

Commercial in Confidence



View500 and PKI - an eB2Bcom White Paper



Date: 10 May 2007

Version: 1.0

Prepared By: eB2Bcom Pty Ltd
Suite 1, 85-87 Charles St
Kew, Vic 3101 Australia
Phone: +61-398518600
Fax: +61-398518601

DOCUMENT DETAILS

Author: Steven Legg Phone: +61 3 9851 8600

DOCUMENT SECURITY

This document contains intellectual property of eB2Bcom, and is provided on a COMMERCIAL-IN-CONFIDENCE basis to the party ("Client") named on the front page. It is solely for the use by the staff of Client for the consideration and evaluation of eB2Bcom's proposals. It must not be provided in whole or in part to third parties without the express approval of an authorised officer of eB2Bcom.

REVIEW

Reviewed by:

DOCUMENT HISTORY

Version	Comment	Date
0.1	First draft – Steven Legg	10 May 2007

TABLE OF CONTENTS

DOCUMENT DETAILS	2
DOCUMENT SECURITY	2
REVIEW	2
DOCUMENT HISTORY	2
TABLE OF CONTENTS	3
CREDENTIALS.....	4
INTRODUCTION	4
DIRECTORIES AS PKI REPOSITORIES.....	4
EXTRACTED ATTRIBUTES VERSUS MATCHING RULES.....	5
SUPPORT FOR PKI MATCHING RULES.....	6
CONCLUSION	7
REFERENCES.....	7

CREDENTIALS

eB2Bcom specialises in security, identity management, and associated infrastructure. We are an established provider of software and services to Defence, Intelligence, Police and Government Agencies as well as the private sector. eB2Bcom has offices and works with a range of partners throughout South East Asia, Australia and New Zealand.

INTRODUCTION

This eB2Bcom white paper describes and compares the three main ways in which Public Key Infrastructure (PKI) applications might use an LDAP or X.500 directory as a repository for application data, and points out the advantages of using eB2Bcom's View500 directory server in such deployments.

DIRECTORIES AS PKI REPOSITORIES

LDAP or X.500 directories are frequently used by X.509 Public Key Infrastructure (PKI) applications as repositories for certificates and certificate revocation lists (CRLs). There are three approaches PKI applications can take to finding certificates and CRLs of interest that have been stored in entries in a directory.

- 1) The PKI application fetches the certificates and CRLs from all the likely candidate entries and filters the returned results itself to identify the data of interest.
- 2) When storing a certificate or CRL in the directory, the PKI application pulls out the values of certain component parts of the certificate or CRL and stores these values as separate attributes alongside the certificate or CRL. This is sometimes called the extracted attributes approach. For example, the serial number from a certificate is extracted and stored as a separate serial number attribute in the same directory entry as the certificate. The serial number attribute can then be used in a search filter to find an entry containing a certificate with a particular serial number. Since the serial number attribute has a primitive syntax (i.e., integer), all off-the-shelf directory implementations can handle indexing and searching of such an attribute. In a typical deployment, a number of components of certificates and CRLs would be extracted as separate attributes.
- 3) The directory client uses PKI matching rules to directly match certificates and CRLs having particular values for nominated components. For example, the certificateMatch matching rule can be used in a search filter to find an entry containing a certificate with a particular serial number. A directory server that supports these matching rules can handle in-situ indexing and searching of the various component parts of certificates and CRLs, without those components needing to be extracted out as separate directory attributes. This means that the client is freed from the requirement to provide and maintain the extracted attributes.

The first approach can be very inefficient if large numbers of certificates or CRLs must be examined. The extracted attributes approach and the PKI matching rules can have comparable performance, however the extracted attributes approach has a number of disadvantages compared to using the PKI matching rules.

EXTRACTED ATTRIBUTES VERSUS MATCHING RULES

The extracted attributes approach makes the directory database larger since information in certificates and CRLs is stored redundantly in the extracted attributes. Accompanying this duplication of information is a risk of the extracted attributes and the corresponding components of the certificates and CRLs becoming inconsistent because of unregulated modifications to the extracted attributes by third-party directory clients. Where such an inconsistency arises, searches using the extracted attributes will no longer produce reliable results.

A perfect association between the values of extracted attributes and the values of the components of a certificate can only exist if there is a single certificate in a directory entry; likewise for CRLs. If an entry contains multiple certificates or multiple CRLs then the association between the values of the extracted attributes and each specific certificate or CRL is lost. Suppose a search filter nominates a particular value of the extracted attribute for the certificate serial number and a particular value of the extracted attribute for the certificate issuer. Assuming that the extracted attributes are consistent with the certificates, an entry will satisfy the search filter if it has a certificate with that serial number and at least one certificate with that issuer, but there is no guarantee that it is the same certificate that satisfies both conditions. The search is thus prone to returning false positive matches. The PKI matching rules don't have this problem because they are evaluated on a certificate or CRL in isolation.

The solution to the problem of false positives is to store each certificate or CRL in a separate directory entry (typically as subordinate entries of the certificate subject's entry), but this has its own problems because it dissociates a certificate and its extracted attributes from the other attributes of the subject. So that it is still possible to search on both extracted attributes and the subject's other attributes, these other attributes must be duplicated in each entry that stores one of the subject's certificates. To support directory clients that look for a subject's certificates in an entry that has the subject's name it is necessary to retain copies of each certificate in the subject's entry. Both these requirements cause a further increase in the size of the directory's database, add more complexity to the PKI application with regard to managing the subjects' directory entries, and provide more opportunities for inconsistencies to arise.

Since there is no standardized set of extracted attributes, two independently developed PKI clients might not extract the components of certificates and CRLs into attributes of the same name and might not even extract the same set of components. Two such clients using the same directory as a certificate repository are likely to interfere with each others' extracted attributes and create inconsistencies. In contrast, the PKI matching rules for use in X.500 directories are standardized by X.509, and RFC 4523 standardizes their representation for LDAP directories.

When the PKI matching rules are available, a PKI application can store each certificate or CRL exactly once in the appropriate entry. It does not need to create any extracted attributes or maintain multiple copies of certificates or CRLs. With no duplication of information, less storage space is required, update processing is simpler and there are no consistency issues. Because certificates and CRLs are matched directly, the PKI application does not need to deal with false positive matches.

With the PKI matching rules, the directory works for the PKI application instead of the PKI application having to work around deficiencies in the directory.

SUPPORT FOR PKI MATCHING RULES

X.509 certificates can be stored by most directories, however, few directories support any matching rules for the PKI attribute syntaxes. View500 is one of these few. The current version of View500 supports these matching rules from X.509:

- certificateExactMatch
- certificateMatch
- certificatePairExactMatch
- certificatePairMatch
- certificateListExactMatch
- certificateListMatch
- algorithmIdentifierMatch
- attributeCertificateExactMatch
- attributeCertificateMatch

View500 has index support for each of these matching rules so that the entries of interest can be quickly found regardless of the total number of entries in the directory. A single View500 server can easily hold tens of millions of entries.

The eB2Bcom development team can readily add support for additional X.509 matching rules if a customer or partner requires them.

RFC 4523 defines the LDAP representation for most of the PKI matching rules. As an example, an LDAP filter to search for the entry containing a user certificate with the serial number 12345 would look like this:

```
(userCertificate:certificateMatch:={ serialNumber 12345 })
```

View500 also supports the component matching rules (RFC 3687), which provide matching capabilities beyond those specified by the matching rules in X.509.

For example, the certificateMatch matching rule allows searching for certificates that have any email address as a subject alternative name, but does not provide the ability to search for a certificate with a particular email address. However, this latter capability is provided by the component matching rules.

Using certificateMatch, the LDAP filter to find all entries containing certificates with an email address as a subject alternative name would be¹:

```
(userCertificate:certificateMatch:=  
  { subjectAltName builtinNameForm:rfc822Name })
```

Using component matching, the LDAP filter to find an entry containing a user certificate that has, for example, bob@eb2bcom.com as a subject alternative name would be:

```
(userCertificate:componentFilterMatch:=item:{  
  component "toBeSigned.extensions.*.extnValue.  
  content.(2.5.29.17).*rfc822Name",  
  rule caseIgnoreIA5Match, value "bob@eb2bcom.com" })
```

The above component filter uses a single assertion on a single component of a user certificate. However, it is possible to construct component filter expressions containing

¹ Line breaks have been added for improved readability.

Commercial in Confidence

multiple assertions over multiple components connected by the boolean operators AND, OR and NOT. In this way it is possible to express arbitrarily complex conditions on the PKI attribute values to be matched. The component matching rules can be applied to any component of any attribute syntax, in contrast to the attribute extraction approach where only those components that have been explicitly extracted as separate attributes are available to be searched. Together the PKI matching rules and component matching rules provide a more comprehensive and precise search capability than the attribute extraction approach.

Component matching is fully supported by View500 for the two hundred plus attribute syntaxes and forty-seven certificate extensions known to the directory server. The list of supported attribute syntaxes includes the Certificate, CertificateList, CertificatePair, SupportedAlgorithm, AttributeCertificate, InfoSyntax, PolicySyntax, PkiPath attribute syntaxes from X.509. The eB2Bcom development team can readily add support for additional attribute syntaxes or certificate extensions if a customer or partner requires them.

View500 allows any component of any attribute syntax to be indexed, which allows rapid evaluation of component filter searches regardless of the total number of entries in the directory.

CONCLUSION

PKI applications using an LDAP or X.500 directory as the certificate and CRL repository can process certificates faster, more reliably and with less effort if that directory supports the PKI matching rules and/or component matching rules. The eB2Bcom View500 directory server has comprehensive support for both the PKI matching rules and the component matching rules.

REFERENCES

ITU-T Recommendation X.509 (08/05) | ISO/IEC 9594-1:2005, **Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks**

Legg, S., **Lightweight Directory Access Protocol (LDAP) and X.500 Component Matching Rules**, RFC 3687, February 2004
(<http://www.ietf.org/rfc/rfc3687.txt>).

Zeilenga, K., **Lightweight Directory Access Protocol (LDAP) Schema Definitions for X.509 Certificates**, RFC 4523, June 2006
(<http://www.ietf.org/rfc/rfc4523.txt>).