NETWORK WORKING GROUP   N. Williams
Internet-Draft   Sun
Intended status: Standards Track   April 21, 2008
Expires: October 23, 2008

Unique Channel Bindings for IPsec Using IKEv2
draft-williams-ipsec-unique-channel-binding-00.txt

Status of this Memo

Copyright Notice

Abstract

This document specifies the unique channel bindings for IPsec channels constructed by connection latching, where the peers used the Internet Key Exchange protocol version 2 (IKEv2). New IKEv2 notification payloads are used to select an IKE_SA from which to derive the unique channel bindings for a given IPsec channel.

Table of Contents

1. Introduction

Given the ability to construct IPsec channels [I-D.ietf-btns-connection-latching] and the ability to bind authentication at application layers to such secure channels [RFC5056] the only missing components are: a definition of IPsec channel bindings, and Application Programming Interfaces (APIs) by which applications can obtain them.

End-point channel bindings for IPsec are described in [I-D.williams-ipsec-channel-binding]. This document specifies how to construct unique channel bindings for IPsec channels. IPsec APIs [I-D.ietf-btns-ipsec-apireq] are out of scope for this document.

The construction of unique channel bindings given below is applicable only to IPsec channels whose IPsec child SAs are negotiated via the Internet Key Exchange protocol (IKEv2) [RFC4306] regardless of peer authentication method used, though it is extensible to any key exchange protocol for IPsec. Manually established SAs are not supported.

Unlike IPsec end-point channel bindings, IPsec unique channel bindings do make reference to the actual contents of an individual key exchange. Also unlike IPsec end-point channel bindings, IPsec unique channel bindings support IKEv2 authentication methods other than public keys.

1.1. Conventions used in this document

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].

2. IPsec Unique Channel Bindings

In order to obtain unique channel bindings for IPsec that are cryptographically strong (so that a man in the middle cannot cause two connection's channel bindings to agree) we need to derive channel bindings from material from IKE_SAs, or from IPsec child SAs. However, the construction of IPsec channels described in [I-D.ietf-btns-connection-latching] is expressly independent from any individual IKE_SAs and IPsec child SAs. Therefore we need to identify an SA, and this requires a way to agree on a representative SA for any IPsec channel whose unique channel bindings are desired.

The unique channel bindings for IPsec channels established via connection latching [I-D.ietf-btns-connection-latching] between peers that use IKEv2 [RFC4306] SHALL be the octet string consisting of the first 16 octets output by prf+(SK_d, "unique channel binding"), where SK_d is taken from the a selected IKE_SA.

The IKE_SA whose SK_d to use SHALL be selected by an exchange of Notify messages as follows.

When an application requests the unique channel bindings for an IPsec channel the node must either already know these from a previous request, or it MUST pick or initiate an IKE_SA with the channel's peer, and send a UNIQUE_CB_PROPOSE notification with the critical bit set. The contents (see below) of this notification identify a connection latch associated with the channel for which the application requested the channel bindings. The peer, upon receipt of this notification, MUST respond with a UNIQUE_CB_AGREE notification whose contents identify the same connection latch, a UNIQUE_CB_DISAGREE notification, if the connection latch in the proposal could not be found, or, if it does not support this feature, an UNSUPPORTED_CRITICAL_PAYLOAD notification (as usual for IKEv2).

The contents of all three of these notifications' payloads are the traffic selectors for a 5-tuple (transport protocol, source address, source port, destination address, destination port), where "source" refers to the sender of the notification.

Note that SCTP associations can have multiple IPv4 and IPv6 addresses for each peer. One can model this as NxM address pairs with one source and destination address each. Any one of those plus the source and destination ports will, for our purposes, identify an established SCTP association.

When a node receives a UNIQUE_CB_PROPOSE notification it MUST first look for IPsec channels identified by the traffic selectors contained therein. If none is found then the node MUST RESPOND with a

UNIQUE_CB_NOTFOUND notification. If an established IPsec channel is found but it already has a unique channel binding computed from a different IKE_SA, or if an as yet unconfirmed UNIQUE_CB_PROPOSE has been sent for the same channel but on a different IKE_SA, then the node MUST respond with a UNIQUE_CB_DISAGREE notification. Otherwise the node MUST compute the unique channel binding from the IKE_SA used to protect the proposal and MUST record the unique channel bindings and the SPI of the IKE_SA in the identified IPsec channel. Once the channel has been updated the node MUST send a UNIQUE_CB_AGREE notification.

It is extremely unlikely that two peers will attempt to simultaneously send a UNIQUE_CB_PROPOSE to each other for the same IPsec channel. That's because UNIQUE_CB_PROPOSE is sent in response to an application's request for unique channel bindings, and channel binding applications tend to follow a synchronized set of steps. However, should this happen there is no problem, as if the two peers send UNIQUE_CB_PROPOSE using the same IKE_SA then they will both agree on the same channel bindings. If the two peers use different IKE_SAs then at least one peer will, by the above rules, reply with UNIQUE_CB_DISAGREE, and eventually they will either agree or give up.

2.1. Formats of UNIQUE_CB_PROPOSE, UNIQUE_CB_AGREE and UNIQUE_CB_DISAGREE payloads

The UNIQUE_CB_* payloads contain:

- o Protocol ID (TCP, UDP, SCTP, ...)
- o One source port
- o One destination port
- o A type of address (IPv4 or IPv6)
- o One source address
- o One destination address

This is not sufficient to represent all of an SCTP association's addresses, but it is sufficient to identify any SCTP association.

```
        0                   1                   2                   3
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       | IP addr type  | IP proto ID   | RESERVED                      |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       | Source port                   | Destination port              |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       !                                                               !
       ~                       Source address                          ~
       |                                                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       !                                                               !
       ~                     Destination address                       ~
       |                                                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Format of UNIQUE_CB_* notification
payload

3. IANA Considerations

This document creates a type of channel binding, and so requires registration in the IANA channel binding registry (set out by [RFC5056]).

The registration procedure will be followed when this document enters the RFC-Editor queue. The registration will be as follows:

- Channel binding unique prefix (name): IPsec-unique

- Channel binding type: unique

- Channel type: IPsec

- Published specification: <TBD - this document>

- Channel binding is secret: no

- Description: see Section 2

- Intended usage: COMMON

- Contact: this document's author/editor

- Owner/Change controller: IETF

4. Security Considerations

The security considerations of [RFC5056], [I-D.ietf-btns-connection-latching], and IPsec generally [RFC4301] apply. The security of an application using channel binding to IPsec channels depends critically on the overall security of each of these components: IPsec [RFC4301], including the Internet Key Exchange (IKEv2) protocol [RFC4306], ESP/AH [RFC4303] [RFC4302], IPsec connection latching [I-D.ietf-btns-connection-latching], and the application's authentication and channel binding mechanism (potentially too many to reference here, but a common example is likely to be the Kerberos V mechanism [RFC4121] for the Generic Security Services API (GSS-API) [RFC2743]. A compromise of any one of those components may compromise the application to varying degrees.

This document describes unique channel bindings for some IPsec channels. Unique channel bindings uniquely identify a connection in time. There are no additional security considerations, relating to the type of this channel binding, beyond those described in [RFC5056].

Use of non-pre-shared Raw RSA public keys or certificates that cannot be validated to a given trust anchor is supported in the Better Than Nothing (BTNS) [I-D.ietf-btns-prob-and-applic] [I-D.ietf-btns-core] model. When combined with connection latching and channel binding BTNS can provide all the security that an application requires but without having to deploy an IPsec authentication infrastructure (e.g., a PKI, manual pre-sharing of raw RSA public keys and/or self-signed certificates).

Unlike the construction of IPsec end-point channel bindings given in [I-D.williams-ipsec-channel-binding], there are no security considerations with respect to hash agility in this construction of IPsec unique channel bindings, none beyond the algorithm agility considerations that apply to IKEv2 anyways.

5. References

5.1. Normative References

[I-D.ietf-btns-connection-latching]
Williams, N., "IPsec Channels: Connection Latching", draft-ietf-btns-connection-latching-06 (work in progress), February 2008.

[I-D.ietf-btns-ipsec-apireq]
Richardson, M. and B. Sommerfeld, "Requirements for an IPsec API", draft-ietf-btns-ipsec-apireq-00 (work in progress), April 2006.

[I-D.williams-ipsec-channel-binding]
Williams, N., "Channel Bindings for IPsec Using IKEv2 and Public Keys", draft-williams-ipsec-channel-binding-00 (work in progress), March 2008.

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

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, December 2005.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, December 2005.

[RFC4306] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", RFC 4306, December 2005.

[RFC5056] Williams, N., "On the Use of Channel Bindings to Secure Channels", RFC 5056, November 2007.

5.2. Informative References

[I-D.ietf-btns-core]
Williams, N. and M. Richardson, "Better-Than-Nothing-Security: An Unauthenticated Mode of IPsec", draft-ietf-btns-core-06 (work in progress), January 2008.

[I-D.ietf-btns-prob-and-applic]
Touch, J., Black, D., and Y. Wang, "Problem and Applicability Statement for Better Than Nothing Security (BTNS)", draft-ietf-btns-prob-and-applic-06 (work in

progress), October 2007.

[RFC2743] Linn, J., "Generic Security Service Application Program Interface Version 2, Update 1", RFC 2743, January 2000.

[RFC4121] Zhu, L., Jaganathan, K., and S. Hartman, "The Kerberos Version 5 Generic Security Service Application Program Interface (GSS-API) Mechanism: Version 2", RFC 4121, July 2005.

Author's Address

Nicolas Williams
Sun Microsystems
5300 Riata Trace Ct
Austin, TX 78727
US

Email: Nicolas.Williams@sun.com

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).