Traffic visibility using IPsec ESP NULL encryption

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026 [i].

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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.”

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt
The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

This document describes leveraging UDP encapsulation for IPsec, using ESP NULL encryption in order for intermediary devices to inspect the IPsec packets. Currently in the IPsec standard, there is no way to
differentiate between ESP encryption and ESP NULL encryption by simply examining a packet.

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 RFC-2119 [ii].

Table of Contents

1. Introduction...................................................2
   1.1 Applicability Statement....................................3
2. UDP Encapsulation of IPsec ESP...............................3
   2.1 Extension Header...........................................4
   2.2 Tunnel and Transport mode of considerations...............5
   2.3 Port conflicts............................................5
3. IANA Considerations............................................6
4. Security Considerations.......................................6
References..................................................................6
Acknowledgments.....................................................7
Author’s Addresses................................................7

1. Introduction

The RFCs for leveraging ESP within IPsec describes how ESP packet encapsulation is performed. Other RFCs describe how ESP can be leveraged using NULL encryption, while preserving data integrity and authenticity. However, the exact encapsulation employed and the algorithms employed are negotiated out-of-band using the Internet-Key-Exchange (IKE) protocol. Once the packet is formatted and sent on the wire, it is infeasible to distinguish if encryption has been employed or is absent (ESP-NULL) by purely examining the packet.

In the case of employing IPsec within the Enterprise environment, it is desirable for intermediate devices (such as load balancers, Intrusion Detection / Prevention Systems (IDS/IPS)) to have access to the data within each packet to preserve existing critical network services. In a mixed mode environment, where some packets containing sensitive data may employ a given encryption cipher suite, while other packets are employing ESP-NULL, the intermediate devices is unable to discern which packets are leveraging ESP-NULL, hence inhibiting further analysis on that packet. This document describes a mechanism leveraging UDP-encapsulation of IPsec ESP packets using a fixed destination port, allowing the intermediate device to differentiate between encrypted data and NULL encrypted data for ESP.
1.1 Applicability Statement

The document is applicable to IPsec ESP only and does not describe any changes to IPsec AH.

2. UDP Encapsulation of IPsec ESP

UDP encapsulation of IPsec ESP packets is defined in RFC 3948 and takes the following basic 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Source Port            |      Destination Port         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Length              |           Checksum            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      ESP header [RFC2406]                     |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

According to RFC 3948, the source / destination port values are as the same as used by IKE.

We extend this to include a discrete destination port (value TBD) which identifies the UDP/ESP payload as accessible for intermediate devices. The source UDP port must be as used by IKE and does not change from the above specification. Having a reserved, unique destination port to identify the payload as decipherable by intermediate devices provides flexibility in adding an additional, unique header following the UDP header which allows the intermediate device to parse the packet according to additional hints on how the packet has been encoded. This is needed to pass information within each packet on the algorithm employed for the data authenticity and hence any IV requirements for that particular algorithm. As different algorithms may have differing IV requirements, this extension allows the intermediate device to take into account IV (/ICV) for a given algorithm and parse the remaining data pertaining to the upper layer protocol. This extension encoding is a fixed size and encodes information about the specific data authenticity algorithm used for the given packet / SA, the offset to the upper layer protocol and the upper layer protocol value. Diagrammatically, this may be depicted as follows.
The attributes in the extension header are described further below.

### 2.1 Extension Header

The extension header is exactly 32-bits, where the first 8-bits are used to convey the upper layer protocol being carried in the ESP payload. The value of this field is copied directly from the Next Header field of the ESP trailer and can be used by the intermediate device to determine/p Hale the upper layer protocol without having to find and parse the ESP trailer. The second 8-bits are used to convey the offset of the upper layer protocol from the end of this extension header (described further below). The third 8 bits are reserved for future expansion and set to zero. The last 8-bits contain the Algorithm Encoding and carries information about the algorithm being used to compute the ICV. This extension is needed in order for the intermediate device to determine which authentication algorithm is being used for generation of the ESP Integrity Check Value (ICV) and further parse the packet to extract the data portion. The size of the IV and ICV in the IPsec packet is algorithm dependent.

In this document, we do not define explicit values for the Algorithm Encoding, but choose to reuse the same values defined in various IPsec RFCs which describe how to negotiate a given algorithm using IKE. In this manner, this specification will be future proofed for any new algorithm definitions. For example, RFC 4302, section 3.3.2 defines specific values for the integrity algorithms, which are used within IKE. These are reserved for IKE via IANA. Additional RFCs defining other (newer) algorithms build upon these definitions and define new values for these algorithms. One example is RFC 4543, which describes usage of AES-GMAC within IPsec and hence defines the values used for different AES key sizes in section 9. The algorithm encoding is also useful in determining the size of the ICV for a...
given algorithm in order to deterministically extract the upper layer payload.

The offset is an 8-bit value, which defines the number of octets between the end of this extension header and the start of the upper layer protocol. This includes the ESP header, any additional IP header for tunnel mode and also the size of the IV, which may be algorithm dependent. Having an explicit value for the offset in the packet allows the intermediate device to directly parse past any algorithm dependent structures within the packet and reach the upper layer protocol header.

The reserved 8-bits are used to pad this extension to a long word alignment. This should be set to 0 by the sender, but it is not mandatory for the recipient to validate this value.

2.2 Tunnel and Transport mode of considerations

This extension is equally applicable for tunnel and transport mode where the ESP Next Header field is used to differentiate between these modes, as per the IPsec specifications.

2.3 Port conflicts

Another consideration is that a legacy client may choose the UDP port reserved for this specification as a random source port for a totally different protocol exchange. Although this should not happen if the client is choosing ports from the dynamic range specified by IANA, it is still possible and hence the responsibility rests on the intermediate device to ensure it can differentiate between the two cases. The intermediate device can ensure that it is looking at ESP-NONE traffic that is UDP encapsulated in this form by validating additional data elements following the UDP header. The format of the UDP extension described above can be validated. If this is deemed insufficient, then as a process of extracting the upper layer payload from the ESP encapsulated packet, the ESP trailer needs to be examined (this will be at a fixed place in the packet, proceeding the ICV value) and can be validated according to IPsec ESP trailer construction, which may include some padding octets. Furthermore, the intermediate device can now validate that the upper layer protocol begins after a fixed length following the UDP header (UDP extension + ESP header). Additionally, if the upper layer protocol contains a checksum, the intermediate device can further validate the checksum to ensure that packet construction is as expected. Validating these additional elements reduces the probability of any random payload for a UDP exchange where the source port is the same as this specification from being treated as an ESP encapsulated payload. These checks are not mandatory, but should be performed to cater for
this exception case. The extent and number of additional the checks performed are protocol dependent.

3. IANA Considerations

Reserving an appropriate value for the UDP destination port in order to provide this encapsulation is TBD and can happen after further peer review of this document.

4. Formal Syntax

The following syntax specification uses the augmented Backus-Naur Form (BNF) as described in RFC-2234 [iii].

There is no new syntax defined by this document.

Security Considerations

As this document augments the UDP encapsulation definitions specified in RFC 3948, the security observations made in that document also apply here. In addition, as this document promotes intermediate device visibility into IPsec ESP encapsulated frames for the purposes of Network monitoring functions, care should be taken not to send sensitive data over connections using definitions from this document. A strong key agreement protocol, such as IKE, together with a strong policy engine should be used to in determining appropriate security policy for the given traffic streams and data over which it is being employed.

References


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

Acknowledgments

Author’s Addresses

Ken Grewal
Intel Corporation
2111 NE 25th Avenue
JF3-232
Hillsboro, OR 97124
USA
Email: ken.grewal@intel.com

David Durham
Intel Corporation
2111 NE 25th Avenue
JF3-232
Hillsboro, OR 97124
USA
Email: david.durham@intel.com
Men Long
Intel Corporation
2111 NE 25th Avenue
JF3-232
Hillsboro, OR 97124
USA
Email: men.long@intel.com