Internet Engineering Task Force (IETF)
Request for Comments: 8002
Obsoletes: 6253
Updates: 7401
Category: Standards Track
ISSN: 2070-1721

T. Heer
Albstadt-Sigmaringen University
S. Varjonen
University of Helsinki

October 2016

Host Identity Protocol Certificates

Abstract

The Certificate (CERT) parameter is a container for digital certificates. It is used for carrying these certificates in Host Identity Protocol (HIP) control packets. This document specifies the certificate parameter and the error signaling in case of a failed verification. Additionally, this document specifies the representations of Host Identity Tags (HITs) in X.509 version 3 (v3).

The concrete use cases of certificates, including how certificates are obtained and requested and which actions are taken upon successful or failed verification, are specific to the scenario in which the certificates are used. Hence, the definition of these scenario-specific aspects is left to the documents that use the CERT parameter.

This document updates RFC 7401 and obsoletes RFC 6253.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc8002.

Table of Contents

1. Introduction

Digital certificates bind pieces of information to a public key by means of a digital signature and thus enable the holder of a private key to generate cryptographically verifiable statements. The Host Identity Protocol (HIP) [RFC7401] defines a new cryptographic namespace based on asymmetric cryptography. The identity of each host is derived from a public key, allowing hosts to digitally sign data and issue certificates with their private key. This document specifies the CERT parameter, which is used to transmit digital certificates in HIP. It fills the placeholder specified in Section 5.2 of [RFC7401] and thus updates [RFC7401].

1.1. Requirements Language

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

2. CERT Parameter

The CERT parameter is a container for certain types of digital certificates. It does not specify any certificate semantics. However, it defines supplementary parameters that help HIP hosts to transmit semantically grouped CERT parameters in a more systematic way. The specific use of the CERT parameter for different use cases is intentionally not discussed in this document. Hence, the use of the CERT parameter will be defined in the documents that use the CERT parameter.

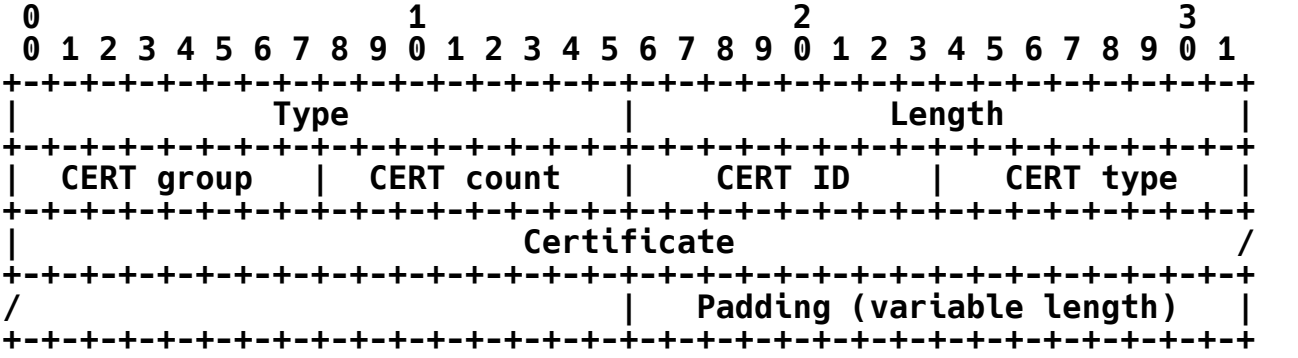
The CERT parameter is covered and protected, when present, by the HIP SIGNATURE field and is a non-critical parameter.

The CERT parameter can be used in all HIP packets. However, using it in the first Initiator (I1) packet is NOT RECOMMENDED because it can increase the processing times of I1s, which can be problematic when processing storms of I1s. Each HIP control packet MAY contain multiple CERT parameters, each carrying one certificate. These parameters MAY be related or unrelated. Related certificates are managed in CERT groups. A CERT group specifies a group of related CERT parameters that SHOULD be interpreted in a certain order (e.g., for expressing certificate chains). Ungrouped certificates exhibit a unique CERT group field and set the CERT count to 1. CERT parameters with the same group number in the CERT group field indicate a logical grouping. The CERT count field indicates the number of CERT parameters in the group.

CERT parameters that belong to the same CERT group MAY be contained in multiple sequential HIP control packets. This is indicated by a higher CERT count than the amount of CERT parameters with matching CERT group fields in a HIP control packet. The CERT parameters MUST be placed in ascending order, within a HIP control packet, according to their CERT group field. CERT groups MAY only span multiple packets if the CERT group does not fit the packet. A HIP packet MUST NOT contain more than one incomplete CERT group that continues in the next HIP control packet.

The CERT ID acts as a sequence number to identify the certificates in a CERT group. The numbers in the CERT ID field MUST start from 1 up to CERT count.

The CERT group and CERT ID namespaces are managed locally by each host that sends CERT parameters in HIP control packets.



Type	768
Length	Length in octets, excluding Type, Length, and Padding.
CERT group	Group ID grouping multiple related CERT parameters.
CERT count	Total count of certificates that are sent, possibly in several consecutive HIP control packets.
CERT ID	The sequence number for this certificate.
CERT Type	Indicates the type of the certificate.
Padding	Any Padding, if necessary, to make the TLV a multiple of 8 bytes. Any added padding bytes MUST be zeroed by the sender, and their values SHOULD NOT be checked by the receiver.

The certificates MUST use the algorithms defined in [RFC7401] as the signature and hash algorithms.

The following certificate types are defined:

CERT format	Type number
Reserved	0
X.509 v3	1
Obsoleted	2
Hash and URL of X.509 v3	3
Obsoleted	4
LDAP URL of X.509 v3	5
Obsoleted	6
Distinguished Name of X.509 v3	7
Obsoleted	8

The next sections outline the use of HITs in X.509 v3. X.509 v3 certificates and the handling procedures are defined in [RFC5280]. The wire format for X.509 v3 is the Distinguished Encoding Rules format as defined in [X.690].

Hash and Uniform Resource Locator (URL) encoding (3) is used as defined in Section 3.6 of [RFC7296]. Using hash and URL encodings result in smaller HIP control packets than by including the certificate(s) but requires the receiver to resolve the URL or check a local cache against the hash.

Lightweight Directory Access Protocol (LDAP) URL encoding (5) is used as defined in [RFC4516]. Using LDAP URL encoding results in smaller HIP control packets but requires the receiver to retrieve the certificate or check a local cache against the URL.

Distinguished Name (DN) encoding (7) is represented by the string representation of the certificate's subject DN as defined in [RFC4514]. Using the DN encoding results in smaller HIP control packets but requires the receiver to retrieve the certificate or check a local cache against the DN.

3. X.509 v3 Certificate Object and Host Identities

If needed, HITs can represent an issuer, a subject, or both in X.509 v3. HITs are represented as IPv6 addresses as defined in [RFC7343]. When the Host Identifier (HI) is used to sign the certificate, the respective HIT SHOULD be placed into the Issuer Alternative Name (IAN) extension using the GeneralName form iPAddress as defined in [RFC5280]. When the certificate is issued for a HIP host, identified by a HIT and an HI, the respective HIT SHOULD be placed into the

Subject Alternative Name (SAN) extension using the GeneralName form iPAddress, and the full HI is presented as the subject's public key info as defined in [RFC5280].

The following examples illustrate how HITs are presented as the issuer and subject in the X.509 v3 extension alternative names.

```
Format of X509v3 extensions:
    X509v3 Issuer Alternative Name:
        IP Address:hit-of-issuer
    X509v3 Subject Alternative Name:
        IP Address:hit-of-subject

Example X509v3 extensions:
    X509v3 Issuer Alternative Name:
        IP Address:2001:24:6cf:fae7:bb79:bf78:7d64:c056
    X509v3 Subject Alternative Name:
        IP Address:2001:2c:5a14:26de:a07c:385b:de35:60e3
```

Appendix A shows a full example X.509 v3 certificate with HIP content.

As another example, consider a managed Public Key Infrastructure (PKI) environment in which the peers have certificates that are anchored in (potentially different) managed trust chains. In this scenario, the certificates issued to HIP hosts are signed by intermediate Certification Authorities (CAs) up to a root CA. In this example, the managed PKI environment is neither HIP aware nor can it be configured to compute HITs and include them in the certificates.

When HIP communications are established, the HIP hosts not only need to send their identity certificates (or pointers to their certificates) but also the chain of intermediate CAs (or pointers to the CAs) up to the root CA, or to a CA that is trusted by the remote peer. This chain of certificates SHOULD be sent in a CERT group as specified in Section 2. The HIP peers validate each other's certificates and compute peer HITs based on the certificate public keys.

4. Revocation of Certificates

Revocation of X.509 v3 certificates is handled as defined in Section 5 of [RFC5280] with two exceptions. First, any HIP certificate serial number that appears on the Certificate Revocation List (CRL) is treated as invalid regardless of the reason code. Second, the certificateHold is not supported.

5. Error Signaling

If the Initiator does not send all the certificates that the Responder requires, the Responder may take actions (e.g., reject the connection). The Responder MAY signal this to the Initiator by sending a HIP NOTIFY message with NOTIFICATION parameter error type CREDENTIALS_REQUIRED.

If the verification of a certificate fails, a verifier MAY signal this to the provider of the certificate by sending a HIP NOTIFY message with NOTIFICATION parameter error type INVALID_CERTIFICATE.

NOTIFICATION PARAMETER - ERROR TYPES	Value
CREDENTIALS_REQUIRED	48
The Responder is unwilling to set up an association, as the Initiator did not send the needed credentials.	
INVALID_CERTIFICATE	50
Sent in response to a failed verification of a certificate. Notification Data MAY contain a CERT group and CERT ID octet (in this order) of the CERT parameter that caused the failure.	

6. IANA Considerations

This document defines the CERT parameter for HIP [RFC7401]. The CERT parameter type number (768) is defined in [RFC7401].

The CERT parameter has an 8-bit unsigned integer field for different certificate types, for which IANA has created and maintains a subregistry entitled "HIP Certificate Types" under "Host Identity Protocol (HIP) Parameters". Values for the "HIP Certificate Types" registry are given in Section 2. New values for the Certificate types from the unassigned space are assigned through IETF Review.

In Section 5, this document defines two types for the "NOTIFY Message Types" subregistry under "Host Identity Protocol (HIP) Parameters".

As this document obsoletes [RFC6253], references to [RFC6253] in IANA registries have been replaced by references to this document. This document changes the "HIP Certificate Types" registry in Section 2.

The following updates to the "HIP Certificate Types" registry have been made.

The references have been updated from [RFC6253] to this document.

This document obsoleted the type numbers "2", "4", "6", and "8" for the Simple Public Key Infrastructure (SPKI) certificates.

7. Security Considerations

Certificate grouping allows the certificates to be sent in multiple consecutive packets. This might allow similar attacks, as IP-layer fragmentation allows, for example, the sending of fragments in the wrong order and skipping some fragments to delay or stall packet processing by the victim in order to use resources (e.g., CPU or memory). Hence, hosts SHOULD implement mechanisms to discard certificate groups with outstanding certificates if state space is scarce.

Although the CERT parameter is allowed in the I1 packet, it is NOT RECOMMENDED because it can increase the processing times of I1s, which can be problematic when processing storms of I1s. Furthermore, the Initiator has to take into consideration that the Responder can drop the CERT parameter in I1 without processing the parameter.

Checking of the URL and LDAP entries might allow denial-of-service (DoS) attacks, where the target host may be subjected to bogus work.

Security considerations for X.509 v3 are discussed in [RFC5280].

8. Differences from RFC 6253

This section summarizes the technical changes made from [RFC6253]. This section is informational and is intended to help implementors of the previous protocol version. If any text in this section contradicts text in other portions of this specification, the text found outside of this section should be considered normative.

The following change has been made.

- Support for SPKI certificates has been removed.

9. References

9.1. 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>.

[RFC4514] Zeilenga, K., Ed., "Lightweight Directory Access Protocol (LDAP): String Representation of Distinguished Names", RFC 4514, DOI 10.17487/RFC4514, June 2006, <http://www.rfc-editor.org/info/rfc4514>.

[RFC4516] Smith, M., Ed. and T. Howes, "Lightweight Directory Access Protocol (LDAP): Uniform Resource Locator", RFC 4516, DOI 10.17487/RFC4516, June 2006, <http://www.rfc-editor.org/info/rfc4516>.

[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>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <http://www.rfc-editor.org/info/rfc7296>.

[RFC7343] Laganier, J. and F. Dupont, "An IPv6 Prefix for Overlay Routable Cryptographic Hash Identifiers Version 2 (ORCHIDv2)", RFC 7343, DOI 10.17487/RFC7343, September 2014, <http://www.rfc-editor.org/info/rfc7343>.

[RFC7401] Moskowitz, R., Ed., Heer, T., Jokela, P., and T. Henderson, "Host Identity Protocol Version 2 (HIPv2)", RFC 7401, DOI 10.17487/RFC7401, April 2015, <http://www.rfc-editor.org/info/rfc7401>.

[X.690] 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 | ISO/IEC 8825-1, August 2015.

9.2. Informative References

[RFC6253] Heer, T. and S. Varjonen, "Host Identity Protocol Certificates", RFC 6253, DOI 10.17487/RFC6253, May 2011, <http://www.rfc-editor.org/info/rfc6253>.

Appendix A. X.509 v3 Certificate Example

This section shows an X.509 v3 certificate with encoded HITs.

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 12705268244493839545 (0xb0522e27291b2cb9)
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: DC=Example, DC=com, CN=Example issuing host
        Validity
            Not Before: Feb 25 11:28:29 2016 GMT
            Not After : Feb 24 11:28:29 2017 GMT
        Subject: DC=Example, DC=com, CN=Example issuing host
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:c9:b0:85:94:af:1f:3a:77:39:c9:d5:81:a5:ee:
                    d2:b5:6b:72:91:5d:22:2c:1e:59:e5:06:29:bd:a2:
                    19:f6:ac:ca:eb:f7:88:d8:54:55:41:01:58:d8:87:
                    64:d8:c8:cf:6e:c2:38:81:22:1a:ae:e9:a6:80:22:
                    03:ee:f3:1b:7e:68:11:e3:f4:7b:98:33:28:bf:40:
                    ec:4f:19:e8:10:8a:8b:07:60:f7:9f:e4:82:f8:a7:
                    58:04:3d:42:07:c8:34:ca:99:6d:11:eb:73:c1:d9:
                    96:93:55:e5:c7:ed:80:4f:8a:f2:1a:6f:83:c8:15:
                    a4:8f:b8:6a:fe:f3:4f:49:1a:5c:1f:89:bb:30:e6:
                    98:bc:ce:a3:a2:37:85:b1:79:1c:26:e6:44:0c:b9:
                    3e:d8:37:81:46:f4:02:25:46:a2:ea:da:25:5c:46:
                    a2:a3:c5:58:80:53:1f:c5:e5:11:a0:da:d8:f2:ad:
                    d6:98:d4:ce:55:35:cc:0b:d3:5b:09:48:ef:57:65:
                    80:cb:65:79:fd:cb:4d:5b:b3:8d:1a:ff:2a:58:3e:
                    96:65:10:3e:04:81:78:2b:d5:ca:89:78:ea:28:5c:
                    bc:02:4a:54:cd:aa:a9:99:8d:d6:39:e9:5e:a9:73:
                    1a:5d:93:55:39:9b:72:1a:c2:a0:1f:e3:4c:b0:41:
                    98:97
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Subject Alternative Name:
                IP Address:2001:27:DCFC:CB8:F885:D53F:4E63:48B7
            X509v3 Issuer Alternative Name:
                IP Address:2001:2D:F878:64C1:67E3:9716:88BD:68E4
```

```
    Signature Algorithm: sha256WithRSAEncryption
         6d:e6:a9:a6:30:c4:ab:3e:86:39:1e:de:76:4d:4e:a4:2d:63:
         4d:bb:41:bf:d3:0c:66:13:8b:4d:b2:50:59:36:fc:ae:42:9e:
         c8:a0:41:1a:1c:94:56:05:28:82:34:4e:63:75:87:31:25:67:
         36:a6:1a:0f:b8:f7:db:03:e7:dd:a6:9a:26:c4:68:e2:cf:59:
         54:e6:ee:cc:a7:ce:fb:56:bf:31:60:f4:cb:e7:f0:0e:50:f8:
         b7:c5:3c:1a:de:74:d0:aa:83:e5:15:25:b1:bf:be:a4:7f:af:
         0a:de:08:09:0e:13:1d:2a:3b:1a:99:d9:af:10:fc:08:92:5f:
         d8:d0:10:d6:b9:0c:86:da:85:3b:44:b5:97:90:10:02:4f:5a:
         1f:ae:07:30:6b:f5:e6:12:93:72:e2:10:c9:8e:2c:00:8b:d6:
         f0:05:c3:ff:91:24:69:6d:5b:5a:0c:40:28:01:f2:5b:45:b8:
         9b:ae:9e:73:e9:dd:83:e0:85:d7:ad:6c:b1:81:ac:a0:30:37:
         9d:60:bd:92:3b:d2:a1:21:87:8b:c4:d9:5a:5c:21:56:3e:02:
         7e:f3:6f:a5:de:40:75:80:f5:41:68:5c:b2:61:fb:1d:9a:a5:
         97:a8:d4:a9:82:45:86:79:3c:63:76:3d:fd:86:a0:f8:14:84:
         55:c1:8c:fa
```

```
-----BEGIN CERTIFICATE-----
MIIDWTCCAkGgAwIBAgIJALBSLicpGyy5MA0GCSqGSIb3DQEBCwUAME0xFzAVBgoJ
kiaJk/IsZAEZFgdFeGFtcGxlMRMwEQYKCZImiZPyLGQBGRYDY29tMR0wGwYDVQQD
ExRFeGFtcGxlIGlzc3VpbmcgaG9zdDAeFw0xNjAyMjUxMTI4MjlaFw0xNzAyMjQx
MTI4MjlaME0xFzAVBgoJkiaJk/IsZAEZFgdFeGFtcGxlMRMwEQYKCZImiZPyLGQB
GRYDY29tMR0wGwYDVQQDExRFeGFtcGxlIGlzc3VpbmcgaG9zdDCCASIwDQYJKoZI
hvcNAQEBBQADggEPADCCAQoCggEBAMmwhZSvHzp3OcnVgaXu0rVrcpFdIiweWeUG
Kb2iGfasyuv3iNhUVUEBWNiHZNjIz27COIEiGq7ppoAiA+7zG35oEeP0e5gzKL9A
7E8Z6BCKiwdg95/kgvinWAQ9QgfINMqZbRHrc8HZlpNV5cftgE+K8hpvg8gVpI+4
av7zT0kaXB+JuzDmmLzOo6I3hbF5HCbmRAy5Ptg3gUb0AiVGouraJVxGoqPFWIBT
H8XlEaDa2PKt1pjUzlU1zAvTWwlI71dlgMtlef3LTVuzjRr/Klg+lmUQPgSBeCvV
yol46ihcvAJKVM2qqZmN1jnpXqlzGl2TVTmbchrCoB/jTLBBmJcCAwEAAaM8MDow
GwYDVR0RBBQwEocQIAEAJ9z8DLj4hdU/TmNItzAbBgNVHRIEFDAShxAgAQAt+Hhk
wWfjlxaIvWjkMA0GCSqGSIb3DQEBCwUAA4IBAQBt5qmmMMSrPoY5Ht52TU6kLWNN
u0G/0wxmE4tNslBZNvyuQp7IoEEaHJRWBSiCNE5jdYcxJWc2phoPuPfbA+fdppom
xGjiz1lU5u7Mp877Vr8xYPTL5/AOUPi3xTwa3nTQqoPlFSWxv76kf68K3ggJDhMd
KjsamdmvEPwIkl/Y0BDWuQyG2oU7RLWXkBACT1ofrgcwa/XmEpNy4hDJjiwAi9bw
BcP/kSRpbVtaDEAoAfJbRbibrp5z6d2D4IXXrWyxgaygMDedYL2SO9KhIYeLxNla
XCFWPgJ+82+l3kB1gPVBaFyyYfsdmqWXqNSpgkWGeTxjdj39hqD4FIRVwYz6
-----END CERTIFICATE-----
```

Acknowledgments

The authors would like to thank A. Keranen, D. Mattes, M. Komu, and T. Henderson for the fruitful conversations on the subject. D. Mattes most notably contributed the non-HIP-aware use case in Section 3.

Authors' Addresses

Tobias Heer
Albstadt-Sigmaringen University
Poststr. 6
72458 Albstadt
Germany

Email: heer@hs-albsig.de

Samu Varjonen
University of Helsinki
Gustaf Haellstroemin katu 2b
00560 Helsinki
Finland

Email: samu.varjonen@helsinki.fi