Deploying Secure Cryptographic Services in Multi-Domain IPv6 Networks

Gabriel López Millán, Félix J. García Clemente, Manuel Gil Pérez,
Gregorio Martínez Pérez, Antonio F. Gómez Skarmeta
Departamento de Ingeniería de la Información y las Comunicaciones
University of Murcia
30.071 Murcia, Spain
Email: {gabilm, fgarcia, manuel, gregorio, skarmeta}@dif.um.es

Abstract

There are several reasons to offer PKI (Public Key Infrastructure) services in IPv6 multi-domain scenarios. The first reason is to provide IPv6-only or dual-stack connectivity to those Internet users and entities who want to use certification services, but there are other important motivations. If we want to enable and promote security services in IPv6 networks, like end-to-end security, AAA (Authentication, Authorization and Accounting) services, HTTP or DNSsec services, or VPN networks, it is needed to offer the public key services required by the involved protocols. Other relevant reason is to allow services or devices to use X.509 public key certificates containing IPv6 information, such as IPv6 addresses used, for example, by any IPsec-based VPN end point. This is the main motivation of the research work presented in this paper where the most relevant design and implementation issues related with the deployment of PKI services in a multi-domain IPv6 network are presented.

1. Introduction

A public key infrastructure (PKI) is a set of hardware, software, people and procedures needed to create, manage, store, distribute and revoke public key certificates. Every PKI provides trusted and efficient private key and public key certificate management, to enable the use of authentication, non-repudiation and confidential security services. These services are mainly used by end users and security-related services using protocols such as IPsec, SSL/TLS, HTTPS, S/MIME, etc.

The University of Murcia has developed a PKI solution to offer basic and advanced certification services that can be accessed by end users and by third trusted parties using the protocols mentioned above. The design of this PKI, named UMU-PKIv6 [1], includes one Certification Authority (CA), one or more Registration Authorities (RAs) and a Request Server, acting as the interconnection point between the CA (which is defined as an off-line component) and the RAs. It is based on open-source software, as Apache, PostgreSQL, OpenLDAP or OpenSSL, and we use the Java language, which is IPv6-enabled from version 1.4 for the development of every internal component.

According to the design of the UMU-PKIv6, end users can use either the RA or a web interface to access basic services like certificate request, renewal, revocation, retrieval, etc. They can also use Java Cards [2] and RSA smart cards to store their public key certificate and private key. Every certification operation is controlled by a Certification Policy which establishes the needed restrictions inside the organisation.

The UMU-PKIv6 design is based on the IETF PKIX WG [3], so it is compliant with the [4] specification. It supports the OCSP (On-line Certificate Status Protocol) [5] and TSP (Time Stamping Protocol) [6] protocols and it allows the use of a LDAP [7] repository to store the public key material. Moreover, a DNSsec [8] server is supported as public key certificate and CRL repository.

Beside basic operations, others advanced services are defined. Certification services through CMC (Certificate Management over CMS) [9] are available to be used by third trusted parties in multi-domain scenarios, and new services like self-revocation and re-issued certificates can be used by end users. Other protocols, like SCEP (Simple Certificate Enrolment Protocol) [10] or SCP are mainly used by IPv6-enabled VPNs peers. The definition of cross-certification relationship between different CAs can also be established using Peer-to-Peer, Hierarchical or BridgeCA models. IPv6 communication between PKI internal components and end users or external entities was defined as a requirement in the design and is available from the first version of this software.

2. Components of an IPv6-enabled PKI

The UMU-PKIv6 is composed by three main components. First, the Registration Authority (RA) is in charge of validating, according to the system certification policy, and sending to the CA the different certification requests. A particular implementation may have several RAs, each one with an administrator.

The second element, the Request Server (RQServer) is in charge of storing all the certification, renewal and revocation requests generated by final users or other components of the PKI, like the RA, a process or a service. All these requests are stored in an internal database so that later the CA can access to them. It is important to mention that there is no direct connection from this server to the CA, because the CA always works in off-line mode and it never accepts incoming connections for security reasons.

Finally, the Certification Authority (CA) is responsible of processing all the requests stored in the Request Server.

The UMU-PKIv6 design and implementation also supports the use of a certificates repository. It can be a LDAPv6 directory or a DNSsec server, where all user certificates, the CA certificate and the CRLs can be stored, so that they can be consulted before establishing a secure communication with any of these entities.

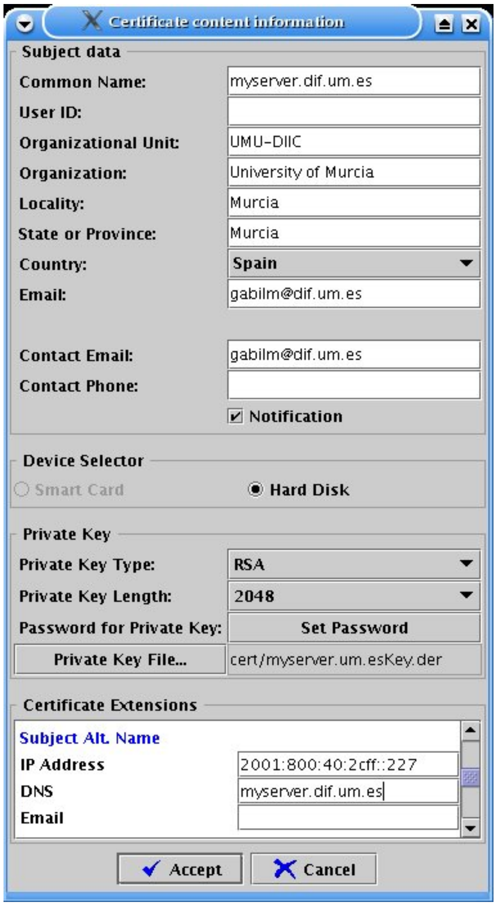
3. Deployment of the main components of an IPv6-enabled PKI

3.1. Certification authority

The CA of the UMU-PKIv6, a Java-based standalone application, processes all requests stored in the RQ Server. If it is a certification request, and the certificate can be issued, this is stored in an internal database, made public in the LDAPv6 repository and the end entity is notified through a signed email. If it is a renewal request the certificate is updated in the internal database and in the LDAPv6 repository, and if it is a revocation request the certificate is marked as revoked in the internal database and it will be included in next published CRL.

3.2. Registration authority

The RA is the component that takes charge of sending requests to be processed by a CA. The system can be formed by several RAs. The administrator will be able to carry out certification, renewal and revocation requests and he will be able to check the local Policy. The RA application supports the full use of smart cards. It is based on Java too. Figure 1 shows an IPv6-enabled certification request from the RA application.



Certificate content information

Subject data

Common Name:	myserver.dif.um.es
User ID:	
Organizational Unit:	UMU-DIIC
Organization:	University of Murcia
Locality:	Murcia
State or Province:	Murcia
Country:	Spain
Email:	gabilm@dif.um.es
Contact Email:	gabilm@dif.um.es
Contact Phone:	
	☑ Notification

Device Selector

○ Smart Card ● Hard Disk

Private Key

Private Key Type:	RSA
Private Key Length:	2048
Password for Private Key:	Set Password
Private Key File...	cert/myserver.um.esKey.der

Certificate Extensions

Subject Alt. Name

IP Address	2001:800:40:2cff::227
DNS	myserver.dif.um.es
Email	

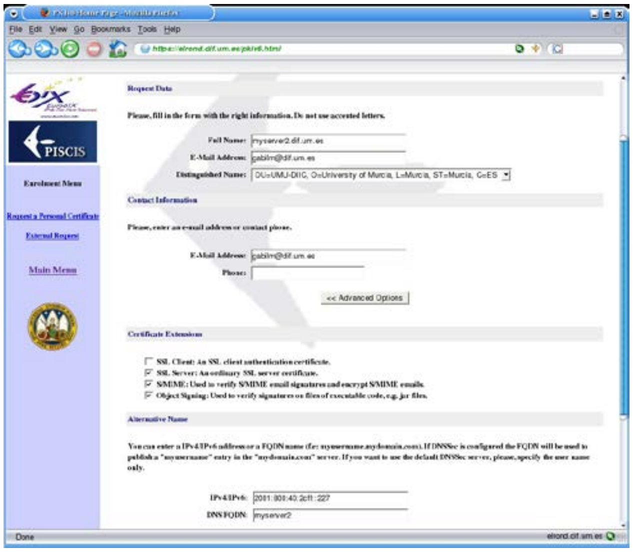
Figure 1. IPv6 certification request from the RA

3.3. Request server

The RQ is the component that stores all requests from the system entities. It stores the RA, final entities and administrator requests. These are stored in an internal database to be processed by the CA.

The RQ is based on the Tomcat HTTP server [11] with a set of IPv6-enabled servlets managing all the certification requests. Tomcat, version 5.0, can be used as a standalone HTTP Server through a connector component supporting the HTTP/1.1 protocol.

IPv6 access through Tomcat 5.0 is possible and it works properly. As Tomcat is a stable solution and it supports HTTP, Java-Servlets and IPv6, it is not necessary to use the Apache HTTP server [12] solution. Figure 2 shows an IPv6-enabled certification request from an IPv6 web interface.


PKIv6 Home Page - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
https://elrond.dif.um.es/pkiv6.html/
PISCIS
Enrolment Menu
Request a Personal Certificate
External Request
Main Menu
Done
elrond.dif.um.es

Request Data

Please, fill in the form with the right information. Do not use accented letters.

Full Name: myserver2.dif.um.es
E-Mail Address: gabilm@dif.um.es
Distinguished Name: OU=UMU-DIIC, O=University of Murcia, L=Murcia, ST=Murcia, C=ES

Contact Information

Please, enter an e-mail address or contact phone.

E-Mail Address: gabilm@dif.um.es
Phone:

<< Advanced Options

Certificate Extensions

☐ SSL Client: An SSL client authentication certificate.
☑ SSL Server: An ordinary SSL server certificate.
☑ S/MIME: Used to verify S/MIME email signatures and encrypt S/MIME emails.
☑ Object Signing: Used to verify signatures on files of executable code, e.g. jar files.

Alternative Name

You can enter a IPv4/IPv6 address or a FQDN name (f.e: myusername.mydomain.com). If DNSSec is configured the FQDN will be used to publish a "myusername" entry in the "mydomain.com" server. If you want to use the default DNSSec server, please, specify the user name only.

IPv4/IPv6: 2001:800:40:2cff::227
DNS FQDN: myserver2

Figure 2. IPv6 certification request from the web

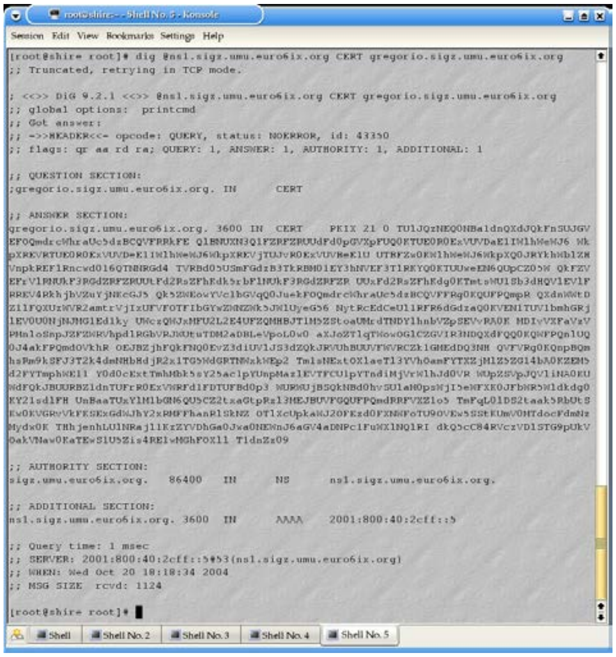
All the certification requests issued by end entities and RAs, and more useful information, are temporally stored in an internal database. The selected database software was PosgreSQL [13], which in 7.0 and later versions supports IPv6 connectivity. Other relevant open-source software solutions, like MySQL [14] can not be used because IPv6 is not supported in any of its stable releases.

3.4. Certificate repositories

The UMU-PKIv6 design and implementation supports the use of certificate repositories. In a LDAPv6 directory all user certificates, CA certificate and CRLs can be stored so that they can be consulted by users beloding to any IPv6-enabled domain.

A DNSsec server can also be used to store the public key certificates. Figure 3 shows an IPv6 DNSsec server storing user certificates.

LDAPv6 and DNSsec servers in our current implementation are based on the open-source implementations OpenLDAP 2.0.25 [15] and Bind 9.2.2 [16]. They are the standard-reference implementation and both support IPv6 connectivity in their stable releases.


root@shire:~ - Shell No. 5 - Konsole
Session Edit View Bookmarks Settings Help
Shell
Shell No. 2
Shell No. 3
Shell No. 4
Shell No. 5

```
[root@shire root]# dig @ns1.sigz.umu.euro6ix.org CERT gregorio.sigz.umu.euro6ix.org
;; Truncated, retrying in TCP mode.

; <<>> DiG 9.2.1 <<>> @ns1.sigz.umu.euro6ix.org CERT gregorio.sigz.umu.euro6ix.org
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 43350
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;gregorio.sigz.umu.euro6ix.org. IN      CERT

;; ANSWER SECTION:
gregorio.sigz.umu.euro6ix.org. 3600 IN  CERT    PKIX 21 0 TU1JQzNEQ0NBaldnQXdJQkFnSUJGV
EFOQmdrcWhraUc5dzBCQVFRRkFE Q1BNUXN3Q1FZRFZRUUdFd0pGVXpFUQ0KTUE0R0ExVUVDaE1IWlhWeWJ6 Wk
pXRkVRTUE0R0ExVUVDeE1IWlhWeWJ6WkpXRkVjTUJvR0ExVUVBeE1U UTRFZw0KWlhWeWJ6WkpXQ0JRYkhWblZM
VnpkREF1Rncwd016QTNNRGd4 TVRBd05USmFGdzB3TkRBMU1EY3hNVEF3T1RKYQ0KTUUweEN6QUpCZ05W QkFZV
EFzV1RNUkF3RGdZRFZRUUtFd2RsZFhEdk5rbFlNUkF3RGdZRFZR UUxFd2RsZFhKdg0KTmtsWU1Sb3dHQV1EV1F
RREV4RkhjbVZuYjNKcGJ5 Qk5ZWEowYVclbGVqQ0JuekFOQmdrcWhraUc5dzBCQVFFRg0KQUFPQmpR QXdnWWtD
Z1lFQXUzWVR2amtrVjIxUFVFOTFIbGYwZWNZWk5JWlUyeG56 NytRcEdCeUllRFR6dGdzaQ0KVENlTUV1bmhGRj
lEV0U0NjNJMG1Edlky UWczQWJxMFU2L2E4UFZQMHBJT1M5ZStoaUMrdTNDYlhnbVZpSEVvRA0K MDIvVXFaVzV
PMnloSnpJZFZWRVhpdlRGbVRJWUtuTDM2aDBLeVpoL0w0 aXJoZTlqTkowOGlCZGVIR3NDQXdFQQ0KQWFPQnlUQ
0J4akFPQmd0VkhR OEJBZjhFQkFNQ0EvZ3diUV1JS3dZQkJRVUhBUUVFWVRCZk1GMEdDQ3NH QVFVRg0KQnpBQm
hsRm9kSFJ3T2k4dmNHbHdjR2x1TG5NdGRTNWxkWEp2 TmlsNExtOXlaeTl3YVh0amFYTXZjM1Z5ZG14bA0KZEM5
d2FYTmphWE11 Y0d0cExtTmhMbk5sY25aclpYUnpMazlEVTFCUlpYTndiMjVrWlhJd0VR WUpZSVpJQVliNA0KU
WdFQkJBUURBZ1dnTUFrR0ExVWRFd1FDTUFBd0p3 NURMUjBSQkNBd0hvSU1aM0psWjI5eWFXK0JFbWR5W1dkdg0
KY21sdlFH UnBaaTUxY1M1bGN6QU5CZ2txaGtpRzl3MEJBUVFGQUFPQmdRRFVXZ1o5 TmFqL0lDS2taak5RbUtS
Kw0KVGRvVkFKSExGdWJhY2xRMFFhanRlSkNZ OTlXcUpkaWJ2OFKzd0FXNWFoTU9OVEw5SStKUmV0NTdocFdmNz
MydXOK THhjenhLUlNRajllKzZYVDhGa0Jwa0NEWnJ6aGV4aDNPc1FuWXlNQlRI dkQ5cC84RVczVDlSTG9pUkV
OakVNaw0KaTEwS1U5Zis4REIwMGhFOXll T1dnZz09

;; AUTHORITY SECTION:
sigz.umu.euro6ix.org.   86400   IN      NS      ns1.sigz.umu.euro6ix.org.

;; ADDITIONAL SECTION:
ns1.sigz.umu.euro6ix.org. 3600  IN      AAAA    2001:800:40:2cff::5

;; Query time: 1 msec
;; SERVER: 2001:800:40:2cff::5#53(ns1.sigz.umu.euro6ix.org)
;; WHEN: Wed Oct 20 18:18:34 2004
;; MSG SIZE  rcvd: 1124

[root@shire root]#
```

Figure 3. IPv6 DNSsec server

3.5. End entities

End entities are able to achieve certification services through IPv4 or IPv6-enabled web browsers and RAs. If it is a certification request, it is stored in the RQ Server and will be validated or removed by the RA. If it is a renewal or revocation request, it will be treated directly by the CA. Users that want to operate through web browser can make use of their smart card to store their cryptographic information thanks to a Microsoft CSP (Cryptographic Service Provider) developed as part of this PKI. It facilitates user mobility; in fact, users can request a new certificate from their web browser or an RA and their private key will be stored in their smart card. Later they can recover the associated certificate and CA s certificate from any other navigator using this cryptographic module.

3.6. Cryptographic provider

Related to all the described components, the Java cryptographic service provider software used in the UMU-PKIv6 is the IAIK Java Cryptography Extension [17], which is a set of APIs and implementations of a group of cryptographic functions, including symmetric, asymmetric, stream, and block encryption methods. It provides the security functionality of the default Java JDK 1.1.x / JDK 1.4, and also includes APIs for SSL communications and S/MIME operations. Others JCA/JCE providers were analysed; Table 1 shows a brief summary of the results. An extended comparison regarding IPv6 requirements can be found on [18].

Table 1. Comparison of the main Java providers

	Version	Open Source	JDK1.4	SSL	S/MIME	X.509	CRL	OCSP	CMS
IAIK	3.11	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
JCSI	2.3	No	Yes	Yes	Yes	Yes	Yes	No	Yes
BC	1.25	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Bee	1.02	Yes	No	No	No	Yes	Yes	No	No
Keytools	5.0	No	No	Yes	Yes	Yes	Yes	Yes	No
Phaos	-	No	Yes	Yes	Yes	Yes	Yes	No	Yes

4. Deployment of PKI services in IPv6 multi-domain scenarios

4.1. Certification request

Certification requests can be done in several ways. In the first way, users can make requests going to a RA with personal documents to prove their identity and, if it is used, their smart cards. RA issues a new request composed by a PKCS#10 object. Then it is send to the RQ Server in a secure way through SSL connection and the associated private key is stored in a file or in the user smart card.

In the second way, service administrators (e.g. VPN managers) can go to the RA with a previously generated PKCS#10 format request. This request should be authenticated by the RA administrator and by the system certification policy. If the validation process is correct, the request will be send to the RQ Server to be processed.

Besides this, applications/services are able to be certified through a KMP (Key Management Protocol) protocol like CMC [9] or SCEP [10], which allows devices to carry out certification operations. VPN devices usually use these services.

In the last way, users can issue a new certification request through the web, using their own browser and, optionally, their smart card. In this way, users send requests to the RQ Server. The RA administrator recovers his request and validates it. Then, the request is sent to the RQ again, but marked to be finally processed (i.e., signed and published in a LDAP and/or DNSsec server) by the CA application.

4.2. Renewal request

Final entities (i.e., users, software processes and devices) can update their certificate validity period according to the system certification policy. This request can be done going to the RA in administrative hours or by means of the UMU-PKIv6 web site using for this a SSL or TLS connection with client authentication (i.e., the final entity will present its personal certificate to the RQ).

4.3. Revocation request

In exceptional conditions, as lost or compromised smart card, digital certificates are invalidated before their validity period expires. Users have two options to revoke their certificates. The first one is to go to the RA and request a certificate revocation. The second one is to issue this in the web site. The last option is only possible if the user still has his certificate private key.

4.4. Key management protocol services

Certificate Management Messages over CMS (CMC) [9] is a standard under development by the PKIX IETF Working Group [3]. CMS provides basic cryptographic services including encryption and signing with and without key management. The CMC protocol is intended to address the need for an interface to public key certification products and services based on PKCS#10 requests.

UMU-PKIv6 has implemented the CMC protocol based on Java and Perl interfaces where it offers six services, i.e., enrolment, revocation and renewal requests, get certificates, get CRLs and query about the status of a pending certificate request. End entities using CMC can request on-line certification services using either IPv4 or IPv6 networks, as it has been considered as a requirement from the early stage of the design and implementation of the CMC system.

4.5. Cross-certification services

Cross-certification is a process carried out by CAs to establish trust relationships between them. When two CAs are cross-certified, all their PKCs (Public Key Certificates) and keys trust the other ones as if all would belong to the same organization. To make it possible, two CAs should exchange their cross-certificates.

In others words, cross-certification is used to allow client systems or end entities in one administrative domain to communicate securely with client systems or end users in another administrative domain. It is quite important for the proper deployment of IPv6 networks as multi-domain scenarios are envisaged, as wells as roaming users moving from one home domain to a foreign domain.

The UMU-PKIv6 design and implementation supports the definition of Hierarchical, Peer-to-Peer and BridgeCA models as described in [19].

Hierarchical and Peer-to-Peer cross-certification have been implemented like internal modules inside the CA application. The CA administrator can issue cross-certificates to subordinate and external CAs checking and validating the certificate extensions, which establish the relationship constraints.

BridgeCA (BCA) cross-certification is implemented like a IPv6-enabled Java-based standalone application, allowing the establishment of bidirectional cross-certification. To define that relationship, the BCA administration can issue a cross-certification request, and waits to be signed by the other external CA (the *forward* certificate). Beside this, the BCA administrator can accept and sign cross-certification requests issued by the external CA (the *reverse* certificate). Before those certificates are issued the restrictions over the relationship are defined by the certificate extensions included in the cross-certificates. UMU-PKIv6 allows the use of the following extensions: *BasicConstraints, CertificatePolicies, PolicyMapping, NameConstraints* and *PolicyConstraints.*

The use of cross-certification involves the certification paths building, composed by the Root CA, intermediate CAs and end entity certificates. To discover and validate these paths, validation services must be defined. Those validation services use CRLs, OCSP services and certificate extensions as those presented through this paper.

The cross-certificate pair used to describe a trust relationship between two CAs are stored in a LDAPv6 repository, using the *crossCertificatePair*. This attribute is used by the validation service to discover the certification paths.

To support revocation of CA certificates, UMU-PKIv6 supports the use of ARLs (Authority Revocation List) implemented like a *issuingDistributionPoint* with the *onlyContainsCACerts* flag active.

5. Conclusions

Most of the security services (e.g., VPNs, SSL communications, S/MIME signed and encrypted email messages) need public key certification services to work properly. This is a reality for both IPv4 and IPv6 networks. However, the level of deployment of such security services is far from being mature in the case of IPv6 networks. As IPv6 is increasing its presence in the local and backbone networks, research and deployment in IPv6 security services is a must that need to be addressed.

In this sense the UMU-PKIv6 design and implementation tries to help in this convergence needed between security services and IPv6 networks, offering basic and advanced certification services. The extension to support multi-domain IPv6 scenarios through the deployment of new cross-certification modules represents also an interesting feature of the UMU-PKIv6.

Acknowledgements

This work has been partially funded by the EU Euro6IX project IST-2001-32161.

References

[1] UMU-PKIv6, University of Murcia Public Key Infrastructure with IPv6 support, University of Murcia (UMU), http://pki.dif.um.es

[2] Sun Microsystems, “Java Card Technology”, http://java.sun.com/products/javacard/

[3] IETF, “Public-Key Infrastructure (X.509) (pkix) Working Group”, http://www.ietf.org/html.charters/pkix-charter.html

[4] R. Housley, W. Ford, D. Solo, “Internet Public Key Infrastructure, X.509 Certificate and CRL Profile”, Request for Comments (RFC) 3280, April 2002.

[5] M. Myers, R. Ankney, A. Malpani, S. Galperin, C. Adams, “OCSP: Online Certificate Status Protocol”, IETF, Request for Comments (RFC) 2560, June 1999.

[6] C. Adams, P. Cain, D. Pinkas, R. Zuccherato, “Time-Stamp Protocol (TSP)”, IETF, Request for Comments (RFC) 3161, Aug. 2001.

[7] S. Boeyen, T. Howes, P. Richard, “Internet X.509 Public Key Infrastructure LDAPv2 Schema”, IETF, Request for Comments (RFC) 2587, June 1999.

[8] DNS Security Extensions, http://www.dnssec.net

[9] M. Mayers, X. Liu, J. Schaad, J. Weinstein. “Certificate Management Messages over CMS”, IETF, Request for Comments (RFC) 2797, April 2000.

[10] X. Liu, C. Madson, D. McGrew, A. Nourse, “Simple Certificate Enrolment Protocol (SCEP)”, IETF, Internet Draft, June 2004.

[11] Apache Jakarta Tomcat Project, http://jakarta.apache.org/tomcat

[12] Apache HTTP Server Project, http://httpd.apache.org

[13] PostgreSQL Database, http://www.postgresql.org

[14] MySQL Database, http://www.mysql.com

[15] OpenLDAP Community, http://openldap.org

[16] Berkeley Internet Name Domain, http://www.isc.org/index.pl?/sw/bind/

[17] Institute for Applied Information Processing and Communications, IAIK-JCE, http://jce.iaik.tugraz.at

[18] A.F. Gómez Skarmeta, G. Martínez Pérez, O. Cánovas Reverte, G. López Millán, “PKI Services for IPv6 Networks”, IEEE Internet Computing, May-June 2003.

[19] L. Steve, D. Fillingham, R. Lampard, S. Orlowski, “CA-CA Interoperability”, PKI Forum, March 2001.