Network Working Group J. Melen
Internet-Draft J. Ylitalo
Expires: January 28, 2009 P. Salmela
Ericsson Research NomadicLab
July 27, 2008

Security Parameter Index multiplexed Network Address Translation (SPINAT)
draft-melen-spinat-01

Status of this Memo

Copyright Notice

Abstract

This drafts defines a Security Parameter Index multiplexed Network Address Translator (SPINAT). The SPINAT method uses the SPI value of ESP packets to de-multiplex multiple IP addresses on single IP address. The solution presented in this draft requires a state in the SPINAT node and in the peer node. The state establishment requires a control signaling messages carrying the SPI values.

Table of Contents

1. Introduction

The IP Network Address Translator (Traditional NAT) functionality is explained in [RFC3022]. From the IP address translation point of view, it is possible to compare the traditional Network Address Port Translation (NAPT) method to SPINAT method, i.e. presented in this draft. A SPINAT node connects a private link to a WAN link, like a NAPT node does. The NAPT method uses port values in TCP/UDP headers for multiplexing IP addresses, while the SPINAT method uses Security Parameter Index (SPI) values in the ESP [RFC4303] headers for that purpose.

There are few differences in the operation of SPINAT compared to NAPT.

> In NAPT the port values in the TCP/UDP headers are not integrity protected, unlike in the SPINAT case where the SPI values in ESP headers are integrity protected. From the SPI translation point of view this is a problem, because the related ESP integrity protection keys are only shared between the end-points, not with the SPINAT nodes. Therefore, the SPINAT nodes cannot transparently translate SPI values like traditional NAPT nodes translate port values. To sustain the integrity of ESP headers and to support SPI translation, the SPINAT nodes MUST inform the sending end-hosts about the translation, unlike the NAPT method about the port translation.
>
> When the Security Associations are setup between the end-hosts, the end-hosts will use a separate control signaling to negotiate the SPI values to be used, unlike in NAPT case where the destination port values are well-known and source ports may be randomly selected and modified. Additionally, the ESP header carries only the destination SPI value, thus a separate control signaling (key-exchange) is needed for state establishment at the SPINAT node, instead of using only ESP packets for state establishment.

The SPINAT node that is in the forwarding path of the two peer nodes will create its state in two steps. The first step is to create a soft-state for the SPI-to-IP address mapping for ESP payload traffic based on the control signaling (key-exchange). However, the SPINAT method does not require explicit state establishment exchange between SPINAT node and the end-hosts. Therefore, the SPINAT method does not increase the amount of signaling compared to a situation when there does not exist SPINAT nodes on the packet forwarding path. The SPINAT node intercepts a control signaling message received from a private link that carries a SPI value. The SPINAT adds a Type-Length-Value (TLV) field containing the SPI mapping information at

the end of the intercepted message before forwarding the message to the WAN link. Next, the state is completed to a hard-state after receiving the first ESP payload packet that carries the SPI corresponding to the SPI-to-IP address mapping.

The SPINAT operation does not require any modifications to the ESP processing at the host in the private network, but requires a modification at the peer host that is in the WAN side to allow the SPINAT node to re-write the SPI value on the received ESP packets. The original SPI value is selected by the receiving end-host in the private IP network, and the value is replaced by the SPINAT node with another SPI value. As a result of the key-exchange, both the SPINAT node and the peer host establish a translation state. The end-host implements the same SPI mapping as SPINAT node for integrity protected ESP packets, but in a reverse order. The SPI translation MUST be made after the ESP integrity protection is computed using the original SPI values.

2. Terminology

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].

client node
   The node residing in the private network.

SPINAT node
   The SPI based Network Address Translation node.

Peer node
   The node residing in the public network.

3. State Establishment

3.1. State Establishment at SPINAT Node

This section describes the state establishment and SA mapping setup in the SPINAT node. After the state has been setup the SPINAT node is ready to forward the packets between the peer nodes.

```
     +--------+                    +--------+                 +--------+
     |   Peer |                    | SPINAT |                 | client |
     +--------+                    +--------+                 +--------+
          |                              |                          |
          |            {SPI=3030-->1212}|              {SPI=3030}|
          |                             \|                         \|
          |<-Key-exchange message-------|--------------------------|
          |                              |                          |
          |                              |                          |
          |{SPI=4545}                    |                          |
          |/                             |                          |
          |-Key-exchange message------->|------------------------->|
```

Figure 1: A round-trip of a key-exchange containing SPI values

Figure 1 illustrates control signaling that contains the SPI values. As a result the SPINAT node creates translation state for the security associations (SAs). The SPI needs to be translated only on the direction from peer to client because only in that directions there is a possibility that two or more clients behind the SPINAT node select the same SPI value for incoming ESP packets causing a SPI collision.

```
     +--------+                    +--------+                 +--------+
     |   Peer |                    | SPINAT |                 | client |
     +--------+                    +--------+                 +--------+
          |                              |                          |
          |(SA-out)       (SA-in-public)|(SA-out-private)   (SA-in)|
          |/          ESP              \|/                        \|
          |---------------------------->|------------------------->|
          |                              |                          |
          |                              |                          |
          |(SA-in)       (SA-out-public)|(SA-in-private)   (SA-out)|
          |/           ESP             \|/                        \|
          |<----------------------------|--------------------------|
```

Figure 2: Naming the Security Associations (SAs)

The translation state contains a incoming SA and a outgoing SA. This pair contains the information needed to forward the packet through the SPINAT node. The information needed in the state is illustrated in the Figure 2

The state on the direction from peer to client contains mapping for SPI value on the public side to a SPI value on the private side and the private address of the client.

The SPI value selected for the SA-in-public MUST be unique for each ESP incoming association. The SPINAT node selects the SPI value for each connection that goes through the SPINAT node. The selected SPI value is carried in the control signaling message to the peer node.

The state on the direction from client to peer contains mapping for client private address to the public address of SPINAT. In this direction there is no need to translate the SPI.

The SPI value used in the SA-out-public is selected by the peer node and thus it does not have to be translated. The SPINAT node is able to use on both sides the same SPI value.

3.2. State Establishment at End-Hosts

This section describes the state needed in the end-hosts to support SPINAT translations. Both End-Hosts create one incoming and outgoing SAs as can be seen from Figure 2.

On the client node that resides behind the SPINAT node in the private network there is no additional state information that should be stored. The client node operates as it would operate in other network and stores the same state data as it would store for any other SA.

The peer node on the WAN side has to store the SPI mapping information that it has received from the SPINAT node in the control signaling. As the SA is created to peer node it needs to store both the SPI value given by the client and the SPI value given by the SPINAT node to the SA. The SPI value given by the client will be used when constructing the ESP packet and calculating message authentication code. This SPI value will later be replaced by the SPI value given by the SPINAT node.

4. Packet Processing

4.1. Control Signaling packet handling

This section describes the packet processing of the control signaling packets.

When a control packet is received from private network following processing steps will be done:

1. Upon receipt of a control signaling packet from private network, the SPINAT node parses the packet.

2. If there is no state for this pair of end-hosts it will create a state that is used to forward the control signaling packets through the SPINAT node.

3. If the packet contains a SPI value the SPINAT node MUST select a unique SPI from its free SPI space. If the packet didn't contain SPI value it will be forwarded and no further processing is applied.

4. SPINAT node MUST add to the state that the mapping between the SPI selected by the client host and the SPI that it has chosen.

5. The SPINAT node adds the SPI TLV to the original control signaling packet.

6. The SPINAT node translates the source address of the packet.

7. SPINAT node forwards the packet to recipient.

When a control packet is received from WAN network following processing steps will be done:

1. Upon receipt of a control signaling packet from WAN network, the SPINAT node looks up the state for the control signaling it has created from previously outgoing packet. If no state is found the packet MUST be discarded.

2. The SPINAT node translates the destination address based on the information stored in to the state.

3. SPINAT forwards the packet to recipient.

4.2. ESP packet processing

4.2.1. IP Address and SPI Translation at SPINAT Nodes

This section describes the packet processing of the ESP packet. Figure 3 illustrates an example of SPINAT packet processing for ESP packets.

```
 +--------+                    +--------+                  +--------+
 |   Peer |                    | SPINAT |                  | client |
 +--------+                    +--------+                  +--------+
      |              {s=138.76.28.4,|           {s=10.0.0.10,  |
      |               d=138.76.29.7,|            d=138.76.29.7,|
      |       ESP     SPI=4545     \|            SPI=4545}    \|
      |<----------------------------|--------------------------|
      |                             |                          |
      | {s=138.76.29.7              | {s=138.76.29.7,          |
      | {d=138.76.29.4,             |  d=10.0.0.10,            |
      |/ SPI=1212}       ESP        |/ SPI=3030}               |
      |---------------------------->|------------------------->|
```

Figure 3: SPINAT Operation for ESP protected payload packets

When a ESP packet is received from private network following processing steps will be done:

1. Upon receipt of a ESP packet from the private network, the SPINAT determines the appropriate translation state, based on the SPI. If no valid state exists for this SPI the SPINAT node MUST discard the packet.

2. The SPINAT node translates the source address of the packet.

3. SPINAT node forwards the packet to recipient.

When a ESP packet is received from WAN network following processing steps will be done:

1. Upon receipt of a ESP packet from the WAN network, the SPINAT determines the appropriate translation state, based on the SPI. If no valid state exists for this SPI the SPINAT node MUST discard the packet.

2. The SPINAT node translates the destination address and the SPI based on the information in the stored in to the state.

3. SPINAT node forwards the packet on to the private network.

4.2.2. SPI Translation at End-hosts

This section describes the packet processing of ESP packets in the end-hosts.

The client node processes the inbound and outbound packets as defined in sections 3.3 and 3.4 of [RFC4303] and in the client host there is no SPI translation takes place.

The peer node processes the inbound packets as defined in section 3.4 of [RFC4303] and no SPI translation takes place. The outbound packets are first processed as defined in section 3.3 of [RFC4303]. When the ESP payload has been constructed, the SPI will be translated to the one selected by SPINAT node before sending the packet to the client node.

5. Parameters and packet formats

The control signaling protocol packet carry the SPI value selected by SPINAT node.

5.1. New Parameters

5.1.1. NAT_ESP_INFO

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Type              |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Reserved                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Original SPI                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Translated SPI                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
Type           XXX
Length         12
Original SPI   Original SPI selected by the peer which is associated
               with peer's address(es) to this SA.
Translated SPI Translated SPI for data sent to address(es) associated
               with this SA.
```

5.2. HIP ESP Security Association Setup

The ESP Security Association is set up during the base exchange. The following subsections define the ESP SA setup procedure both using base exchange messages (I2) and using UPDATE messages.

5.2.1. Modifications in I2

When HIP is used with ESP, the I2 packet MUST carry an ESP_INFO parameter. Intermediate SPINAT nodes MUST add NAT_ESP_INFO parameter to the I2 packet. The packet is signed for the benefit of the intermediate SPINAT nodes to be able to verify the origin of the packet.

The NAT_ESP_INFO contains the translated SPI for this association as well as the sender's original SPI.

The following figure shows the contents of a I2 packet after it has passed the SPINAT node processing.

The HIP parameters for the I2 packet:

```
IP ( HIP ( ESP_INFO,
           [R1_COUNTER,]
           SOLUTION,
           DIFFIE_HELLMAN,
           HIP_TRANSFORM,
           ESP_TRANSFORM,
           ENCRYPTED { HOST_ID },
           [ ECHO_RESPONSE ,]
           HMAC,
           HIP_SIGNATURE
           [, ECHO_RESPONSE],
           NAT_ESP_INFO ) )
```

5.3. HIP ESP Rekeying

In this section, the procedure for rekeying an existing ESP SA is presented. Only the first packet of the UPDATE packet exchange is modified by the SPINAT node.

5.3.1. Initializing Rekeying

When HIP is used with ESP, the UPDATE packet is used to initiate rekeying. The UPDATE packet that initiates the rekeying MUST carry an ESP_INFO and MAY carry a DIFFIE_HELLMAN parameter.

Intermediate SPINAT nodes will have to inspect HIP UPDATE packets. Those that carry rekeying information the SPINAT node MUST add NAT_ESP_INFO parameter. The packet is signed for the benefit of the intermediate SPINAT nodes to be able to verify the origin of the packet.

The following figure shows the contents of a rekeying initialization UPDATE packet after it has passed the SPINAT node processing.

The HIP parameters for the UPDATE packet initiating rekeying:

```
IP ( HIP ( ESP_INFO,
           SEQ,
           [DIFFIE_HELLMAN, ]
           HMAC,
           HIP_SIGNATURE,
           NAT_ESP_INFO ) )
```

6. Security Considerations

The translated SPI values included in the key-exchange messages and ESP headers are not integrity protected with signatures or HMAC computation. Therefore, a Man-in-the-Middle (MitM) attacker MAY change the SPIs in the packets. However, the SPI value is only an index to a specific IPsec Security Association (SA) at the receiving party. The actual security is based on the shared session keys. Hence, an SPI changing attack does not affect the confidentiality or integrity properties of the protocol.

It must be noted that changing SPI values is only possible for an on-path attacker that is able to modify packets on the fly. Such an attacker is not only able to change the SPI values, but he can block all communications between the parties. Therefore, having unsigned and changeable SPIs does not introduce new security vulnerabilities to ESP. A host trusts a SPINAT device to change the SPI values in the same way it trusts the NAPT to change the port values.

7. IANA Considerations

8. Acknowledgments

A number of people have contributed to the text and ideas. The list of these people include Pekka Nikander, Petri Jokela, Raimo Vuopionperae, Yuri Ismailov, Jan Hoeller and Hannes Tschofenig. Our apologies to anyone whose name is missing.

9. References

9.1. Normative References

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

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

9.2. Informative References

[RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, January 2001.

Authors' Addresses

Jan Melen
Ericsson Research NomadicLab
JORVAS FIN-02420
FINLAND

Phone: +358 9 299 1
Email: jan.melen@nomadiclab.com

Jukka Ylitalo
Ericsson Research NomadicLab
JORVAS FIN-02420
FINLAND

Phone: +358 9 299 1
Email: jukka.ylitalo@nomadiclab.com

Patrik Salmela
Ericsson Research NomadicLab
JORVAS FIN-02420
FINLAND

Phone: +358 9 299 1
Email: patrik.salmela@nomadiclab.com

Full Copyright Statement

Intellectual Property

Acknowledgment

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