The Realities of PKI Inter-operability

John Hughes

Claridge House
29 Barnes High Street
London, SW13 9LW
England

john.hughes@entegrity.com

Many vendors are claiming that their products are “open” and “inter-operable”. This paper is intended to explore what this could mean, and in reality what is available in the market place, highlighting any issues found by the author.

1 Introduction

Public Key Infrastructure (PKI) offers a method of protecting an enterprise's electronic communications using public key cryptography. Because PKI is an “infrastructure”, it is usually necessary to obtain board-level approval for funding or to find some sponsor in the enterprise. As this is sometime difficult to justify local business units are implementing secure solutions, that just happen to be PKI based. As a result of this situation PKI Islands are developing.

PKI Islands have the advantage of allowing an enterprise to seed PKI throughout its business without the need to undergo the turmoil of a “big bang” – an approach sometimes termed “PKI by Stealth”. However, at some stage an enterprise will need to connect the Islands together and also secure its communications with trading partners. If inter-operability issues have been ignored at the design stage, problems will almost certainly emerge later.

This paper summarises some of the technical issues of making various PKI components inter-operate (or why they cannot operate with each other!). Entegrity Solutions have defined an Inter-operability Model which is used within the company to examine areas of product deficiency and product enhancements providing additional flexibility.

2 Inter-operability Model

Whist the PKIX have defined a Certificate and CRL profile, one has not been produced for all elements of a PKI. The intent of the Model is to assist in formulating such profiles. Following a summary of the model the paper summaries some of our experiences in achieving interoperability.

The major elements that the Interoperability Model covers are summarized in the following paragraphs.
Key Generation. Three basic key generation schemes are possible: Centralized at the CA, at the EE Client or to have Split Keys. In this last case one set of keys are generated at the EE for a particular set of purposes and another set generated at the CA.

Encapsulation protocol. If the public key is generated at the EE then a means of securely sending the public key and receiving back the certificate is required. Four main encapsulation protocols are (or will be soon) available from products. These protocols are PKCS#10-PKCS#7, Verisign’s CRS, PKIX CMP and PKIX CMC.

Transport. The encapsulation protocols provide message protection. However they still have to be transported around the network. Whilst web-based http is clearly the natural (and dominant) technique, it is actually quite complicated partly due to the authentication issue (see below). As those with Internet issued certificates will be aware, obtaining them for a Web Browser is a painful multi-step process, involving a number of Web dialogues plus an e-mail transaction. Because it is a manual, user driven process, most Web Browsers can interact quite successfully with CAs. However dealing with other applications illustrates the point that in general PKI-enabled applications are not well integrated with CAs.

Authentication. There are two basic schemes involved in authenticating the owner of a public key prior to its certification. In the case of class 1 VeriSign certificates, no real authentication is actually performed. Therefore anyone could claim a false identity and obtain a certificate issued in that name. This is not the case though for higher grade certificates, such as VeriSign class 3 certificates. The authentication schemes generally available are:

- Manual approval at the CA
- Automated approval at the CA using some type of “secret value” look-up
- Centralized Token issuing with pre-authorization information.

Token/PSE Format. There is no widely deployed standard that currently defines the format and contents of the PSE. Currently a new standard is being drafted with the intention of defining the PSE for Smart Cards – this is PKCS#15.

Token Plug and Play. When using physical cryptographic tokens the emerging dominant API standard is PKCS#11. Most smart card vendors and high-end cryptographic accelerators support PKCS#11. PKCS#11 defines an API to add, modify and use cryptographic keys on the Token. The intention being for a PKI application supporting PKCS#11 to plug in any PKCS#11 Token. In reality it’s not quite that simple.

Publication/Retrieval protocol. The market leader in this area is the LDAP protocol. The version of LDAP most widely used at present is version 2 (LDAPv2), but increasingly LDAPv3 server products are appearing.

Publication Protection. Certificates and CRLs are self-protecting objects, and therefore the connection for publishing them on the LDAP server, at first sight, does not seem to require protection. However as the connection requires “write access” to the server, it is important to control access to limit those network entities that can write to the server. The techniques used to protect this connection, include SSL.