Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: December 25, 2008

K. Grewal
Intel Corporation
G. Montenegro
Microsoft Corporation

June 23, 2008

XESP for Traffic Visibility
draft-grewal-ipsec-traffic-visibility-01

Status of this Memo

Abstract

This document describes an ESP encapsulation for IPsec, allowing intermediate devices to ascertain if ESP-NULL is being employed and hence inspect the IPsec packets for network monitoring and access control functions. Currently in the IPsec standard, there is no way to differentiate between ESP encryption and ESP NULL encryption by simply examining a packet.

1. Introduction

Use of ESP within IPsec [RFC4303] specifies how ESP packet

encapsulation is performed. It also specifies that ESP can use NULL encryption [RFC2410] while preserving data integrity and authenticity. The exact encapsulation and algorithms employed are negotiated out-of-band using, for example, IKE [RFC2409] or IKEv2 [RFC4306] and based on policy.

Enterprise environments typically employ numerous security policies (and tools for enforcing them), as related to access control, firewalls, network monitoring functions, deep packet inspection, Intrusion Detection and Prevention Systems (IDS and IPS), scanning and detection of viruses and worms, etc. In order to enforce these policies, network tools and intermediate devices require visibility into packets, ranging from simple packet header inspection to deeper payload examination. Network security protocols which encrypt the data in transit prevent these network tools from performing the aforementioned functions.

When employing IPsec within an enterprise environment, it is desirable to employ ESP instead of AH [RFC4302], as AH does not work in NAT environments. Furthermore, in order to preserve the above network monitoring functions, it is desirable to use ESP-NULL. In a mixed mode environment some packets containing sensitive data employ a given encryption cipher suite, while other packets employ ESP-NULL. For an intermediate device to unambiguously distinguish which packets are leveraging ESP-NULL, they would require knowledge of all the policies being employed for each protected session. This is clearly not practical. Heuristic-based methods can be employed to parse the packets, but these can be very expensive, containing numerous rules based on each different protocol and payload. Even then, the parsing may not be robust in cases where fields within a given encrypted packet happen to resemble the fields for a given protocol or heuristic rule. This is even more problematic when different length Initialization Vectors (IVs), Integrity Check Values (ICVs) and padding are used for different security associations, making it difficult to determine the start and end of the payload data, let alone attempting any further parsing. Furthermore, storage, lookup and cross-checking a set of comprehensive rules against every packet adds cost to hardware implementations and degrades performance. In cases where the packets may be encrypted, it is also wasteful to check against heuristics-based rules, when a simple exception policy (e.g., allow, drop or redirect) can be employed to handle the encrypted packets. Because of the non-deterministic nature of heuristics-based rules for disambiguating between encrypted and non-encrypted data, an alternative method for enabling intermediate devices to function in encrypted data environments needs to be defined. Enterprise environments typically use both stateful and stateless packet inspection mechanisms. The previous considerations weigh particularly heavy on stateless mechanisms such as router ACLs

and NetFlow exporters.

This document defines a mechanism to prove additional information in relevant IPsec packets so intermediate devices can efficiently differentiate between encrypted ESP packets and ESP packets with NULL encryption.

The document is consistent with the operation of ESP in NAT environments [RFC3947].

The design principles for this protocol are the following:

- o Allow easy identification and parsing of integrity-only IPsec traffic
- o Leverage the existing hardware IPsec parsing engines as much as possible to minimize additional hardware design costs
- o Minimize the packet overhead in the common case

1.1. Requirements Language

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 [RFC2119].

1.2. Applicability Statement

The document is applicable only to the Extended ESP header defined below, and does not describe any changes to either ESP [RFC4303] nor AH [RFC4302].

2. Extended ESP (XESP) Header format

The proposal is to define an Extended ESP protocol number, which provides additional attributes in each packet. The value of the new protocol is TBD and the format of the new encapsulation is defined below.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Next Header   |  HdrLen       | TrailerLen    | Flags         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Security Parameters Index (SPI)              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Sequence Number                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      IV (variable)                            |
~                                                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Payload Data                             |
~                                                               ~
|                                                               |
+               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               |            TFC Padding * (optional, variable) |
+-+-+-+-+-+-+-+-+               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                 | Padding (variable)          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Padding (0-255 bytes)          |PAD Length     | Next Header   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Integrity Check Value-ICV (variable)             |
~                                                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

XESP Header

Figure 1

Where:

Next Header: next protocol header (encrypted in ESP trailer, but in the clear in header), providing easy access to a HW parser to extract the upper layer protocol. Note: For security concerns, this value may optionally be set to zero, in which case the next header can be extracted from the ESP trailer.

HdrLen: includes the new header + full ESP header + the IV (if present). It is an offset to the beginning of the Payload Data.

TrailerLen: Offset from the end of the packet including the ICV, pad length, and any padding. It is an offset from the end of the packet to the last byte of the payload data.

Flags

2 bits: Version

1 bit: IntegrityOnly: Payload Data is not encrypted (ESP-NULL).

5 bits: reserved for future use. These MUST be set to zero per this specification, but usage may be defined by other specifications.

As can be seen, this Extended ESP format simply extended the standard ESP header by the first 4 octets.

2.1. UDP Encapsulation

This section describes a mechanism for running the new packet format over the existing UDP encapsulation of ESP as defined in RFC 3948. This allows leveraging the existing IKE negotiation of the UDP port for NAT-T discovery and usage [RFC3947], as well as preserving the existing UDP ports for ESP (port 4500). With UDP encapsulation, the packet format can be depicted as follows.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Src Port (4500)        | Dest Port (4500)              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Checksum                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Protocol Identifier (value = 0x00000001)             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Next Header   |  HdrLen       | TrailerLen    | Flags         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Security Parameters Index (SPI)              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Sequence Number                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      IV (variable)                            |
~                                                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Payload Data                             |
~                                                               ~
|                                                               |
+               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               |            TFC Padding * (optional, variable) |
+-+-+-+-+-+-+-+-+               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               | Padding (0-255 bytes)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Padding (0-255 bytes)          |PAD Length     | Next Header   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Integrity Check Value-ICV (variable)             |
~                                                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

UDP-encapsulated XESP Header

Figure 2

Where:

Source/Destination port (4500) and checksum: describes the UDP encapsulation header, per RFC3948.

Protocol Identifier: new field to demultiplex between UDP encapsulation of IKE, UDP encapsulation of ESP per RFC 3948, and this proposal.

According to RFC 3948, clause 2.2, a 4 octet value of zero (0) immediately following the UDP header indicates a Non-ESP marker, which can be used to assume that the data following that value is an IKE packet. Similarly, a value of non-zero indicates that the packet is an ESP packet and the 4-octet value can be treated as the ESP SPI. However, RFC 4303, clause 2.1 indicates that the values 1-255 are reserved and cannot be used as the SPI. We leverage that knowledge and use a value of 1 to indicate that the UDP encapsulated ESP header contains this new packet format for ESP encapsulation.

The remaining fields in the packet have the same meaning as per section 2.0 above.

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 existing IPsec specifications.

2.3. IKE Considerations

In order to negotiate the new format of ESP encapsulation via IKE, both sides of the security channel need to agree upon using the new packet format. This can be achieved by proposing a new protocol ID within the existing IKE proposal structure as defined by RFC 4306, clause 3.3.1. The existing proposal substructure in this clause allows negotiation of ESP/AH (among others) by using different protocol Ids for these protocols. By using the same protocol substructure in the proposal payload and using a new value (TBD) for this encapsulation, the existing IKE negotiation can be leverage with minimal changes to support negotiation of this encapsulation.

Furthermore, because the negotiation is at the protocol level, other transforms remain valid for this new encapsulation and consistent with IKEv2 [RFC4306]. Additionally, NAT-T [RFC3948] is wholly compatible with this extended frame format and can be used as-is, without any modifications, in environments where NAT is present and needs to be taken into account.

3. Acknowledgements

The authors would like to acknowledge the following people for their

feedback on updating the definitions in this document.

David McGrew, Brian Weis, Philippe Joubert, Brian Swander, Yaron Sheffer, Men Long, David Durham, Prashant Dewan, Marc Millier among others.

4. IANA Considerations

Reserving an appropriate value for this encapsulation as well as a new value for the protocol in the IKE negotiation is TBD by IANA.

5. Security Considerations

As this document augments the existing ESP encapsulation format, UDP encapsulation definitions specified in RFC 3948 and IKE negotiation of the new encapsulation, the security observations made in those documents also apply here. In addition, as this document allows 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, based on network domain/administrative policy. 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.

6. References

6.1. Normative References

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

[RFC2410] Glenn, R. and S. Kent, "The NULL Encryption Algorithm and Its Use With IPsec", RFC 2410, November 1998.

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

6.2. Informative References

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

[RFC3947] Kivinen, T., Swander, B., Huttunen, A., and V. Volpe,

"Negotiation of NAT-Traversal in the IKE", RFC 3947, January 2005.

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

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

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

Authors' Addresses

Ken Grewal
Intel Corporation
2111 NE 25th Avenue, JF3-232
Hillsboro, OR 97124
USA

Phone:
Email: ken.grewal@intel.com

Gabriel Montenegro
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
USA

Phone:
Email: gabriel.montenegro@microsoft.com

Full Copyright Statement

Intellectual Property