Encapsulating ESP in UDP for Load-balancing

draft-xu-ipsecme-esp-in-udp-lb-00

Xiaohu Xu (Huawei)

Dacheng Zhang (Huawei)

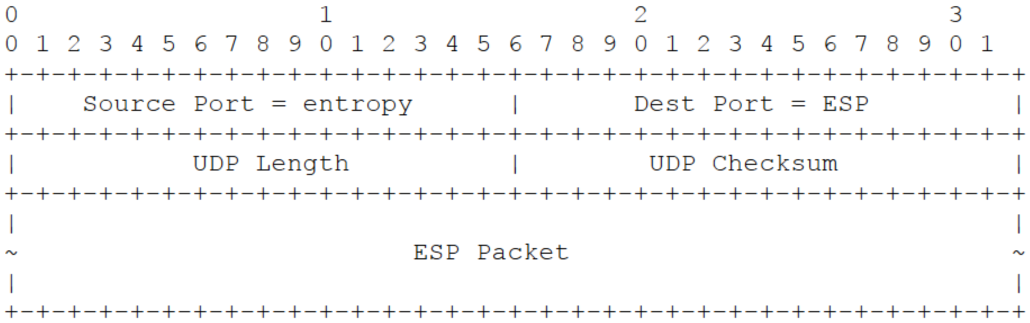
Liang Xia (Huawei)

IETF97, Seoul

Motivations

- IPsec Virtual Private Network (VPN) is widely used by enterprises to interconnect their geographical dispersed branch office locations across IP Wide Area Network (WAN).
- To fully utilize the bandwidth available in IP WAN, load balancing of traffic between different IPsec VPN sites over Equal Cost Multi-Path (ECMP) and/or Link Aggregation Group (LAG) within IP WAN is much attractive to those enterprises that deploy IPsec VPN solutions.
- Since most existing core routers within IP WAN can already support balancing IP traffic flows based on the hash of the five-tuple of UDP packets, by encapsulating IPsec Encapsulating Security Payload (ESP) packets inside UDP packets with the UDP source port being used as an entropy field, it will enable existing core routers to perform efficient load-balancing of the IPsec tunneled traffic without requiring any change to them.

ESP-in-UDP Encapsulation Format



```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Source Port = entropy     |       Dest Port = ESP         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           UDP Length          |        UDP Checksum           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                          ESP Packet                           ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- Source Port of UDP
  - This field contains a 16-bit entropy value that is generated by the encapsulator to uniquely identify a flow. What constitutes a flow is locally determined by the encapsulator and therefore is outside the scope of this document. What algorithm is actually used by the encapsulator to generate an entropy value is outside the scope of this document. In case the tunnel does not need entropy, this field of all packets belonging to a given flow SHOULD be set to a randomly selected constant value so as to avoid packet reordering.

ESP-in-UDP Encapsulation Format (con’t)

- Destination Port of UDP
  - This field is set to a value (TBD) indicating the encapsulated payload in the UDP header is an ESP packet.
- UDP Length
  - The usage of this field is in accordance with the current UDP specification [RFC768].
- UDP Checksum
  - For IPv4 UDP encapsulation, this field is RECOMMENDED to be set to zero for performance or implementation reasons because the IPv4 header includes a checksum and use of the UDP checksum is optional with IPv4. For IPv6 UDP encapsulation, the IPv6 header does not include a checksum, so this field MUST contain a UDP checksum that MUST be used as specified in [RFC0768] and [RFC2460] unless one of the exceptions that allows use of UDP zero-checksum mode [RFC6935] applies.

Clarifications

- The difference between the ESP-in-UDP encapsulation as proposed in this document and the ESP-in-UDP encapsulation as described in [RFC3948] is that the former uses the UDP tunnel for load-balancing improvement purpose and therefore the source port is used as an entropy field while the latter uses the UDP tunnel for NAT traverse purpose and therefore the source port is set to a constant value (i.e., 4500).
- In addition, the document only discusses about the tunnel mode ESP encapsulation.

Next-steps

- WG adoption?