INTERNET-DRAFT
Intended Status: Proposed Standard
Updates: 5280 (if approved)
Expires: September 13, 2012

D. Cooper
NIST

March 12, 2012

Updates to the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile <draft-ietf-pkix-rfc5280-clarifications-04.txt>

Status of this Memo

Copyright and License Notice

Abstract

This document updates RFC 5280, the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. This document changes the set of acceptable encoding methods for the explicitText field of the user notice policy qualifier and clarifies the rules for converting internationalized domain name labels to ASCII.

Table of Contents

1. Introduction

This document updates the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile [RFC5280].

The ASN.1 [X.680] syntax for the user notice certificate policy qualifier allows for the explicitText field to be encoded using one of four possible encoding methods: IA5String, VisibleString, BMPString, or UTF8String. RFC 5280 permits certification authorities (CA) to encode strings in the explicitText field as either UTF8String or IA5String while forbidding the use of VisibleString and BMPString. However, after RFC 5280 was published, an examination of existing certificates found that the VisibleString encoding was commonly used. This document brings the requirements into closer alignment with existing practice by stating that the explicitText field may be encoded in either UTF8String, VisibleString, or BMPString while forbidding the use of IA5String.

Section 7.3 of RFC 5280 specifies rules for converting

internationalized domain name labels that are to appear in a domainComponent attribute to ASCII. The conversion process specified in RFC 5280 did not specify that the "UseSTD3ASCIIRules" flag needed to be set. This document modifies the conversion process specified in Section 7.3 of RFC 5280 to clarify that "UseSTD3ASCIIRules" flag should be set. The result of this is to indicate that the check for conformance to [RFC1123] should be performed.

1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Update to RFC 5280, Section 3.2: Certification Paths and Trust

Add the following paragraph to the end of RFC 5280, Section 3.2:

| In some cases, a self-signed certificate is used to convey a public
| key that is not intended to be used to begin certification paths.
| Such a self-signed certificate is commonly referred to as a self-
| signed end-entity certificate. While Section 6.2 notes that
| implementations are not required to process extensions that appear in
| self-signed certificates, self-signed end-entity certificates SHOULD
| NOT include a BasicConstraints extension that asserts the cA boolean.
| The use of self-signed certificates for purposes other than to convey
| a public key for use to begin certification paths is outside the
| scope of this specification.

3. Update to RFC 5280, Section 4.2.1.4: Certificate Policies

RFC 5280, Section 4.2.1.4, the tenth paragraph says:

| An explicitText field includes the textual statement directly in
| the certificate. The explicitText field is a string with a
| maximum size of 200 characters. Conforming CAs SHOULD use the
| UTF8String encoding for explicitText, but MAY use IA5String.
| Conforming CAs MUST NOT encode explicitText as VisibleString or
| BMPString. The explicitText string SHOULD NOT include any control
| characters (e.g., U+0000 to U+001F and U+007F to U+009F). When
| the UTF8String encoding is used, all character sequences SHOULD be
| normalized according to Unicode normalization form C (NFC) [NFC].

This paragraph is replaced with:

| An explicitText field includes the textual statement directly in
| the certificate. The explicitText field is a string with a
| maximum size of 200 characters. Conforming CAs SHOULD use the

| UTF8String encoding for explicitText, but MAY use VisibleString or
| BMPString. Conforming CAs MUST NOT encode explicitText as
| IA5String. The explicitText string SHOULD NOT include any control
| characters (e.g., U+0000 to U+001F and U+007F to U+009F). When
| the UTF8String or BMPString encoding is used, all character
| sequences SHOULD be normalized according to Unicode normalization
| form C (NFC) [NFC].

4. Update to RFC 5280, Section 6.2: Using the Path Validation Algorithm

RFC 5280, Section 6.2, the third paragraph says:

| Where a CA distributes self-signed certificates to specify trust
| anchor information, certificate extensions can be used to specify
| recommended inputs to path validation. For example, a policy
| constraints extension could be included in the self-signed
| certificate to indicate that paths beginning with this trust anchor
| should be trusted only for the specified policies. Similarly, a name
| constraints extension could be included to indicate that paths
| beginning with this trust anchor should be trusted only for the
| specified name spaces. The path validation algorithm presented in
| Section 6.1 does not assume that trust anchor information is provided
| in self-signed certificates and does not specify processing rules for
| additional information included in such certificates.
| Implementations that use self-signed certificates to specify trust
| anchor information are free to process or ignore such information.

This paragraph is replaced with:

| Where a CA distributes self-signed certificates to specify trust
| anchor information, certificate extensions can be used to specify
| recommended inputs to path validation. For example, a policy
| constraints extension could be included in the self-signed
| certificate to indicate that paths beginning with this trust anchor
| should be trusted only for the specified policies. Similarly, a name
| constraints extension could be included to indicate that paths
| beginning with this trust anchor should be trusted only for the
| specified name spaces. While the path validation algorithm presented
| in Section 6.1 does not assume that trust anchor information is
| provided in self-signed certificates and does not specify processing
| rules for additional information included in such certificates,
| [RFC5937] provides an example of how additional information included
| in self-signed certificates may be used to initialize the path
| validation inputs. Implementations that use self-signed certificates
| to specify trust anchor information are free to process any
| additional information that is included in the certificates or ignore
| such information.

5. Update to RFC 5280, Section 7.3: Internationalized Domain Names in Distinguished Names

RFC 5280, Section 7.3, the first paragraph says:

> Domain Names may also be represented as distinguished names using domain components in the subject field, the issuer field, the subjectAltName extension, or the issuerAltName extension. As with the dNSName in the GeneralName type, the value of this attribute is defined as an IA5String. Each domainComponent attribute represents a single label. To represent a label from an IDN in the distinguished name, the implementation MUST perform the "ToASCII" label conversion specified in Section 4.1 of RFC 3490. The label SHALL be considered a "stored string". That is, the AllowUnassigned flag SHALL NOT be set.

This paragraph is replaced with:

> Domain Names may also be represented as distinguished names using domain components in the subject field, the issuer field, the subjectAltName extension, or the issuerAltName extension. As with the dNSName in the GeneralName type, the value of this attribute is defined as an IA5String. Each domainComponent attribute represents a single label. To represent a label from an IDN in the distinguished name, the implementation MUST perform the "ToASCII" label conversion specified in Section 4.1 of RFC 3490 with the UseSTD3ASCIIRules flag set. The label SHALL be considered a "stored string". That is, the AllowUnassigned flag SHALL NOT be set. The conversion process is the same as is performed in step 4 in Section 7.2.

6. Security Considerations

This document modifies the Security Considerations section of RFC 5280 as follows. The fifth paragraph of the Security Considerations section of RFC 5280 says:

> The protection afforded private keys is a critical security factor. On a small scale, failure of users to protect their private keys will permit an attacker to masquerade as them or decrypt their personal information. On a larger scale, compromise of a CA's private signing key may have a catastrophic effect. If an attacker obtains the private key unnoticed, the attacker may issue bogus certificates and CRLs. Existence of bogus certificates and CRLs will undermine confidence in the system. If such a compromise is detected, all certificates issued to the compromised CA MUST be revoked, preventing services between its users and users of other CAs. Rebuilding after such a compromise will be problematic, so CAs are advised to implement a combination of strong technical measures (e.g., tamper-resistant cryptographic modules) and appropriate management procedures (e.g., separation of duties) to avoid such an incident.

This paragraph is replaced with:

> The protection afforded private keys is a critical security factor. On a small scale, failure of users to protect their private keys will permit an attacker to masquerade as them or decrypt their personal information. On a larger scale, compromise of a CA's private signing key may have a catastrophic effect.
>
> If an attacker obtains the private key of a CA unnoticed, the attacker may issue bogus certificates and CRLs. Even if an attacker is unable to obtain a copy of a CA's private key, the attacker may be able to issue bogus certificates and CRLs by making unauthorized use of the CA's workstation or of an RA's workstation. Such an attack may be the result of an attacker obtaining unauthorized access to the workstation, either locally or remotely, or may be the result of inappropriate activity by an insider. Existence of bogus certificates and CRLs will undermine confidence in the system. Among many other possible attacks, the attacker may issue bogus certificates that have the same subject names as legitimate certificates in order impersonate legitimate certificate subjects. This could include bogus CA certificates in which the subject names in the bogus certificates match the names under which legitimate CAs issue certificates and CRLs. This would allow the attacker to issue bogus certificates and CRLs that have the same issuer names, and possibly the same serial numbers, as certificates and CRLs issued by legitimate CAs.

7. IANA Considerations

This document has no actions for IANA.

8. References

8.1. Normative References

[RFC1123] Braden, R., Ed., "Requirements for Internet Hosts - Application and Support", STD 3, RFC 1123, October 1989.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.

8.2. Informative References

[RFC5937] Ashmore, S. and C. Wallace, "Using Trust Anchor Constraints during Certification Path Processing", RFC 5937, August 2010.

[X.680] ITU-T Recommendation X.680 (2002) | ISO/IEC 8824-1:2002, Information Technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation.

[NFC] Davis, M. and M. Duerst, "Unicode Standard Annex #15: Unicode Normalization Forms", October 2006, <http://www.unicode.org/reports/tr15/>.

Author's Address

David Cooper
National Institute of Standards and Technology
100 Bureau Drive, Mail Stop 8930
Gaithersburg, MD 20899-8930
USA

EMail: david.cooper@nist.gov