, more diverse PKI.

Isolated CAs can be combined to form larger PKIs in two basic ways: using superior-subordinate relationships, or peer-to-peer relationships. In theory, any organizational structure can be realized using either of the two methods. In practice, however, there are technical and political issues encountered when architecting organizational PKIs. Each method has its strengths and weaknesses, and organizations can select the method that meets their needs. For larger, more complex organizations, these methods may need to be combined to develop the optimal PKI.

To illustrate this point, consider Figure 1. Alice and Bob are the community of users associated with CA-1; Carol is the community of users associated with CA-2, and David is the community of users associated with CA-3. These communities cannot interact in a trusted manner, as their CAs do not have trust relationships. If all the users are to leverage their existing certificates to obtain secure services, then trust relationships must be established between the three CAs to form a larger PKI. Should the trust relationships between the CAs be superior-subordinate or peer-to-peer relationships? The answer to that question will depend on a number of factors.

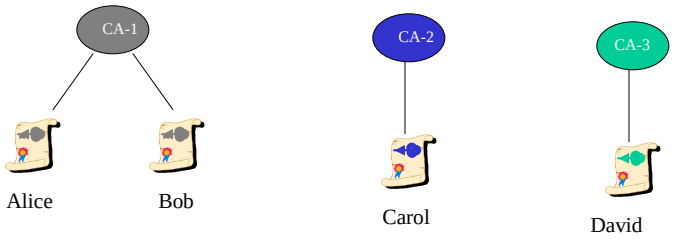


Figure 1- Three Separate PKI User Communities

A PKI constructed with superior-subordinate CA relationships is called a hierarchical PKI. In this configuration, all users trust the same “root” CA. That is, all users of a

hierarchical PKI begin certification paths with the “root” CA's public key. In general, the “root” CA does not issue certificates to users but only issues certificates to subordinate CAs. Each subordinate CA may issue certificates to users or another level of subordinate CAs. In a hierarchical PKI, the trust relationship is only specified in one direction. That is, subordinate CAs do not issue certificates to their superior CA. Figure 2 shows two examples of hierarchical PKIs where the “root” CAs are shown in red. The superior CA imposes conditions governing the types of certificates subordinate CAs can issue. These conditions are well known to users of the PKI, so they need not be specified in the certificate contents.

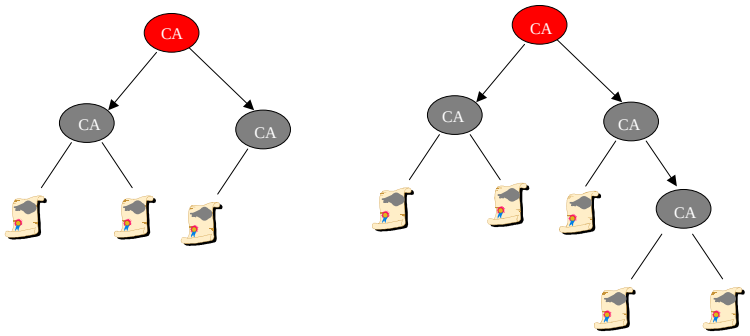
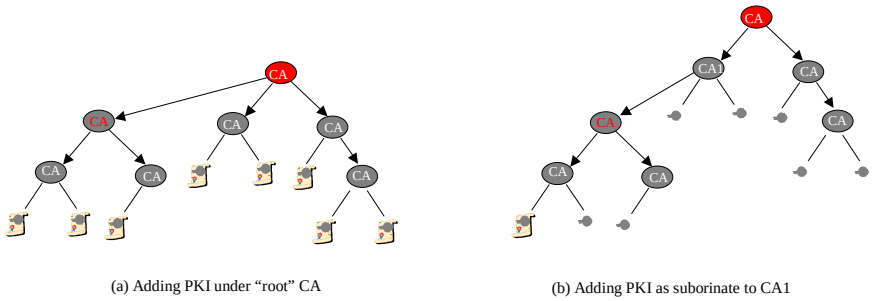


Figure 2 - Sample Hierarchical PKI Architectures

Hierarchical PKIs have four attractive properties due to the simple structure and unidirectional trust relationships. The first property is that hierarchical PKIs are scalable; to incorporate a new community of users, the “root” CA establishes a trust relationship with that community’s CA. Figure 3 shows two ways to create a new hierarchical PKI from the two PKIs shown in Figure 2. In figure 3(a), the “root” CA of PKI 1 was grafted directly under the “root” CA of PKI 2 and is now a subordinate CA within PKI 2. In figure 3(b) the “root” CA of PKI 1 has become a subordinate CA to one of the subordinate CAs within PKI 2. The actual position of a CA within the hierarchy would be determined by the political realities of the organization. The second property is that certification paths are easy to develop because they are unidirectional. This results in a simple, obvious, and deterministic path from a user’s certificate back to the trust point. The third property is that certification paths are relatively short. The longest path is equal to the depth of the tree plus one: a CA certificate for each subordinate CA plus the user's certificate. The fourth property is that users of a hierarchy know implicitly which applications a certificate may be used for, based on the position of the CA within the

hierarchy. As a result, certificates used in a hierarchy may be smaller and simpler than those used in a mesh PKI.



(a) Adding PKI under "root" CA

(b) Adding PKI as suborinate to CA1

Figure 3 - Connecting Two Hierarchical PKIs Together to Form a New Hierarchy

Hierarchical PKIs have drawbacks as well. These drawbacks result from the reliance on a single trust point. The compromise of a "root" CA, everyone's trust point, results in a compromise of the entire PKI. Worse yet, there are no straightforward recovery techniques. The nature of a hierarchical PKI is that all trust is concentrated in the "root" CA and failure of that trust point is catastrophic. Another drawback is that agreement on a single "root" CA may be politically impossible. Turf wars and inter-organizational competition may preclude such an agreement. Finally, transition from a set of isolated CAs to a hierarchical PKI may be logistically impractical because all users must adjust their trust points.

The traditional alternative to a hierarchical PKI is to connect CAs with a peer-to-peer relationship. A PKI constructed of peer-to-peer CA relationships is called a mesh PKI or a "web of trust" as shown in Figure 4. All the CAs in a mesh PKI can be trust points – in general, users will trust the CA that issued their certificate. CAs issue certificates to each other; the pair of certificates describes their bi-directional trust relationship. Since the CAs have peer-to-peer relationships, they cannot impose conditions governing the types of certificates other CAs can issue. However, the trust relationship may not be unconditional. If a CA wishes to limit the trust, it must specify these limitations in the certificates issued to its peers.

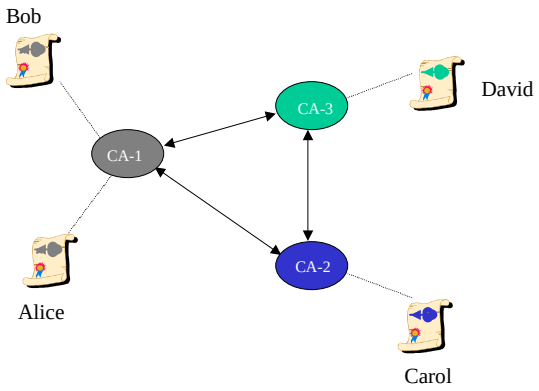


Figure 4 - Sample Mesh PKI Architecture

Mesh PKIs have several attractive properties. Mesh PKIs can easily incorporate a new community of users; any one of the CAs in the mesh simply establishes a trust relationship with that community's CA. Mesh PKIs are very resilient since there are multiple trust points. Compromise of a single CA cannot bring down the entire PKI. CAs that have issued certificates to the compromised CA simply revoke them, removing the compromised CA from the PKI. Users associated with other CAs will still have a valid trust point, and can communicate securely with the remaining users in their PKI.[1] Recovery from a compromise is simpler in a mesh CA than in a hierarchical PKI, if only because it affects fewer users. A mesh PKI can easily be constructed from a set of isolated CAs because the users do not need to change their trust point (or anything else). All that is required is that the CAs issue certificates to at least one CA within the mesh. This is very desirable when an organization wants to merge separately developed PKIs.

However, mesh PKIs have some undesirable properties as well due to the bi-directional trust model. Certification path development is more complex than in a hierarchy. Unlike a hierarchy, building a certification path from a user's certificate to a trust point is non-deterministic. This makes path discovery more difficult since there are multiple choices. Some of these choices lead to a valid path while others result in useless dead-ends. Even worse, it is possible in a mesh PKI to construct an endless loop of certificates. In a mesh PKI, scalability is a curse as well as a blessing. The maximum length of a certification path in a mesh PKI is the number of CAs in the PKI! Users of a mesh must also determine which applications a certificate may be used for based on the contents of the certificates rather than the CA's location in the PKI. This determination must be

[1] In the best case, the PKI shrinks by a single CA and its user community. At worst, the PKI fragments into several smaller PKIs..

performed for every certificate in a certification path. This requires larger and more complex certificates and more complicated certificate path processing.

Earlier, we posed the question: "Should the trust relationships between the CAs be superior-subordinate or peer-to-peer relationships?" The answer is, either could work. It depends upon the relationships between the user communities. Now that we have examined the strengths and weaknesses of the two architectures, we can consider when each model is appropriate. We will consider two possible scenarios:
(1) The user communities represent different organizations within a single company.
(2) The user communities represent distinct companies. These companies have contractual relationships (e.g., they are trading partners), but are separately owned and managed.

Figure 5 shows a PKI hierarchy consisting of a “root” CA and three subordinate CAs. Creation of the hierarchy requires that each user from the different user communities adjust his trust point to the newly established “root” CA. This represents a fundamental change in the trust relationships of the PKI, since users had no previous contact with the new “root” CA. In scenario (1), this fundamental “re-organization” of the PKI could work. The user communities have relationships defined through an existing organizational structure. Changing their trust point is simply an acknowledgement of their position in the current corporate structure. In scenario (2), this re-organization is doomed to fail. The relationships between the user communities are not based upon superior-subordinate relationships. Since the user communities are separate companies, there is no clear central authority. In the absence of this central authority, the communities may not be able to agree on a mutually acceptable third party to establish the “root” CA of the new PKI hierarchy. Changing their trust point would be in conflict with the established relationships between these communities.

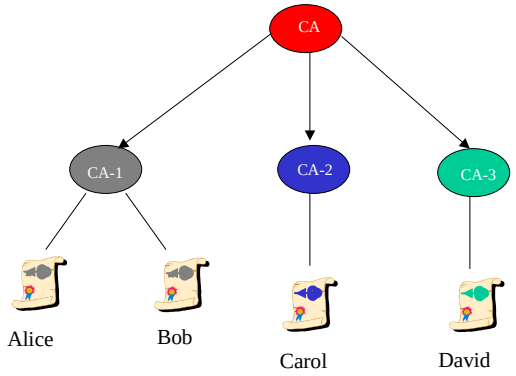


Figure 5 - Connecting PKIs Into A Hierarchical PKI

Figure 4 shows the mesh PKI alternative, where the CAs created bi-directional trust relationships by issuing certificates to each other. Users of the different communities do not share a single trust point. Each user still trusts the CA that issued his certificate. In scenario (2), this solution seems natural and would be easy to implement. This architecture does not impose a central authority where none naturally exists. The user communities are related by bilateral agreements (e.g., contracts and partnerships), which are represented accurately through the CAs' peer-to-peer relationships. In scenario (1), this solution would be very difficult to implement. Superior-subordinate relationships are clearly defined in most corporate structures, but the mesh does not reflect these relationships. Peer relationships may be more controversial; two parts of an organization may report to the same entity, but they are not necessarily peers. Finally, the central authority for the company is not recognized by the architecture.

Bridge Certification Authority Architecture

The Bridge Certification Authority (BCA) architecture was designed to address the shortcomings of the two basic PKI architectures, and to link PKIs that implement different architectures. Unlike a mesh PKI CA, the BCA does not issue certificates directly to users. In addition, the BCA is not intended to be used as a trust point by the users of the PKI, unlike the “root” CA in a hierarchy. The BCA establishes peer-to-peer trust relationships with the different user communities, which elevates political issues between organizations and allows the users to keep their natural trust points. These relationships are combined to form a “bridge of trust” enabling users from the different user communities to interact with each other through the BCA with a specified level of trust.

If a user community implements a trust domain in the form of a hierarchical PKI, the BCA will establish a relationship with the PKI’s “root” CA. However, if the user community implements a trust domain by creating a mesh PKI, the BCA need only establish a relationship with one of the PKI’s CAs. In either case, the CA of the PKI that enters into a trust relationship with the BCA is termed a principal CA.

Figure 6 shows a BCA that has established trust relationships with the user communities of a mesh PKI and a hierarchical PKI. Alice received her certificate from a CA in the hierarchical PKI, and uses the “root” CA of the hierarchy as her trust point. Harry received his certificate from a CA of the mesh PKI and uses that CA as his trust point. Harry and Alice can use the “bridge of trust” that exists through the BCA and their respective trust points to establish a relationship that enables them to interact with each other in a trusted manner.

A PKI created with a BCA is sometimes called a "hub-and-spoke" PKI. The BCA links many PKIs at a single, known hub. In comparison to a mesh PKI, certification path discovery becomes easier in a BCA-architected PKI. The user typically knows their path to the BCA; they need only determine the path from the BCA to the user's certificate. In addition, a BCA-architected PKI will typically have shorter trust paths than a random mesh PKI with the same number of CAs. Certification path discovery is still more

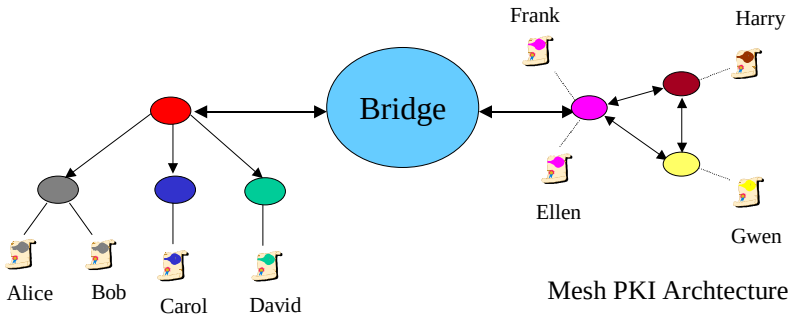


Figure 6 - Bridge CA Connecting Different PKI Architectures

difficult than in a hierarchy, and the typical path length is approximately doubled. However, the decentralized nature of a BCA-architected PKI more accurately represents the real world of organizational relationships.

A BCA Implementation: The Federal Bridge Certificate Authority

An example of a PKI based on the BCA concept is the U.S. Government's Federal PKI. The initial designs for a federal PKI were hierarchical in nature because of government's inherent hierarchical organizational structure. However, these initial PKI plans ran into several obstacles. There was no clear organization within the government that could be identified and agreed upon to run a governmental "root" CA. While the search for an appropriate organization dragged on, federal agencies began to deploy autonomous PKIs to enable their electronic processes. The search for a "root" CA for a hierarchical federal PKI was abandoned, due to the difficulties of imposing a hierarchy after the fact. Instead, plans were developed to integrate the agency PKIs into a unified federal mesh PKI.

The number and complexity of trust relationships required for the new federal mesh PKI was daunting. Government agencies began to search for a solution that would represent their trust relationships in a more manageable fashion. The concept of the BCA was developed and was chosen as the vehicle for a unified federal PKI. A prototype Federal BCA (FBCA) has been implemented and demonstrated. The prototype FBCA links five trust domains representing three federal departments, one state government, and one foreign government. The five trust domains include both hierarchical and mesh PKIs. In the FBCA demonstration, users from different trust domains were able to communicate with each other in a trusted fashion using secure electronic mail.

BCA Economics 101: How the BCA enables B2B E-commerce

Telecommuting, electronic mail, and web-based document delivery are commonplace in today's business operations. These applications help businesses streamline their processes, but they can also place sensitive information and assets at risk. To mitigate these threats, many organizations are implementing PKIs to secure internal operations. By deploying PKIs, organizations can protect their assets and still achieve the cost and time savings of electronic processing.

Most organizations have business partners and a desire to extend their electronic processing capability beyond their organizational boundaries. Businesses wish to leverage existing PKIs to support electronic processing with their business partners. However, these economic alliances are inherently dynamic. An organization may purchase widgets from one supplier today; if a new supplier can offer decreased cost or increased quality, they may select a new supplier tomorrow. Establishing or terminating peer-to-peer relationships each time business partners change is impractical given the dynamic nature of today's business relationships. In addition, the relationships between partnering organizations do not lend themselves naturally to support a hierarchical PKI.

The BCA architecture is ideally suited to support business-to-business (B2B) relationships. While each company has a limited set of business partners at any given moment, this set is very fluid. Companies establish and terminate these relationships with astonishing speed. However, the companies within a particular industry are not so dynamic. To illustrate this point, consider companies that manufacture sailboats.

Sailboat manufacturers include either a gasoline or diesel engine in their boats. They don't build engines; they buy them from marine engine suppliers and install them in the boat. Sailboat manufacturers may change engine suppliers to obtain more powerful, lighter or less expensive engines as new products emerge. However, there is a limited set of sailboat manufacturers and a limited set of marine engine manufacturers. While we cannot predict which sailboat and engine manufacturers will partner in the future, we can identify the two pools of candidates – the set of engine suppliers and the set of sailboat manufacturers.

In general, sailboat manufacturers are competitors, but they all buy the same kinds of products from the same set of vendors. They need fiberglass cloth and epoxy resin for hulls, wood for the interior, aluminum tubing for spars and masts, and engines for calm days. Sailboat manufacturers compete on design, quality, and appearance as well as price. Where opportunities exist to increase efficiency and reduce prices across the board, they have an incentive to work cooperatively.

Electronic commerce with consumers has limited appeal for sailboat manufacturers because sailboat buyers want to see and touch a sailboat for themselves before they make a purchase. However, B2B electronic commerce between sailboat manufacturers and their suppliers has significant promise to increase construction efficiency and reduce the overall cost of a sailboat. A sailboat builder does not want to maintain a large inventory of expensive marine engines. On the other hand, they lose money if they stop production

while they wait for a new engine to be delivered. Electronic commerce could help the manufacturer maintain just the right level of inventory by placing and filling orders efficiently.

However, a sailboat manufacturer cannot always predict which engines will be used in their boats. Building a PKI that links a sailboat manufacturer with the engine supplier they use today is useful, but the sailboat manufacturer may have to repeat the process next year if they decide to change suppliers. To complicate the situation, if an engine supplier's workers go on strike, a sailboat manufacturer may have to change suppliers even sooner. If a sailboat manufacturer does not establish a PKI with an engine supplier, the advantages of B2B electronic commerce are lost.

This type of situation is the motivating factor for establishing BCAs for specific economic industries. The sailboat industry (the manufacturers and their suppliers) would benefit as a whole from electronic commerce by deploying a BCA. By pooling resources, the sailboat industry could establish a BCA that would link all the sailboat builders and the parts manufacturers. The BCA would establish peer-to-peer relationships, eliminating arguments about sailboat manufacturer superiority. The BCA would have relationships with all the marine engine builders, so the sailboat manufacturers could order engines and check delivery dates regardless of which supplier is chosen.

Technical Challenges of the BCA based PKI Architectures

While the major impediment to deploying BCA based PKI architectures is political, there are still more technical challenges to be faced, such as efficient discovery and validation of certification paths and the interoperability of large PKI directories.

As previously noted, certification path discovery and validation is significantly harder in mesh PKIs than in a hierarchy because there are multiple trust points within the PKI and because a possibility of non-termination trust cycles exists. A BCA based PKI architecture will inevitably include some mesh PKI segments within its overall structure, resulting in the requirement that all PKI users be able to develop and validate complex certification paths. In addition, the BCA must use certificate information to constrain trust relationships between different enterprise PKIs. This implies that certificates will be more complex, and the PKI users must be prepared to process and use this additional trust information during the validation of certification paths.

Another technical challenge of BCA based PKIs, which has largely been ignored until now, is the distribution of certificates and certificate status information in a way useful to users and their applications. In an effective PKI, users must be able to readily obtain CA and user certificates and the corresponding certificate status information from a distribution mechanism. Early PKI designers expected a global X.500 directory to emerge and solve this problem. A PKI user would request specific certificates and certificate status information (specifically via certificate revocation lists (CRLs)) from the local X.500 directory. If the local X.500 directory did not contain the requested

information, the directory would be able to find it through the wonders of chaining to the global X.500 directory. Either way, a PKI user would easily be able to find the information needed by querying a local directory. The federal PKI effort is a real world example of an X.500 directory-centric PKI.

Unfortunately, the global X.500 directory did not emerge and probably never will. PKIs are being deployed using Lightweight Directory Access Protocol (LDAP) directories, web servers, and ftp servers to distribute certificates and certificate status information. In PKIs where multiple distribution mechanisms are used, user applications need to implement multiple retrieval protocols to find the information needed. These applications must be able to use complex certificates that contain location information and access protocols pointing to the relevant certificate information. The distribution solutions being fielded are not as elegant as the X.500 directories but are the reality of today. The challenge of obtaining PKI information is a difficult problem when connecting established PKIs because they most likely will have used different certificate and certificate status information distribution mechanisms.

Today's commercial off the shelf (COTS) products are not entirely prepared to meet these challenges. However, a handful of COTS products are equipped to discover and validate complex certificate paths and deal with different PKI information distribution mechanisms. As these technical challenges are resolved and incorporated into COTS products, industrial partners will reap the rewards of connecting their PKIs to enhance B2B electronic commerce. The development of industrial sector BCAs will help simplify and ease impediments to connecting PKIs of businesses with common industrial interests.

References:

1. William E. Burr, *Public Key Infrastructure (PKI) Technical Specifications: Part A - Technical Concept of Operations*. Federal Public Key Infrastructure Technical Working Group, Washington, D.C., September 1998. Available at http://csrc.nist.gov/pki/twg/welcome.html.

2. David Drucker, *Test Show PKI Promise: 'Bridge' architecture links certificates from multiple vendors*, Internet Week, April 17, 2000.

3. William E. Burr and William T. Polk, *A Federal PKI with Multiple Digital Signature Algorithms*, PKS98 Conference, April 1998. Available at http://csrc.nist.gov/pki/papers/W_E_Burr_PKCS98_paper1.doc

4. International Telecommunications Union (ITU formerly CCITT), Geneva, Switzerland. *IUT-T Recommendation X.509: Information Technology – Open Systems Interconnection – The Directory: Authenication Framework*, 1997.

5. International Telegraph and Telephone Consultative Committee (CCITT), Geneva, Switzerland. *CCITT Recommendation X.500: The Directory*, 1993.

When Should Alice Trust A Certificate?

If Alice wants to use Bob's public key, she can obtain the key from his certificate. The question is, should Alice trust the key contained in the certificate? Every certificate is digitally signed by its *issuer*-the CA that created it. Alice can trust the certificate if she can trust the digital signature that protects the integrity of the certificate's contents. This is straightforward if Alice knows and trusts the issuer.

In practice, Alice only trusts a small number of issuers. (Ideally, she trusts only one issuer.) However, Bob may get his certificate from an issuer that Alice does not know and trust. How can Alice determine if Bob's certificate is trustworthy? In a PKI, issuers can establish trust relationships for their users' convenience. If the issuer Alice trusts has a relationship with Bob's issuer, Alice may be able to use this information to trust Bob's certificate. This is accomplished by constructing and validating a *certification path*.

A certification path is a chain of certificates that uses trust relationships between issuers to determine if a certificate signed by another issuer is trustworthy. The chain starts from the public key of a CA trusted by the verifier, and ends with the certificate that contains the public key. Each certificate in the certification path is signed by its predecessor's key. A relying party verifies a signature by successively verifying the signatures on the certificates in the path. The certification path is the essential architectural construct of a PKI.

An example of a certification path is shown as Figure Sidebar-1. Alice wants to verify Bob's signature on a document. Alice trusts only the issuer called CA-1. She knows CA-1's public key, and trusts the certificates CA-1 has signed. Bob obtained his certificate from CA-2. Alice cannot verify the signature on Bob's certificate with the key she trusts. However, she can verify the signature on the certificate CA-1 issued to CA-2. Now she can rely on the public key in that certificate! She uses CA-2's public key to

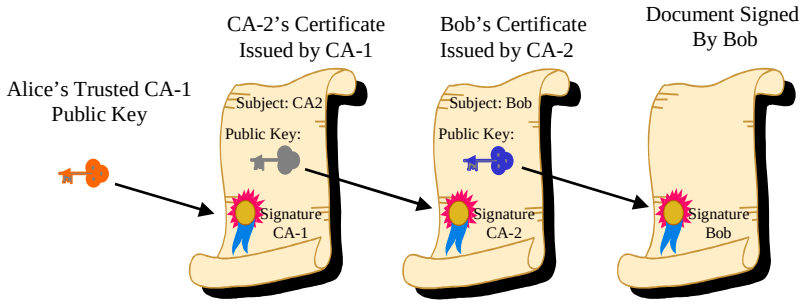


Figure Sidebar 1 - Example Certificate Path

verify the signature on Bob's certificate. Now Alice knows that she can rely on Bob's public key.

This is a very simple example. Fortunately for Alice, her CA had a direct relationship with Bob's CA. The chain of certificates may be much longer, and finding the appropriate certificates to construct the chain is one of the challenges for PKI users.