Internet Draft
draft-hoffman-pkix-stringmatch-00.txt
July 7, 2004
Expires in six months

Paul Hoffman
Internet Mail Consortium
Steve Hanna
Sun Microsystems

Matching Text Strings in PKIX Certificates

Abstract

Strings of text appear in many fields in PKIX certificates. Some applications need to compare the values in these fields to strings from other certificates, or to values obtained in other manners. For many string encodings, this can be done in an octet-by-octet fashion. Other encodings, however, require preparation before the strings can be compared. This document describes that preparation and when it needs to be applied.

1. Introduction

Because of the shared background of PKIX, LDAP, and X.500, matching strings from PKIX certificates uses the same principles as matching strings from X.500 directories. Therefore, this document simply refers to the specification for matching LDAP strings. The [LDAP-STRINGPREP] document fully specifies all the steps needed for string comparison. This specification applies to all string encodings supported by PKIX.

When matching names in certificates, having consistent and well-defined matching rules is important. Otherwise, many security problems can occur, as described in the Security Considerations section of this document. In addition to these security problems, there will likely be usability problems. In Unicode, there are often several valid ways to encode a string. For example, the character for "e with an accent" can be represented either as a single character, or as a string of two characters: the letter "e" followed by the combining accent character. These should be considered equivalent. With stringprep, they are, but with binary comparison, they are not.

1.1 Terminology

The key words "MUST", "SHALL", "REQUIRED", "SHOULD", "RECOMMENDED", and "MAY" in this document are to be interpreted as described in RFC 2119 [KEYWORDS].

2. String Comparison

To compare a string from a PKIX certificate to another string, both strings MUST be prepared as specified in section 2 of [LDAP-STRINGPREP]. All six sub-sections of that specification MUST be performed, and they MUST be performed in the order specified.

2.1 Applicability of this specification

The most important certificate fields that this specification applies to are those with names that are created by and for people. These include any place a Name, GeneralName, or GeneralNames structures. The most commonly used fields in an X.509 certificate that use these structures are issuer, subject, subjectAltName, NameConstraints, and DistributionPoint.

This algorithm SHOULD NOT be used for matching CRL distribution point names or cRLIssuer. A binary comparison SHOULD be used for such mapping. There is no reason why a distribution point name or cRLIssuer should not match exactly because they are both generated by the same party or its CRL issuer. Also, an interloper might use an inexact match to forge a CRL by getting a certificate whose subject name matches the cRLIssuer contained in another certificate due to the inexact match.

3. Security considerations

All security considerations from [LDAP-STRINGPREP] are inherited. In addition to the security considerations mentioned in [LDAP-STRINGPREP], there are several others specific to certificates.

Matching of X.500 names currently varies widely in different implementations. Some do a binary comparison for UTF8String. Some attempt to do a more sophisticated case-insensitive comparison but use an algorithm different than the one given in this specification. Some will soon be using this algorithm. We can expect that such wide variations will continue in the future during a transition period and even beyond this period if bugs are present.

Often, such variations will cause systems to fail safe. Path validation will fail when a relying party uses binary comparison to match X.500 names that differ slightly if the match is performed as part of subject-issuer name chaining or permittedSubtrees processing.

However, security vulnerabilities may be opened if the CA assumed binary comparison would be used and the relying party uses the more lenient matching for subject-issuer name chaining or permittedSubtrees processing. For subject-issuer name chaining, the signature check should catch the problem, but for permittedSubtrees processing, the problem remains. In specific, an X.500 name that the CA did not intend to

include in the permittedSubtrees might be allowed. It is not clear that this is a substantial problem since the name will likely be very similar in appearance to a name that was meant to be allowed.

More substantial problems can occur if a relying party uses binary comparison to determine that an X.500 name does not match excludedSubtrees and the CA expected that it would match because of the more lenient processing. However, this problem is probably already present since some relying parties do not perform case-insensitive comparisons of PrintableString components in a DirectoryString, even though it is required by RFC 3280. To avoid these problems, systems need to avoid the use of excludedSubtrees unless they are satisfied with a binary match against subject names and subject alternative names.

4. References

4.1 Normative references

[KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[LDAP-STRINGPREP] Zeilenga, K., LDAP: Internationalized String Preparation draft-ietf-ldapbis-strprep, work in progress

5. Author's address

Paul Hoffman
Internet Mail Consortium
phoffman@imc.org

Steve Hanna
Steve.Hanna@Sun.COM

6. Open issues

Should this specification apply to all names in an X.509 certificate, or only DirectoryStrings in X.500 names?

Should this spec apply to self-signed certificates, or should they be an exception?

7. Notices

INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

</x-flowed>