Clarifying PKI interoperability

Is there a topic in PKI more important and yet more confusing than "interoperability"? A senior finance sector executive captured the uncertainty perfectly: *"[PKI] interoperability is something of a will-o'-the-wisp. You think you understand what people mean by it, and then quickly realise that you don't. In my experience, it's possible when discussing interoperability to be at cross-purposes for all of the time. Interoperability between members of the same PKI is axiomatic. Certificates issued by one bank should be recognisable by another. Interoperability becomes an issue when it is between different PKIs … But this still leaves the basic question of interoperable in respect of what?"*.[1] Indeed, interoperability is *so* "axiomatic" that many pivotal papers on the topic (like [1]) omit to define the term, or to spell out its precise objectives.

It really isn't complicated

The best place to start thinking about interoperability is to unpack how digital certificates can help with the act of authentication. A fine definition of authentication comes from the APEC eSecurity Task Group: "The means by which the recipient of a transaction or message can make an assessment as to whether to accept or reject that transaction" [2]. In the case of digital certificates, from the perspective of the receiver or Relying Party, the central question is very simple: *What information is available – in the certificate chain and elsewhere – to help the receiver decide whether to accept or reject the certificate and hence the message?*

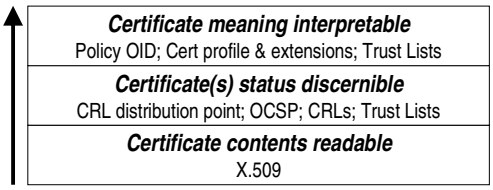
What does the receiver need to know?

There are three main things the receiver needs to know about a certificate.

1. ***What representations does the certificate make about its holder?*** Or equivalently, was the certificate intended to be used in the transaction concerned? Increasingly, digital certificates are used to represent specific credentials or memberships and to thereby confer particular authorisations [3]. For example, a certificate issued by a medical authority can confer the rights of the holder to write prescriptions, claim government rebates and so on. Such digital credentials will bear a unique Policy OID, and perhaps the holder's registration numbers as well.
2. ***Is the certificate subject still valid (i.e. not revoked)?*** Sometimes the question will be "backdated"; that is, was the certificate holder valid at the time they launched the transaction?
3. ***Was the certificate issuer acting in compliance with applicable standards and regulations?*** Relevant standards will vary from one domain to another; examples include Gatekeeper, Identrus and WebTrust for CAs. A CA's status should be made available online; one way to do this is for a regulator to issue a compliance certificate to the CA, like the proposed *Gatekeeper Accreditation Certificate*.

An *interoperability stack*

The diagram shows how the information needed to verify a certificate can be obtained, and which standards help make that information available.[2] To be able to read a certificate at all requires it to comply with a syntactical standard like X.509. For the status of a certificate to be discernible, we need to know where to find Certificate Revocation Lists and the like. And for the representations made by a certificate to make sense, we need to know about the Policy Object Identifier, the profile, and have access to the public keys of trusted Root CAs (via for instance a "Trust List"). Crucially, this last layer of information should be available at design time.



Everything could be in the certificate chain

This stack should support the ultimate goal of "application level" interoperability. In fact, all the information an application needs in order to accept or reject a certificate could be found in the certificate chain, under the right circumstances. We need to be clear what certificates issued to CAs represent. If they represent each CA's compliance with standards (like Gatekeeper or Identrus) then when an End User certificate chains back to the Root we can be sure that all intermediate CAs are doing what they're supposed to do. And if the End User certificate's Policy OID matches our expected value, then the certificate can be relied upon.

References

[1] *CA-CA Interoperability*, The PKI Forum 2001 www.pkiforum.org/pdfs/ca-ca_interop.pdf

[2] *Public Key Authentication Task Group Preliminary Report*, APEC 1997 www.apectelwg.org/contents/documents/eaTG/eaTG-1.html

[3] *Relationship Certificates*, www.lockstep.com.au/library/pki/relationship_certificates

[1] http://tinyurl.com/hj5v9

[2] Verifying a digital signature involves cryptographic processes covered by a host of relatively mature API standards at a deeper layer, not shown in the diagram.