IPv6 Operations WG
Internet-Draft
Expires: February 26, 2006

R. Graveman
RFG Security, LLC
M. Parthasarathy
Nokia
P. Savola
CSC/FUNET
H. Tschofenig
Siemens
August 25, 2005

Using IPsec to Secure IPv6-in-IPv4 Tunnels
draft-ietf-v6ops-ipsec-tunnels-01.txt

Status of this Memo

Copyright Notice

Abstract

This document gives guidance on securing manually configured IPv6-in-IPv4 tunnels using IPsec. No additional protocol extensions are described beyond those available with the IPsec framework.

Table of Contents

1. Introduction

The IPv6 operations (v6ops) working group has selected (manually configured) IPv6-in-IPv4 tunneling [I-D.ietf-v6ops-mech-v2] as one of the IPv6 transition mechanisms for IPv6 deployment.

[I-D.ietf-v6ops-mech-v2] identified a number of threats which had not been adequately analyzed or addressed in its predecessor, [RFC2893]. The most complete solution is to use IPsec to protect IPv6-in-IPv4 tunneling. The document was intentionally not expanded to include the details on how to set up an IPsec-protected tunnel in an interoperable manner, but instead the details were deferred to this memo.

First this document analyses the threats and scenarios that can be addressed by IPsec. Next, this document discusses some of the assumptions made by this document for successful IPsec Security Association (SA) establishment. Then, it gives the details of Internet Key Exchange (IKE) and IP security (IPsec) exchange with packet formats and Security Policy Database (SPD) entries. Finally, it discusses the usage of IPsec NAT-traversal mechanism that can be used with configured tunnels in some scenarios.

This document does not address the use of IPsec for tunnels which are not manually configured (e.g., 6to4 tunnels [RFC3056]). Presumably, some form of opportunistic encryption or "better-than-nothing security" might or might not be applicable. Similarly, propagating quality of service attributes (apart from Explicit Congestion Notification (ECN) bits [I-D.ietf-v6ops-mech-v2]) from the encapsulated packets to the tunnel path is out of scope.

2. Threats and the Use of IPsec

[I-D.ietf-v6ops-mech-v2] is mostly concerned about address spoofing threats:

1. IPv4 address of the encapsulating ("outer") packet can be spoofed.

2. IPv6 address of the encapsulated ("inner") packet can be spoofed.

IPsec can obviously also provide payload integrity and confidentiality as well for the part of the end-to-end path that is tunneled.

The reason for threat (1) is the lack of widespread deployment of IPv4 ingress filtering [RFC3704]. The reason for threat (2) is that

the IPv6 packet is encapsulated in IPv4 and hence may escape IPv6 ingress filtering. [I-D.ietf-v6ops-mech-v2] specifies the following strict address checks as mitigating measures:

- To mitigate threat (1), the decapsulator verifies that the IPv4 source address of the packet is the same as the address of the configured tunnel endpoint. The decapsulator may also implement IPv4 ingress filtering, i.e., checks whether the packet is received on a legitimate interface.

- To mitigate threat (2), the decapsulator verifies whether the inner IPv6 address is a valid IPv6 address and also applies IPv6 ingress filtering before accepting the IPv6 packet.

This memo proposes using IPsec for providing stronger security in preventing these threats and additionally providing integrity and confidentiality. IPsec can be used in two ways, in transport and tunnel mode; further comparison is done in Section 5.1.

2.1. IPsec in Transport Mode

In transport mode, the IPsec security association (SA) is established to protect the traffic defined by (IPv4-source, IPv4-dest, protocol = 41). On receiving such an IPsec packet, the receiver first applies the IPsec transform (ESP) and then matches the packet against the Security Parameter Index (SPI) and the inbound selectors associated with the SA to verify that the packet is appropriate for the SA via which it was received. A successful verification implies that the packet came from the right IPv4 endpoint as the SA is bound to the IPv4 source address.

This prevents threat (1) but not the threat (2). IPsec in transport mode does not verify the contents of the payload itself where the IPv6 addresses are carried, that is, two nodes that are using IPsec transport mode to secure the tunnel can spoof the inner payload. The packet will be decapsulated successfully and accepted.

The shortcoming can be mitigated by IPv6 ingress filtering i.e., check that the packet is arriving from the interface in the direction of the route towards the tunnel end-point, similar to a Strict Reverse Path Forwarding (RPF) check [RFC3704].

In most implementations, a transport mode SA is applied to a normal IPv6-in-IPv4 tunnel. Therefore, ingress filtering can be applied in the tunnel interface. (Transport mode is often also used in other kind of tunnels such as GRE and L2TP.)

2.2. IPsec in Tunnel Mode

In tunnel mode, the IPsec SA is established to protect the traffic defined by (IPv6-source, IPv6-destination). On receiving such an IPsec packet, the receiver first applies the IPsec transform (ESP) and then matches the packet against the SPI and the inbound selectors associated with the SA to verify that the packet is appropriate for the SA via which it was received. The successful verification implies that the packet came from the right endpoint.

The outer IPv4 addresses may be spoofed and IPsec cannot detect it in this mode; the packets will be demultiplexed based on the SPI and possibly the IPv6 address bound to the SA. Thus, the outer address spoofing is irrelevant as long as the decryption succeeds and the inner IPv6 packet can be verified to come from the right tunnel endpoint.

Tunnel mode SA can be used in two ways depending on whether it is modelled as an interface or not. These are described in section Section 5.3.

3. Scenarios and Overview

There are roughly three kinds of scenarios:

1. (generic) router-to-router tunnels.

2. site-to-route/router-to-site tunnels. This refers to a tunnel between a site's IPv6 (border) device to an IPv6 upstream provider's router. A degenerate case of a site is a single host.

3. Host-to-host tunnels.

3.1. Router-to-Router Tunnels

IPv6/IPv4 hosts and routers can tunnel IPv6 datagrams over regions of IPv4 routing topology by encapsulating them within IPv4 packets. Tunneling can be used in a variety of ways.

```
   .--------.          _----_          .--------.
   |v6-in-v4|        _( IPv4 )_        |v6-in-v4|
   | Router | <======( Internet )=====> | Router |
   |   A    |        (_      _)        |   B    |
   '--------'          '----'          '--------'
       ^        IPsec tunnel between        ^
       |        Router A and Router B       |
       V                                    V
```

Figure 1: Router-to-Router Scenario

IPv6/IPv4 routers interconnected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. In this case, the tunnel spans one segment of the end-to-end path that the IPv6 packet takes.

The source and destination addresses of the IPv6 packets traversing the tunnel could come from a wide range of IPv6 prefixes, so binding IPv6 addresses to be used to the SA is not feasible. IPv6 ingress filtering must be performed to mitigate the IPv6 address spoofing threat.

A specific case of router-to-router tunnels, when one router resides at an end site, is described in the next section.

3.2. Site-to-Router/Router-to-Site Tunnels

This is a generalization of host-to-router and router-to-host tunneling, because the issues when connecting a whole site (using a router), and connecting a single host are roughly equal.

```
      _----_        .---------. IPsec     _----_    IPsec  .-------.
    _( IPv6 )_      |v6-in-v4 | Tunnel  _( IPv4 )_  Tunnel | V4/V6  |
   ( Internet )<--->| Router  |<=======( Internet )=======>| Site B |
    (_      _)      |   A     |        (_      _)          '--------'
      '----'        '---------'          '----'
        ^
        |
        V
    .--------.
    | Native |
    | IPv6   |
    | node   |
    '--------'
```

Figure 2: Router-to-Site Scenario

IPv6/IPv4 routers can tunnel IPv6 packets to their final destination IPv6/IPv4 site. This tunnel spans only the last segment of the end-

to-end path.

```
                                +---------------------+
                                |      IPv6 Network   |
                                |                     |
.--------.        _----_        |     .--------.      |
| V6/V4  |      _( IPv4 )_      |     |v6-in-v4|      |
| Site B |<====( Internet )==========>| Router |      |
'--------'      (_      _)      |     |   A    |      |
                  '----'        |     '--------'      |
        IPsec tunnel between    |         ^           |
        IPv6 Site and Router A  |         |           |
                                |         V           |
                                |     .-------.       |
                                |     |  V6    |      |
                                |     |  Hosts |      |
                                |     '--------'      |
                                +---------------------+
```

Figure 3: Site-to-Router Scenario

Respectively, IPv6/IPv4 hosts can tunnel IPv6 packets to an intermediary IPv6/IPv4 router that is reachable via an IPv4 infrastructure. This type of tunnel spans the first segment of the packet's end-to-end path.

The hosts in the site originate the packets with source addresses coming from a well known prefix whereas the destination address could be any node on the Internet.

In this case, the IPsec tunnel mode SA can be bound to the prefix that was allocated to the router at Site B and router A can verify that the source address of the packet matches the prefix. Site B will not be able to do a similar verification for the packets it receives. This may be quite reasonable for most of the deployment cases, for example, the Internet Service Provider (ISP) allocating a /48 to a customer. The Customer Premises Equipment (CPE) where the tunnel is terminated "trusts" (in a weak sense) the ISP's router and the ISP's router can verify that the Site B is the only one that can originate packets within the /48.

IPv6 spoofing must be prevented, and setting up ingress filtering may require some amount of manual configuration; see more of these options in Section 5.

3.3. Host-to-Host Tunnels

```
 .--------.          _----_          .--------.
 | V6/V4  |        _( IPv4 )_        | V6/V4  |
 | Host   | <======( Internet )=====> | Host   |
 |   A    |        (_        _)        |   B    |
 '--------'          '----'          '--------'
                IPsec tunnel between
                Host A and Host B
```

Figure 4: Host-to-Host Scenario

IPv6/IPv4 hosts that are interconnected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. In this case, the tunnel spans the entire end-to-end path that the packet takes.

In this case, the source and the destination IPv6 address are known a priori. A tunnel mode SA can be bound to the specific address. The address verification prevents IPv6 address spoofing completely.

As noted in Introduction, automatic host-to-host tunneling methods (e.g., 6to4) are out of scope of this memo.

4. IKE and IPsec Versions

This section discusses the different versions of the IKE and IPsec security architecture and their applicability to this document.

The IPsec security architecture was originally defined in [RFC2401] and now superseded by [I-D.ietf-ipsec-rfc2401bis]. IKE was originally defined in [RFC2409] (which is referred to as IKEv1 in this document) and is now superseded by [I-D.ietf-ipsec-ikev2] (referred to as IKEv2). There are several differences between them. The differences relevant to this document are discussed below.

1. [RFC2401] does not allow IP as the next layer protocol in traffic selectors when IPsec SA is negotiated. [I-D.ietf-ipsec-rfc2401bis] also allows IP as the next layer protocol like TCP or UDP in traffic selectors.

2. [RFC2401] does not support transport mode SAs between hosts and security gateways. [I-D.ietf-ipsec-rfc2401bis] supports transport mode SA between hosts and security gateway to provide link security e.g., IP-IP tunnel protected with IPsec.

3. [I-D.ietf-ipsec-rfc2401bis] assumes IKEv2, as some of the new features cannot be negotiated using IKEv1. It is valid to

negotiate multiple traffic selectors for a given IPsec SA in [I-D.ietf-ipsec-rfc2401bis]. This is possible only with [I-D.ietf-ipsec-ikev2]. If [RFC2409] is used, then multiple SAs need to be set up for each traffic selector.

Note that the existing implementations based on [RFC2409] may already be able to support the [I-D.ietf-ipsec-rfc2401bis] features described in (1) and (2). If appropriate, the deployment may choose to use the two versions of the security architecture.

IKEv2 supports features that are useful for configuring and securing tunnels which are not present with IKEv1.

1. IKEv2 supports legacy authentication methods by carrying them in EAP payloads. This can be used to authenticate the hosts/sites to the ISP using EAP methods that support username and password.

2. IKEv2 supports dynamic address configuration which may be used to configure the IPv6 address of the host.

NAT traversal works with both the old and revised IPsec architectures, but the negotiation is integrated with IKEv2.

We do not consider the usage of the IP Authentication Header (AH) [I-D.ietf-ipsec-rfc2402bis] as ESP [I-D.ietf-ipsec-esp-v3] provides security services (such as integrity protection without confidentiality protection using 'NULL' encryption) which are comparable with AH. This is explicitly stated in [I-D.ietf-ipsec-rfc2401bis].

5. IPsec Configuration Details

This section describes details about establishment of an IPsec tunnel for the protection of IPv4/IPv6 data traffic. However, first we will take a look at the packet format on the wire, and the salient differences between transport and tunnel modes.

The packet format is the same for both transport mode and tunnel mode as shown in Table 1.

Components (first to last)	Contains
IPv4 header	(src = IPV4-TEP1, dst = IPV4-TEP2)
ESP header	
IPv6 header	(src = IPV6-EP1, dst = IPV6-EP2)
(payload)	

Table 1

5.1. Transport vs Tunnel Mode

Transport mode is typically used by setting up a regular IPv6-in-IPv4 (or GRE, L2TP, ...) tunnel, and then applying a transport mode SA to protect the packets before they are sent out over an interface.

Tunnel mode can be deployed in two very different ways depending on the implementation:

1. "Generic SPDs": some implementations model the tunnel mode SA as an IP interface. In this case, an IPsec tunnel interface is created and used with "any" address ("::/0 <-> ::/0") as IPsec traffic selectors while setting up the SA. Though this allows all traffic between the two nodes to be protected by IPsec, the routing table would decide what traffic gets sent over the tunnel. Ingress filtering must be separately applied on the tunnel interface as the IPsec policy checks do not check the IPv6 addresses at all. Routing protocols, multicast, etc. will work through this tunnel. This mode is very similar to the transport mode.

2. "Specific SPDs": some implementations don't model the tunnel mode SA as an IP interface. Traffic selection is done based on specific SPD entries, e.g., "2001:db8:1::/48 <-> 2001:db8:2::/48". As the IPsec session between two endpoints does not have an interface (though an implementation may have a common pseudo-interface for all IPsec traffic), there is no DAD, MLD, or link-local traffic to protect; multicast is not possible over such a tunnel. Ingress filtering is performed automatically by the IPsec traffic selectors.

Ingress filtering is guaranteed by IPsec processing when option (2) is chosen whereas the operator has to enable them explicitly when transport mode or option (1) of tunnel mode SA is chosen.

We describe the specific SPD case in Appendix A due to its length and relative complexity compared to transport mode or generic SPD tunnel

mode.

5.2. IPsec Transport Mode

The transport mode has typically been applied to L2TP, GRE, and other kind of tunneling methods, especially when the user wants to tunnel non-IP traffic. [RFC3884] provides an example of applicability.

IPv6 ingress filtering must be applied on the tunnel interface on all the packets which pass the inbound IPsec processing.

The following SPD entries assume that there are two routers Router1 and Router2, with tunnel endpoint IPv4 addresses are denoted by IPV4-TEP1 and IPV4-TEP2 respectively. (In other scenarios, the SPDs are set up in a similar fashion.) Implementations that are strictly conformant to [RFC2401] may not be able to setup the IPsec transport mode SA.

```
Router1's SPD OUT :

IF SRC = IPV4-TEP1 && DST = IPV4-TEP2 && protocol = 41
    THEN USE ESP TRANSPORT MODE SA

Router1's SPD IN:

IF SRC = IPV4-TEP2 && DST = IPV4-TEP1 && protocol = 41
    THEN USE ESP TRANSPORT MODE SA


Router2's SPD OUT:

IF SRC = IPV4-TEP2 && DST = IPV4-TEP1 && protocol = 41
    THEN USE ESP TRANSPORT MODE SA

Router2's SPD IN:

IF SRC = IPV4-TEP1 && DST = IPV4-TEP2 && protocol = 41
    THEN USE ESP TRANSPORT MODE SA
```

The IDci and IDcr payloads of IKEv1 carry the IPv4-TEP1, IPV4-TEP2 and protocol value 41 as phase 2 identities. With IKEv2, the traffic selectors are used to carry the same information.

5.3. IPsec Tunnel Mode

As we described above, tunnel mode can be used either with "generic" or "specific" SPDs. We describe the generic approach below, and

specific SPDs in Appendix A.

Implementations may or may not model a tunnel mode SA as a separate interface between each IPsec peer. A separate interface for each is simple as long as generic SPDs are used. However, with specific SPDs, having an interface becomes highly problematic. That is, interfaces must always have link-local addresses, run Duplicate Address Detection, etc. -- which results in packets which must be secured. These would require a set-up of a number of complex SPDs because link-local addresses are not unique. Therefore, this memo restricts to describing only the scenario where SPD tunnel mode is not modelled as separate interfaces.

Routing protocols, multicast, etc. work fine over generic SPD tunnel mode, but are not feasible with specific SPDs.

5.3.1. Generic SPDs for Tunnel Mode

In the generic SPD case, for any scenario, SPDs are not really used for traffic selectors. All the SPD entries match all the traffic, i.e., "src = ::/0 & destination = ::/0" (we do not write these out as the SPD entries are trivial). We assume that the tunnel is modelled as an interface, one for each IPsec session. Instead of SPDs, the routing table is used to perform outbound traffic selection, and all the traffic that is passed to the interface, gets IPsec-protected.

Similarly, the inbound SPD matches everything, so demultiplexing is done based on the SPI. This is secure; while an attacker could spoof packets with the correct SPI (and even tunnel source/destination addresses), the attacker would not know the keying material and such packets would fail IPsec processing.

This mode obviously does not prevent an attacker from spoofing IPv6 addresses, as any traffic sent by the IPsec peer is accepted. Therefore, ingress filtering must be applied on the tunnel interface.

As all (IP) traffic will pass on this kind of tunnel, routing protocols, multicast, etc. will work without problems.

6. Dynamic Address Configuration

With the exchange of protected configuration payloads, IKEv2 is able to provide the IKEv2 peer with DHCP-like information payloads. These configuration payloads are exchanged between the IKEv2 initiator and the responder.

This can be used (for example) by the host in the host-to-router

scenario to obtain the IPv6 address from the ISP as part of setting up the IPsec tunnel mode SA. The details of these procedures are out of scope of this memo.

7. NAT Traversal and Mobility

Network address (and port) translation devices are commonly found in today's networks. A detailed description of the problem of IPsec protected data traffic traversing a NAT including requirements are discussed in [RFC3715].

IKEv2 can detect the presence of a NAT automatically by sending an Informational exchange with NAT_DETECTION_SOURCE_IP and NAT_DETECTION_DESTINATION_IP payloads before establishing an IPsec SA. These payloads are processed in the same way as in the initial IKE_SA_INIT exchange. Once a NAT is detected and both end points support IPsec NAT traversal extensions UDP encapsulation can be enabled.

More details about UDP encapsulation of IPsec protected IP packets can be found in [RFC3948].

For IPv6-in-IPv4 tunneling, NAT traversal is interesting for two reasons:

1. One of the tunnel endpoints is often behind a NAT, and configured tunneling, using protocol 41, is not guaranteed to traverse the NAT. Hence, using IPsec tunnels would enable one to both set-up a secure tunnel, and set-up a tunnel where it might not always be possible without other tunneling mechanisms.

2. Using NAT traversal allows the outer address to change without having to renegotiate the SAs. This could be very beneficial for a crude form of mobility, and in scenarios where the NAT changes the IP addresses frequently. However, as the outer address may change, this might introduce new security issues, and using tunnel mode would be most appropriate.

When NAT is not applied, the second benefit would still be desirable. In particular, using manually configured tunneling is an operational challenge with dynamic IP addresses as both ends need to be reconfigured if an address changes. Therefore an easy and efficient way to re-establish the IPsec tunnel if the IP address changes would be desirable. The IETF MOBIKE working group is looking into providing a solution for IKEv2 but the work is still in progress [I-D.ietf-mobike-protocol].

8. Tunnel Endpoint Discovery

The IKEv2 initiator needs to know the address of the IKEv2 responder to start IKEv2 signaling. A number of ways can be used to provide the initiator with this information, for example:

- Using out-of-band mechanisms, e.g., from the ISP's web page.
- Using DNS to look up a service name by appending it to the DNS search path provided by DHCPv4 (e.g. "tunnel-service.example.com").
- Using a DHCP option.
- Using a pre-configured or pre-determined IPv4 anycast address.
- Using other, unspecified or proprietary methods.

For the purpose of this document it is assumed that this address can be obtained somehow. Once the address has been learned, it is configured as the tunnel end-point for the configured IPv6-in-IPv4 tunnel.

This problem is also discussed at more length in [I-D.palet-v6ops-tun-auto-disc].

9. Recommendations

In Section 5 we examined the differences of setting up an IPsec IPv6-in-IPv4 using either tunnel or transport mode. We observe that the transport mode and tunnel mode with generic SPDs are very similar; multicast and routing protocols work over both, and ingress filtering must be applied on the tunnel interface manually.

Tunnel mode with specific SPDs is slightly more complicated. The approach does not seem feasible if modelled as an interface, so we do not recommend it. Without an interface, the main benefit is that it automatically applies ingress filtering within the IPsec processing. However, multicast, routing protocols, etc. are not feasible with this approach, so its applicability is limited to host-to-host or edge tunnel cases.

Tunnel mode may be more attractive when the IPv4 tunnel endpoint addresses change, as MOBIKE only supports tunnel mode.

Therefore our primary recommendation is to use either tunnel mode with generic SPDs or transport mode, and apply ingress filtering on

the tunnel.

10. IANA Considerations

This memo makes no request to IANA. [[RFC-editor: please remove this section prior to publication.]]

11. Security Considerations

When you run IPv6-in-IPv4 tunnels (unsecured) over the Internet, it is possible to "inject" packets into the tunnel by spoofing the source address (data plane security), or if the tunnel is signalled somehow (e.g., some messages where you authenticate to the server, so that you would get a static v6 prefix), someone might be able to spoof the signalling (control plane security).

The IPsec framework plays an important role in adding security to both the protocol for tunnel setup and data traffic.

Either IKEv1 or IKEv2 provides a secure signaling protocol for establishing, maintaining and deleting an IPsec tunnel.

IPsec, with the Encapsulating Security Payload (ESP), offers integrity and data origin authentication, confidentiality, with optional (at the discretion of the receiver) anti-replay features. The usage of confidentity-only is discouraged. ESP furthermore provides limited traffic flow confidentality.

IPsec provides access control mechanisms through the distribution of keys and also through the usage of policies dictated by the Security Policy Database (SPD).

The NAT traversal mechanism provided by IKEv2 introduces some weaknesses into IKE and IPsec. These issues are discussed in more detail in [I-D.ietf-ipsec-ikev2].

Please note that the usage of IPsec for the scenarios described in Figure 3, Figure 2 and Figure 1 does not aim to protect the end-to-end communication. It protects just the tunnel part. It is still possible for an IPv6 endpoint that is not attached to the IPsec tunnel to spoof packets.

12. Contributors

The authors are listed in alphabetical order.

Suresh Satapati also participated in the initial discussions on the topic.

13. Acknowledgments

The authors would like to thank Stephen Kent, Michael Richardson, Florian Weimer, Elwyn Davies, and Eric Vyncke for their substantive feedback.

We would like to thank Pasi Eronen for his text contributions.

14. References

14.1. Normative References

[I-D.ietf-ipsec-esp-v3]
Kent, S., "IP Encapsulating Security Payload (ESP)", draft-ietf-ipsec-esp-v3-10 (work in progress), March 2005.

[I-D.ietf-ipsec-ikev2]
Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", draft-ietf-ipsec-ikev2-17 (work in progress), October 2004.

[I-D.ietf-ipsec-rfc2401bis]
Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", draft-ietf-ipsec-rfc2401bis-06 (work in progress), April 2005.

[I-D.ietf-v6ops-mech-v2]
Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", draft-ietf-v6ops-mech-v2-07 (work in progress), March 2005.

[RFC2401] Kent, S. and R. Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, November 1998.

[RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

[RFC2461] Narten, T., Nordmark, E., and W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)", RFC 2461, December 1998.

[RFC2710] Deering, S., Fenner, W., and B. Haberman, "Multicast Listener Discovery (MLD) for IPv6", RFC 2710,

October 1999.

[RFC3704] Baker, F. and P. Savola, "Ingress Filtering for Multihomed Networks", BCP 84, RFC 3704, March 2004.

[RFC3948] Huttunen, A., Swander, B., Volpe, V., DiBurro, L., and M. Stenberg, "UDP Encapsulation of IPsec ESP Packets", RFC 3948, January 2005.

14.2. Informative References

[I-D.ietf-ipsec-rfc2402bis] Kent, S., "IP Authentication Header", draft-ietf-ipsec-rfc2402bis-11 (work in progress), March 2005.

[I-D.ietf-mobike-protocol] Eronen, P., "IKEv2 Mobility and Multihoming Protocol (MOBIKE)", draft-ietf-mobike-protocol-01 (work in progress), July 2005.

[I-D.palet-v6ops-tun-auto-disc] Palet, J. and M. Diaz, "Analysis of IPv6 Tunnel End-point Discovery Mechanisms", draft-palet-v6ops-tun-auto-disc-03 (work in progress), January 2005.

[RFC2893] Gilligan, R. and E. Nordmark, "Transition Mechanisms for IPv6 Hosts and Routers", RFC 2893, August 2000.

[RFC3056] Carpenter, B. and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", RFC 3056, February 2001.

[RFC3715] Aboba, B. and W. Dixon, "IPsec-Network Address Translation (NAT) Compatibility Requirements", RFC 3715, March 2004.

[RFC3884] Touch, J., Eggert, L., and Y. Wang, "Use of IPsec Transport Mode for Dynamic Routing", RFC 3884, September 2004.

Appendix A. Specific SPDs for Tunnel Mode

We describe the specific SPD case in an appendix due to its length and relative complexity compared to transport mode or generic SPD tunnel mode.

We assume that this kind of IPsec association is not modelled an interface, because then the link-local traffic would require very

complex SPDs as well.

A.1. Specific SPD for Host-to-Host Scenario

The following SPD entries assume that there are two hosts Host1 and Host2, whose IPv6 addresses are denoted by IPV6-EP1 and IPV6-EP2 (global addresses) and IPV4 addresses of the tunnel endpoints are denoted by IPV4-TEP1 and IPV4-TEP2 respectively.

The outbound SPD will encrypt the traffic to the specified global IPv6 address.

```
Host1's SPD OUT :

IF SRC = IPV6-EP1 & DST = IPV6-EP2
    THEN USE ESP TUNNEL MODE SA:
        outer source = IPv4-TEP1
        outer dest   = IPV4-TEP2

Host1's SPD IN:

IF SRC = IPV6-EP2 && DST = IPV6-EP1
    THEN USE ESP TUNNEL MODE SA
        outer source = IPV4-TEP2
        outer dest   = IPV4-TEP1

Host2's SPD OUT:

IF SRC = IPV6-EP2 & DST = IPV6-EP1
    THEN USE ESP TUNNEL MODE SA:
        outer source = IPv4-TEP2
        outer dest   = IPV4-TEP1

Host2's SPD IN:

IF SRC = IPV6-EP1 && DST = IPV6-EP2
    THEN USE ESP TUNNEL MODE SA:
        outer source = IPv4-TEP1
        outer dest   = IPV4-TEP2
```

The IDci and IDcr payloads of IKEv1 carry the IPV6-EP1 and IPV6-TEP2 as phase 2 identities. With IKEv2, the traffic selectors are used to carry the same information.

A.2. Specific SPD for Host-to-Router scenario

The following SPD entries assume that the host has the IPv6 address

IPV6-EP1 and the tunnel end points of the host and router are IPV4-TEP1 and IPV4-TEP2 respectively. If the tunnel is between a router and a host where the router has allocated a IPV6-PREF/48 to the host, the corresponding SPD entries can be derived by substituting IPV6-EP1 by IPV6-PREF/48.

Please note the bypass entry for host's outbound SPD, and the corresponding router's inbound SPD. While this might be an implementation matter for host-to-router tunneling, having a similar entry, "SRC=IPV6-PREF/48 & destination=IPV6-PREF/48" would be critical for site-to-router tunneling.

```
Host's SPD OUT:

IF SRC=IPV6-EP1 & DST = IPV6-EP1
    THEN BYPASS

IF SRC = IPV6-EP1 & DST = any
    THEN USE ESP TUNNEL MODE SA:
        outer source = IPv4-TEP1
        outer dest   = IPV4-TEP2

Host's SPD IN:

IF SRC = any && DST = IPV6-EP1
    THEN use ESP TUNNEL MODE SA
        outer source = IPV4-TEP2
        outer dest   = IPV4-TEP1

Router's SPD OUT:

IF SRC = any & DST = IPV6-EP1
    THEN USE ESP TUNNEL MODE SA:
        outer source = IPv4-TEP2
        outer dest   = IPV4-TEP1

Router's SPD IN:

IF SRC=IPV6-EP1 & DST = IPV6-EP1
    THEN BYPASS

IF SRC = IPV6-EP1 && DST = any
    THEN use ESP TUNNEL MODE SA
        outer source = IPV4-TEP1
        outer dest   = IPV4-TEP2
```

The IDci and IDcr payloads of IKEv1 carry the IPV6-EP1 and

ID_IPV6_ADDR_RANGE or ID_IPV6_ADDR_SUBNET as its phase 2 identity. The starting address is zero IP address and the end address is all ones for ID_IPV6_ADDR_RANGE. The starting address is zero IP address and the end address is all zeroes for ID_IPV6_ADDR_SUBNET. With IKEv2, the traffic selectors are used to carry the same information.

Authors' Addresses

Richard Graveman
RFG Security, LLC
15 Park Avenue
Morristown, New Jersey 07960
USA

Email: rfg@acm.org

Mohan Parthasarathy
Nokia
313 Fairchild Drive
Mountain View CA-94043
USA

Email: mohanp@sbcglobal.net

Pekka Savola
CSC/FUNET
Espoo
Finnland

Email: psavola@funet.fi

Hannes Tschofenig
Siemens
Otto-Hahn-Ring 6
Munich, Bayern 81739
Germany

Email: Hannes.Tschofenig@siemens.com

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.