Workgroup: Network Working Group
Internet-Draft:
draft-becker-guthrie-cert-binding-for-multi-
auth-00
Published: 21 March 2022
Intended Status: Standards Track
Expires: 22 September 2022
Authors: A. Becker R. Guthrie M. Jenkins
NSA NSA NSA

Binding Certificates for Multiple Authentications within a Protocol

Abstract

This document defines a new CSR attribute, bindingRequest, and a new X.509 certificate extension, BoundCertificates. The use of the bindingRequest attribute in a CSR and the inclusion of the BoundCertificates extension in the resulting certificate together provide additional assurance that multiple certificates each belong to the same end entity. This mechanism is particularly useful in the context of non-composite hybrid authentication, which enables users to employ the same certificates in hybrid authentication as in authentication done with only traditional or PQ algorithms.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on 22 September 2022.

Copyright Notice

Table of Contents

1. Introduction

The goal of this document is to define a method for binding together multiple X.509 (aka PKIX) end-entity certificates in order to perform multiple authentications, where each certificate corresponds to a distinct digital signature, while minimizing changes to the certificate format [RFC5280] and to current PKI best practices.

When using non-composite hybrid public-key mechanisms, the party relying on a certificate (an authentication verifier or a key-establishment initiator) will want assurance that the private keys associated with each certificate are under the control of the same entity. This document defines a certificate extension, BoundCertificates, that signals that the certificate containing the extension is able to be used in combination with other certificate(s).

A certification authority (CA) that is asked to issue a certificate with such an extension may want assurance from a registration authority (RA) that the private keys (for example, corresponding to two public keys - one in an extant certificate, and one in a current request) belong to the same entity. To facilitate this, a CSR attribute is defined, called bindingRequest, that permits an RA to make such an attestation.

2. Requirements Language

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 [RFC2119]

3. Binding CSR to Certificates

3.1. The bindingRequest Attribute

This section defines a CSR attribute [RFC2986] designed to allow the RA to attest that the owner of the public key on the CSR also owns the public key associated with each end-entity certificate identified in this attribute. The bindingRequest attribute indicates previously-issued certificate(s) that the requesting entity owns and wants linked to the new certificate requested through the CSR.

The bindingRequests attribute has the following syntax:

```
bindingRequest ATTRIBUTE ::= {
    WITH SYNTAX BindingInfo
    ID {TBD?}
}

BindingInfo ::= SEQUENCE OF RequesterCertificate

RequesterCertificate ::= SEQUENCE {
     certID    IssuerAndSerialNumber
     signature BIT STRING
}
```

The RequesterCertificate type uses IssuerAndSerialNumber [RFC5652], repeated here for convenience.

```
IssuerAndSerialNumber ::= SEQUENCE {
    issuer Name,
    serialNumber CertificateSerialNumber }

    CertificateSerialNumber ::= INTEGER
```

The BindingInfo type is defined as a SEQUENCE OF RequesterCertificate type, which is a SEQUENCE of IssuerAndSerialNumber and signature.

BindingInfo includes a RequesterCertificate type for each certificate that the requesting entity would like linked to the CSR.

The RequesterCertificate type has two fields:

- The IssuerAndSerialNumber field identifies an end-entity certificate that the requesting entity would like linked to the CSR.

- The signature field provides evidence that the requesting entity owns this certificate. Specifically, the signature field contains a digital signature over IssuerAndSerialNumber, using the signature algorithm and private key associated with the certificate identified by the IssuerAndSerialNumber field.

The validation of this signature by the CA ensures that the owner of the CSR also owns the certificate(s) indicated in the bindingRequest attribute.

3.2. CSR Processing

If a CA receives a CSR containing the bindingRequest attribute, the CA:

- MUST verify the signature field(s) of the attribute. The CA validates the signature(s) using the public key associated with the certificate identified by the corresponding IssuerAndSerialNumber field. The details of the validation process are outside the scope of this document.

- SHOULD issue the certificate containing a BoundCertificates extension as specified in [Section 4], which references the associated certificate(s) indicated in the attribute.

The RA should only allow previously issued certificate(s) to be indicated in the bindingRequest attribute, to enable the CA to perform the signature verification described above.

It is not required that the requesting entity only include certificates in the bindingRequests attribute that were issued by the CA the CSR is being submitted to.

4. Binding Certificates

4.1. The BoundCertificates Extension

This section profiles a new X.509v3 certificate extension, BoundCertificates. BoundCertificates creates an association between the certificate containing the BoundCertificates extension and the certificate(s) referenced in the extension. When multiple end-entity certificates are used in a protocol, where one of the certificates contains a BoundCertificates extension that references another certificate(s), the authenticating entity is provided with

additional assurance that all certificates belong to the same entity.

The BoundCertificates extension is a list of entries, where each entry contains data that uniquely identifies a distinct end-entity certificate.

The BoundCertificates extension has the following syntax:

```
--  Object Identifiers for certificate extension
  id-boundCertificates OBJECT IDENTIFIER ::= { TBD }

--  X.509 Certificate extension
  BoundCertificates ::= SEQUENCE OF CertHash
                         -- hash of IssuerAndSerialNumber

--  Data types
  CertHash ::= SEQUENCE {
       hashAlgorithm  AlgorithmIdentifier,
       hashValue      OCTET STRING }
```

The CertHash hashValue is the digest value obtained from hashing the DER-encoded IssuerAndSerialNumber type from [RFC5652, Section 10.2.4] with the hash function identified in hashAlgorithm. This type is repeated here for convenience:

```
IssuerAndSerialNumber ::= SEQUENCE {
    issuer Name,
    serialNumber CertificateSerialNumber }

    CertificateSerialNumber ::= INTEGER
```

This extension SHOULD NOT be marked critical. Marking this extension critical would severely impact interoperability.

For certificate chains, this extension MUST only be included in the end-entity certificate.

For the BoundCertificates extension to be meaningful, a CA that issues a certificate with this extension:

- MUST only include CertHash types for certificates that were listed and validated in the bindingRequest attribute of the CSR submitted by the requesting entity.

- MUST ensure that all certificates are intended for the same use case.

*SHOULD determine that all certificates are valid at the time of issuance. The usable overlap of validity periods is a Subscriber concern.

4.2. Endpoint Protocol Multiple Authentication Processing

When the preference to use a non-composite hybrid authentication mode is expressed by an endpoint through the protocol itself, the use of the BoundCertificates extension and its enforcement are left to the protocol's native authorization mechanism (along with other decisions endpoints make about whether to complete or drop a connection).

If an endpoint has indicated that it is capable of non-composite hybrid authentication, and receives the appropriate authentication data, it SHOULD check end-entity certificates for the BoundCertificates extension. If present in one certificate, it SHOULD:

*Use the hash algorithm given in the extension to compute the appropriate hash of the DER-encoded IssuerAndSerialNumber of the other end-entity certificate(s) received.

*Verify that the hash value matches a hashValue in the BoundCertificates extension.

It is outside the scope of this document how to proceed with authentication based on the outcome of this verification process.

5. Security Considerations

This document inherits security considerations identified in [RFC5280].

The mechanisms described in this document provide only a means to express that multiple certificates are related. They are intended for the interpretation of the recipient of the data in which they are embedded (i.e. a CSR or certificate). They do not by themselves effect any security function.

Authentication, unlike key establishment, is necessarily a one-way arrangement. That is, authentication is an assertion made by a claimant to a verifier. The means and strength of mechanism for authentication have to be to the satisfaction of the verifier. A system security designer needs to be aware of what authentication assurances are needed in various parts of the system and how to achieve that assurance. In a closed system (e.g. Company X distributing firmware to its own devices) the approach may be implicit. In an online protocol like IPsec where the peers are generally known, any mechanism selected from a pre-established set

may be sufficient. For more promiscuous online protocols, like TLS, the ability for the verifier to express what is possible and what is preferred - and to assess that it got what it needed - is important.

A certificate is an assertion of binding between an identity and a public key. However, that assertion is based on several other assurances - specifically, that the identity belongs to a particular physical entity, and that that physical entity has control over the private key corresponding to the public. For any hybrid approach, it is important that there be evidence that the same entity controls all private keys at time of use (to the verifier) and at time of registration (to the CA).

All hybrid implementations are vulnerable to a downgrade attack in which a malicious peer does not express support for PQ algorithms, resulting in an exchange that can only rely upon traditional algorithms for security.

6. IANA Considerations

This document defines an extension for use with X.509 certificates. IANA is requested to register an OID in the PKIX certificate extensions arc [RFC7299]:

```
id-boundCerts OBJECT IDENTIFIER ::= { id-pkix 1 tbd }
```

with this document as the Required Specification.

This document also defines a CSR attribute. IANA is requested to register an OID:

```
id-bindingRequest OBJECT IDENTIFIER ::= { tbd }
```

7. References

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

[RFC2986] Nystrom, M. and B. Kaliski, "PKCS #10: Certification Request Syntax Specification Version 1.7", RFC 2986, DOI 10.17487/RFC2986, November 2000, <https://www.rfc-editor.org/info/rfc2986>.

[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, <https://www.rfc-editor.org/info/rfc5280>.

[RFC5652] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, RFC 5652, DOI 10.17487/RFC5652, September 2009, <https://www.rfc-editor.org/info/rfc5652>.

[RFC7299] Housley, R., "Object Identifier Registry for the PKIX Working Group", RFC 7299, DOI 10.17487/RFC7299, July 2014, <https://www.rfc-editor.org/info/rfc7299>.

Authors' Addresses

Alison Becker
National Security Agency

Email: aebecke@uwe.nsa.gov

Rebecca Guthrie
National Security Agency

Email: rmguthr@uwe.nsa.gov

Michael Jenkins
National Security Agency

Email: mjjenki@cyber.nsa.gov