

INTERNET-DRAFT
Intended Status: Standards Track
Updates: RFC [5280](#) (if approved)
Expires: 27 November 2016

R. Housley
Vigil Security
26 May 2016

Extended Key Usage Constraints
draft-housley-spasm-eku-constraints-03

Abstract

This document specifies the extended key usage constraints certificate extension, which is used to place restrictions on the key purpose identifiers that are authorized to appear in the end-entity certificate in a certification path. Restrictions apply to the extended key usage certificate extension, which is described in [RFC 5280](#).

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/lid-abstracts.html>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>

Copyright and License Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

INTERNET-DRAFT

EKU Constraints

26 May 2016

This document is subject to [BCP 78](http://trustee.ietf.org/license-info) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

[1](#) Introduction

This document specifies the extended key usage constraints certificate extension, which is used to place restrictions on the key purpose identifiers that are authorized to appear in subsequent certificates in a certification path. Restrictions apply to the extended key usage certificate extension, which is described in [Section 4.2.1.12 of RFC 5280](#) [[RFC5280](#)].

[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](#)].

[1.2](#). ASN.1

Certificates are generated using ASN.1 [[X680](#)] and the Distinguished Encoding Rules (DER) [[X690](#)].

[RFC 5280](#) [[RFC5280](#)] contains two ASN.1 modules that make use of an older version of the syntax (the 1988 Syntax). [RFC 5912](#) [[RFC5912](#)] provides these same ASN.1 modules in the newer syntax. The appendix of this document provides an ASN.1 module; it employs the newer syntax.

[2](#). Extended Key Usage Constraints Certificate Extension

The extended key usage (EKU) constraints certificate extension, which MUST be used only in a CA certificate, indicates the extended key usage values that are authorized to appear in subsequent certificates in a certification path. Restrictions apply to the extended key usage certificate extension, which is described in [Section 4.2.1.12](#)

Restrictions are defined in terms of permitted or excluded key purpose identifiers.

The permitted key purpose identifiers begins with the universal set. Then, as each certificate in the certification path is processed, the permitted key purpose identifiers are reduced to the intersection of the previous set and the ones listed in the `permittedKeyPurposeIds` field. Finally, each key purpose identifier in the extended key usage extension of the end-entity certificate **MUST** appear in the permitted key purpose identifiers set. The `permittedKeyPurposeIds` field **MUST NOT** be an empty sequence.

The excluded key purpose identifiers begins with the empty set. Then, as each certificate in the certification path is processed, the excluded key purpose identifiers are increased to the union of the previous set and the ones listed in the `excludedKeyPurposeIds` field. Finally, each key purpose identifier in the extended key usage extension of the end-entity certificate **MUST NOT** appear in the excluded key purpose identifiers set. The `excludedKeyPurposeIds` field **MUST NOT** be an empty sequence.

The special key purpose identifier `anyExtendedKeyUsage` is not treated differently than any other key purpose identifier in processing the constraints. If the `anyExtendedKeyUsage` key purpose identifier appears in the extended key usage extension of the end-entity certificate, then the `anyExtendedKeyUsage` key purpose identifier **MUST** appear in the permitted key purpose identifiers set and the `anyExtendedKeyUsage` key purpose identifier **MUST NOT** appear in the excluded key purpose identifiers set.

This extension **MAY**, at the option of the certificate issuer, be either critical or non-critical.

Conforming applications **MUST** be able to process this extension. If any CA certificate in the certification path includes an extended key usage constraints extension and the end-entity certificate includes an extended key usage certificate extension, then the application **MUST** either process the extended key usage extension constraint or reject the certificate.

```

ext-ExtKeyUsageConstraints EXTENSION ::= {
    SYNTAX EKUConstraints
    IDENTIFIED BY id-ce-ekuConstraints }

id-ce-ekuConstraints OBJECT IDENTIFIER ::= { id-pe TBD }

EKUConstraints ::= CHOICE {
    permittedKeyPurposeIds    [0] KeyPurposeIds,
    excludedKeyPurposeIds     [1] KeyPurposeIds }

KeyPurposeIds ::= SEQUENCE SIZE (1..MAX) OF KeyPurposeId

```

[3.](#) Basic Path Validation

Certification path validation is described in [Section 6.1 of RFC 5280 \[RFC5280\]](#). Certification path processing verifies the binding between the subject name and the subject public key. The binding is limited by constraints that are specified in the certificates that comprise the path and inputs that are specified by the relying party. Certification path processing requires the name and public key for a trust anchor.

This section extends certification path processing to include EKU constraints.

The resulting certification path validation processing is compatible with the trust anchor constraints processing described in [RFC 5937 \[RFC5937\]](#).

[3.1.](#) Inputs

No additional inputs are needed.

[3.2.](#) Initialization

Two additional values are initialized.

- (1) `permitted_key_purpose_ids`: a set of key purpose identifiers; all of the key purpose identifiers in the end-entity certificate MUST be included in this set. If the set is empty, then the certification path will be considered invalid if the end-entity

certificate includes an extended key usage extension. The initial value is the special value that represents the universal set.

- (m) `excluded_key_purpose_ids`: a set of key purpose identifiers; the key purpose identifiers in the end-entity certificate **MUST NOT** be included in this set. If the set is empty, then no key purpose identifiers are excluded. The initial value is the empty set.

[3.3.](#) Basic Certificate Processing

No additional processing steps are needed.

[3.4.](#) Preparation for Certificate *i+1*

One additional processing step is needed.

- (p) If a EKU constraints extension is included in the certificate, then modify the `permitted_key_purpose_ids` and `excluded_key_purpose_ids` state variables as follows:
 - (1) If `permittedKeyPurposeIds` is present in the certificate, set the `permitted_key_purpose_ids` state variable to the intersection of its previous value and the value indicated in the extension field.
 - (2) If `excludedKeyPurposeIds` is present in the certificate, set the `excluded_key_purpose_ids` state variable to the union of its previous value and the value indicated in the extension field.

[3.5.](#) Wrap-Up Procedure

Two additional processing steps are needed.

- (h) If the EKU extension is included in the end-entity certificate, then confirm that the values meet the restrictions in the `permitted_key_purpose_ids` and `excluded_key_purpose_ids` state variables as follows:
- (1) If `permitted_key_purpose_ids` state variable is empty, then return a failure indication and an appropriate reason.
 - (2) If `excluded_key_purpose_ids` state variable is not empty, then confirm that none of the key purpose identifiers in the state variable are present in the end-entity certificate. If any are present, then return a failure indication and an appropriate reason.
 - (3) If `permitted_key_purpose_ids` state variable is not the special value that represents the universal set, then confirm that all of the key purpose identifiers in the end-entity certificate are present in the state variable. If any are missing, then return a failure indication and an appropriate reason.

- (i) If the EKU extension is not present in the end-entity certificate, then confirm that the `permitted_key_purpose_ids` state variable is the special value that represents the universal set and the `excluded_key_purpose_ids` state variable is the empty set. Otherwise, return a failure indication and an appropriate reason.

[3.6.](#) Outputs

No additional output values are returned.

[4.](#) IANA Considerations

Please assign an object identifier for the certificate extension specified in this document.

Please assign an object identifier for the ASN.1 module in the Appendix.

5. Security Considerations

When a CA includes the extended key usage constraints certificate extension marked as non-critical, a relying party that does not understand this extension will ignore it. As a result, the relying party might accept some key purpose identifiers in the end-entity certificate that would have been unauthorized. If it would be preferable for the certification path to be rejected, then the CA SHOULD mark the extended key usage constraints certificate extension as critical.

When a CA includes the extended key usage constraints certificate extension for a subordinate CA, the OCSPSigning key purpose identifier SHOULD be included in the permittedKeyPurposeIds field to enable the issuance of delegated OCSP Responder certificates.

6. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [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](#), DOI 10.17487/RFC5280, May 2008, <<http://www.rfc-editor.org/info/rfc5280>>.

- [X680] ITU-T, "Information technology -- Abstract Syntax Notation One (ASN.1): Specification of basic notation", ITU-T Recommendation X.680, 2002.
- [X690] ITU-T, "Information technology -- ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", ITU-T Recommendation X.690, 2002.

7. Informative References

- [RFC5912] Hoffman, P. and J. Schaad, "New ASN.1 Modules for the Public Key Infrastructure Using X.509 (PKIX)", [RFC 5912](#), DOI 10.17487/RFC5912, June 2010, <<http://www.rfc-editor.org/info/rfc5912>>.
- [RFC5937] Ashmore, S. and C. Wallace, "Using Trust Anchor Constraints during Certification Path Processing", [RFC 5937](#), DOI 10.17487/RFC5937, August 2010, <<http://www.rfc-editor.org/info/rfc5937>>.


```

EKUConstraints2016 { iso(1) identified-organization(3) dod(6)
    internet(1) security(5) mechanisms(5) pkix(7) id-mod(0)
    id-mod-ekuConstraints2016(TBD) }

DEFINITIONS IMPLICIT TAGS ::= BEGIN

-- EXPORTS ALL --

IMPORTS
    EXTENSION
    FROM PKIX-CommonTypes-2009
        { iso(1) identified-organization(3) dod(6) internet(1)
          security(5) mechanisms(5) pkix(7) id-mod(0)
          id-mod-pkixCommon-02(57) }

    id-pe
    FROM PKIX1Explicit-2009
        { iso(1) identified-organization(3) dod(6) internet(1)
          security(5) mechanisms(5) pkix(7) id-mod(0)
          id-mod-pkix1-explicit-02(51) }

    KeyPurposeId
    FROM PKIX1Implicit-2009
        { iso(1) identified-organization(3) dod(6) internet(1)
          security(5) mechanisms(5) pkix(7) id-mod(0)
          id-mod-pkix1-implicit-02(59) } ;

MoreCertExtensions EXTENSION ::= {
    ext-ExtKeyUsageConstraints, ... }

ext-ExtKeyUsageConstraints EXTENSION ::= {
    SYNTAX EKUConstraints
    IDENTIFIED BY id-ce-ekuConstraints }

id-ce-ekuConstraints OBJECT IDENTIFIER ::= { id-pe TBD }

EKUConstraints ::= CHOICE {
    permittedKeyPurposeIds    [0] KeyPurposeIds,
    excludedKeyPurposeIds     [1] KeyPurposeIds }

KeyPurposeIds ::= SEQUENCE SIZE (1..MAX) OF KeyPurposeId

END

```

Acknowledgements

Many thanks to review and insightful comments from Santosh Chokhani, Stephen Farrell, Tom Gindin, Sean Leonard, Michael Richardson, Stefan Santesson, Jim Schaad, and Mike St.Johns.

Author's Address

Russell Housley
Vigil Security, LLC
918 Spring Knoll Drive
Herndon, VA 20170
USA
EMail: housley@vigilsec.com

