

Host Identity Protocol
Internet-Draft
Updates: [5201](#) (if approved)
Intended status: Experimental
Expires: September 15, 2011

Heer
Communication and Distributed
Systems, RWTH Aachen University
Varjonen
Helsinki Institute for Information
Technology
March 14, 2011

Host Identity Protocol Certificates draft-ietf-hip-cert-12

Abstract

The 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 in X.509 version 3 (v3) and SPKI certificates.

The concrete use of certificates including how certificates are obtained, 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 are left to the documents that use the CERT parameter.

This document updates [RFC 5201](#).

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 <http://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 September 15, 2011.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) 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.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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) [[RFC5201](#)] 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 \[RFC5201\]](#), and thus, updates [[RFC5201](#)].

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 because it is specific to a concrete use case. Hence, the use of the CERT parameter will be defined in the documents that use the CERT parameter.

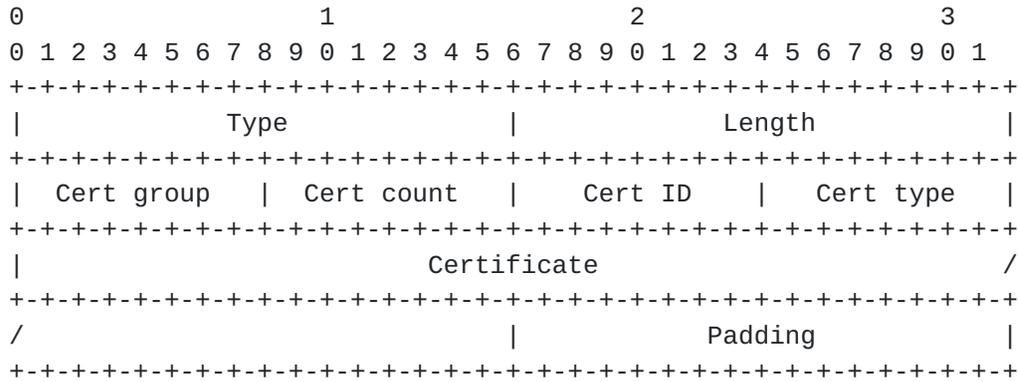
The CERT parameter is covered and protected, when present, by the HIP SIGNATURE field and is a non-critical parameter.

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. 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). For grouping CERT parameters, the Cert group and the Cert count field MUST be set. Ungrouped certificates exhibit a unique Cert group field and set the Cert count to 1. CERT parameters with the same Cert group number in the 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.

The certificates MUST use the algorithms defined in [[RFC5201](#)] as the signature and hash algorithms.

The following certificate types are defined:

```

+-----+-----+
|           Cert format           | Type number |
+-----+-----+
|           Reserved              |           0  |
|           X.509 v3              |           1  |
|           SPKI                  |           2  |
| Hash and URL of X.509 v3       |           3  |
| Hash and URL of SPKI          |           4  |
| LDAP URL of X.509 v3          |           5  |
| LDAP URL of SPKI              |           6  |
| Distinguished Name of X.509 v3 |           7  |
| Distinguished Name of SPKI    |           8  |
+-----+-----+
    
```

The next sections outline the use of Host Identity Tags (HITs) in X.509 v3 and in Simple Public Key Infrastructure (SPKI) certificates. X.509 v3 certificates and the handling procedures are defined in [[RFC5280](#)]. The wire format for X.509 v3 is Distinguished Encoding Rules format as defined in [[X.690](#)]. The SPKI, the handling procedures, and the formats are defined in [[RFC2693](#)].

Hash and Uniform Resource Locator (URL) encodings (3 and 4) are used as defined in [\[RFC5996\] Section 3.6](#). Using hash and URL encodings results 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.

LDAP URL encodings (5 and 6) are 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) encodings (7 and 8) are 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 [\[RFC4843\]](#). When Host Identifier (HI) is used to sign the certificate the respective HIT MUST be placed in to 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 HI, the respective HIT MUST be placed in to the Subject Alternative Name (SAN) extension using the GeneralName form `iPAddress` and the full HI is presented as the subjects public key info as defined in [\[RFC5280\]](#).

The following examples illustrate how HITs are presented as 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:2001:14:6cf:fae7:bb79:bf78:7d64:c056

X509v3 Subject Alternative Name:

IP Address:2001:1C:5a14:26de:a07C:385b:de35:60e3

[Appendix B](#) 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 MUST 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.](#) SPKI Cert Object and Host Identities

When using SPKI certificates to transmit information related to HIP hosts, HITs need to be enclosed within the certificates. HITs can represent an issuer, a subject, or both. In the following we define the representation of those identifiers for SPKI given as S-expressions. Note that the S-expressions are only the human-readable representation of SPKI certificates. Full HIs are presented in the public key sequences of SPKI certificates.

As an example the Host Identity Tag of a host is expressed as follows:

Format: (hash hit hit-of-host)

Example: (hash hit 2001:13:724d:f3c0:6ff0:33c2:15d8:5f50)

[Appendix A](#) shows a full example SPKI certificate with HIP content.

[5.](#) Revocation of Certificates

Revocation of X.509 v3 certificates is handled as defined in [Section 5 of \[RFC5280\]](#). Revocation of SPKI certificates is handled as defined in [Section 5 of \[RFC2693\]](#).

6. Error signaling

If the Initiator does not send the certificate 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_NEEDED.

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 n groups of 2 octets (n calculated from the NOTIFICATION parameter length), in order Cert group and Cert ID of the certificate parameter that caused the failure.

7. IANA Considerations

This document defines the CERT parameter for the Host Identity Protocol [[RFC5201](#)]. This parameter is defined in [Section 2](#) with type 768. The parameter type number is also defined in [[RFC5201](#)].

The CERT parameter has 8-bit unsigned integer field for different certificate types, for which IANA is to create and maintain a new sub-registry entitled "HIP certificate types" under the "Host Identity Protocol (HIP) Parameters". Initial values for the Certificate type registry are given in [Section 2](#). New values for the Certificate types from the unassigned space are assigned through IETF Review.

In [Section 6](#) this document defines two new types for "NOTIFY message types" sub-registry under "Host Identity Protocol (HIP) Parameters".

8. 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 sending of fragments in 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.

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

Security considerations for SPKI certificates are discussed in [[RFC2693](#)] and for X.509 v3 in [[RFC5280](#)]

9. Acknowledgements

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](#).

10. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2693] Ellison, C., Frantz, B., Lampson, B., Rivest, R., Thomas, B., and T. Ylonen, "SPKI Certificate Theory", [RFC 2693](#), September 1999.
- [RFC4514] Zeilenga, K., "Lightweight Directory Access Protocol (LDAP): String Representation of Distinguished Names", [RFC 4514](#), June 2006.
- [RFC4516] Smith, M. and T. Howes, "Lightweight Directory Access Protocol (LDAP): Uniform Resource Locator", [RFC 4516](#), June 2006.
- [RFC4843] Nikander, P., Laganier, J., and F. Dupont, "An IPv6 Prefix for Overlay Routable Cryptographic Hash Identifiers (ORCHID)", [RFC 4843](#), April 2007.
- [RFC5201] Moskowitz, R., Nikander, P., Jokela, P., and T. Henderson, "Host Identity Protocol", [RFC 5201](#), April 2008.

- [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.
- [RFC5996] Kaufman, C., Hoffman, P., Nir, Y., and P. Eronen, "Internet Key Exchange Protocol Version 2 (IKEv2)", [RFC 5996](#), September 2010.
- [X.690] ITU-T, "Recommendation X.690 (2002) | ISO/IEC 8825-1:2002, Information Technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", July 2002.

[Appendix A](#). SPKI certificate example

This section shows a SPKI certificate with encoded HITs. The example has been indented for readability.


```

(sequence
  (public_key
    (rsa-pkcs1-sha1
      (e #010001#)
      (n |yDwzn0wX0w+zvQbpWoTnfWrUPLKW2NFrpXbsIch/QBSLb
        k1RKTZhLasFwvtSHAjqh220W8gRiQAGIqKplyrDEqSrJp
        OdIsHIQ8BQhJAyILWA1Sa6f5wAnWozDfgdXoKLNdT8ZNB
        mzluPiw4ozc78p6MHElH75Hm3yHaWxT+s83M=|
      )
    )
  )
)
(cert
  (issuer
    (hash hit 2001:15:2453:698a:9aa:253a:dcb5:981e)
  )
  (subject
    (hash hit 2001:12:ccd6:4715:72a3:2ab1:77e4:4acc)
  )
  (not-before "2011-01-12_13:43:09")
  (not-after "2011-01-22_13:43:09")
)
(signature
  (hash sha1 |h5fC8HUMATTtK0cjYqIgeN3HCIMA|
    |u8NTRutINI/AeeZgN6bngjvjYPtVahvY7MhGfenTpT7MCgBy
    NoZglqH5Cy2vH6LrQFYwx0MjWoYwHKimEuBKCnd4TK6hrCyAI
    CIDJAZ70TyKXgONwDNWPOmcc3lFmsih8ezkoBseFWHqRGISIm
    MLdeaMciP4lVfxPY2AQKdMrBc=|
  )
)
)

```

[Appendix B.](#) X.509.v3 certificate example

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

Certificate:

Data:

```
Version: 3 (0x2)
Serial Number: 0 (0x0)
Signature Algorithm: sha1WithRSAEncryption
Issuer: CN=Example issuing host, DC=example, DC=com
Validity
  Not Before: Mar 11 09:01:39 2011 GMT
  Not After : Mar 21 09:01:39 2011 GMT
Subject: CN=Example subject host, DC=example, DC=com
Subject Public Key Info:
  Public Key Algorithm: rsaEncryption
  RSA Public Key: (1024 bit)
  Modulus (1024 bit):
    00:c0:db:38:50:8e:63:ed:96:ea:c6:c4:ec:a3:36:
    62:e2:28:e9:74:9c:f5:2f:cb:58:0e:52:54:60:b5:
    fa:98:87:0d:22:ab:d8:6a:61:74:a9:ee:0b:ae:cd:
    18:6f:05:ab:69:66:42:46:00:a2:c0:0c:3a:28:67:
    09:cc:52:27:da:79:3e:67:d7:d8:d0:7c:f1:a1:26:
    fa:38:8f:73:f5:b0:20:c6:f2:0b:7d:77:43:aa:c7:
    98:91:7e:1e:04:31:0d:ca:94:55:20:c4:4f:ba:b1:
    df:d4:61:9d:dd:b9:b5:47:94:6c:06:91:69:30:42:
    9c:0a:8b:e3:00:ce:49:ab:e3
  Exponent: 65537 (0x10001)
X509v3 extensions:
  X509v3 Issuer Alternative Name:
    IP Address:2001:13:8D83:41C5:DC9F:38ED:E742:7281
  X509v3 Subject Alternative Name:
    IP Address:2001:1C:6E02:D3E0:9B90:8417:673E:99DB
Signature Algorithm: sha1WithRSAEncryption
83:68:b4:38:63:a6:ae:57:68:e2:4d:73:5d:8f:11:e4:ba:30:
a0:19:ca:86:22:e9:6b:e9:36:96:af:95:bd:e8:02:b9:72:2f:
30:a2:62:ac:b2:fa:3d:25:c5:24:fd:8d:32:aa:01:4f:a5:8a:
f5:06:52:56:0a:86:55:39:2b:ee:7a:7b:46:14:d7:5d:15:82:
4d:74:06:ca:b7:8c:54:c1:6b:33:7f:77:82:d8:95:e1:05:ca:
e2:0d:22:1d:86:fc:1c:c4:a4:cf:c6:bc:ab:ec:b8:2a:1e:4b:
04:7e:49:9c:8f:9d:98:58:9c:63:c5:97:b5:41:94:f7:ef:93:
57:29
```

[Appendix C. Change log](#)

Changes from version 00 to 01:

- o Revised text on DN usage.
- o Revised text on Cert group usage.

Changes from version 01 to 02:

- o Revised the type numbers.
- o Added a section on signaling.

Changes from version 02 to 03:

- o Revised text on CERT usage in control packets.

Changes from version 03 to 04:

- o Added the non-HIP aware use case to the [Section 3](#).
- o Clarified that the HITs are not always required in the certificates.
- o Rewrote the signaling section.
- o LDAP URL to LDAP DN in [Section 2](#) last paragraph.
- o CERT is always covered by a signature as it's type number requires
- o New example certificates
- o Style and language clean-ups
- o Changed IANA considerations
- o Revised the type numbers
- o [RFC 2119](#) keywords
- o Updated the IANA considerations section
- o Rewrote the abstract

Changes from version 04 to 05:

- o Clarified the examples in [Section 3](#).
- o Clarifications to Section [Section 3](#).
- o Modified the explanation of INVALID_CERTIFICATE to allow multiple certs.
- o Added reference to the IPV6 colon delimited presentation format.

- o Small editorial changes.

Changes from version 05 to 06:

- o Editorial changes.
- o Unified the example in [Section 3](#).

Changes from version 06 to 07:

- o Editorial changes.
- o Removed a the second paragraph in [section 8](#).
- o Changed the example in [Appendix A](#) (Cert created without the leading zeroes in HITs).

Changes from version 07 to 08:

- o Updated and checked the references.

Changes from version 08 to 09:

- o Fixing boilerplate.

Changes from version 09 to 10:

- o IANA considerations updated based on the IANA review.
- o Updates based on the hip-chairs review.
- o Updates based on the Gen-ART review.

Changes from version 10 to 11:

- o Fixed the nits.

Changes from version 11 to 12:

- o Updates based on the PKIX WG review.

Authors' Addresses

Tobias Heer
Communication and Distributed Systems, RWTH Aachen University
Ahornstrasse 55
Aachen
Germany

Phone: +49 241 80 20 776

Email: heer@cs.rwth-aachen.de

URI: <http://www.comsys.rwth-aachen.de/team/tobias-heer/>

Samu Varjonen
Helsinki Institute for Information Technology
Gustaf Haellstroemin katu 2b
Helsinki
Finland

Email: samu.varjonen@hiit.fi

URI: <http://www.hiit.fi>

