Network Working Group   K. Grewal
Internet Draft   Intel Corporation
Intended status: Standards Track   G. Montenegro
Expires: April 07, 2010   Microsoft Corporation
                                                          M. Bhatia
                                                      Alcatel-Lucent
                                                    October 07, 2009

                    Wrapped ESP for Traffic Visibility
                 draft-ietf-ipsecme-traffic-visibility-09.txt

Status of this Memo

   This Internet-Draft is submitted to IETF in full conformance
   with the provisions of BCP 78 and BCP 79. This document may
   contain material from IETF Documents or IETF Contributions
   published or made publicly available before November 10, 2008.
   The person(s) controlling the copyright in some of this material
   may not have granted the IETF Trust the right to allow
   modifications of such material outside the IETF Standards
   Process. Without obtaining an adequate license from the
   person(s) controlling the copyright in such materials, this
   document may not be modified outside the IETF Standards Process,
   and derivative works of it may not be created outside the IETF
   Standards Process, except to format it for publication as an RFC
   or to translate it into languages other than English.

   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.

   This Internet-Draft will expire on April 07, 2010.

Abstract

   This document describes the Wrapped Encapsulating Security
   Payload (WESP) protocol, which builds on the Encapsulating
   Security Payload (ESP) [RFC4303], and is designed to allow
   intermediate devices to (1) ascertain if data confidentiality is
   being employed within ESP and if not, (2) inspect the IPsec
   packets for network monitoring and access control functions.
   Currently in the IPsec ESP standard, there is no way to
   differentiate between encrypted and unencrypted payloads by
   simply examining a packet. This poses certain challenges to the
   intermediate devices that need to deep inspect the packet before
   making a decision on what should be done with that packet
   (Inspect and/or Allow/Drop). The mechanism described in this
   document can be used to easily disambiguate integrity-only ESP
   from ESP-encrypted packets, without compromising on the security
   provided by ESP.

Table of Contents

1. Introduction

   Use of ESP within IPsec [RFC4303] specifies how ESP packet
   encapsulation is performed. It also specifies that ESP can
   provide data confidentiality and data integrity services. Data
   integrity without data confidentiality ("integrity-only ESP") is
   possible via the ESP-NULL encryption algorithm [RFC2410] or via
   combined-mode algorithms such as AES-GMAC [RFC4543]. The exact
   encapsulation and algorithms employed are negotiated out-of-band
   using, for example, IKEv2 [RFC4306] and based on policy.

   Enterprise environments typically employ numerous security
   policies (and tools for enforcing them), as related to access
   control, content screening, 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
   integrity-only ESP. In a mixed-mode environment, some packets
   containing sensitive data employ a given encryption cipher
   suite, while other packets employ integrity-only ESP. For an
   intermediate device to unambiguously distinguish which packets
   are using integrity-only ESP requires knowledge of all the
   policies being employed for each protected session. This is
   clearly not practical. Heuristics-based methods can be employed
   to parse the packets, but these can be very expensive, requiring
   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. 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. Additionally there are many types and classes of
network devices employed within a given network and a
deterministic approach provides a simple solution for all of
them. 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. Nevertheless, a
deterministic approach provides a simple solution for the myriad
types of devices employed within a network, regardless of their
stateful or stateless nature.

This document defines a mechanism to provide additional
information in relevant IPsec packets so intermediate devices
can efficiently differentiate between encrypted and integrity-
only packets. Additionally and in the interest of consistency,
this extended format can also be used to carry encrypted packets
without loss in disambiguation.

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 wrapped ESP header
defined below, and does not describe any changes to either ESP
[RFC4303] nor IP Authentication Header (AH) [RFC4302].

There are two ways to enable intermediate security devices to
distinguish between encrypted and unencrypted ESP traffic:

- The heuristics approach [Heuristics I-D] has the intermediate

node inspect the unchanged ESP traffic, to determine with

extremely high probability whether or not the traffic stream is
encrypted.

- The Wrapped ESP (WESP) approach described in this document, in
contrast, requires the ESP endpoints to be modified to support
the new protocol. WESP allows the intermediate node to
distinguish encrypted and unencrypted traffic deterministically,
using a simpler implementation for the intermediate node.

Both approaches are being documented simultaneously by the IP
Security Maintenance and Extensions (IPsecME) Working Group,
with WESP being put on Standards Track while the heuristics
approach is being published as an Informational RFC. While
endpoints are being modified to adopt WESP, we expect both
approaches to coexist for years, because the heuristic approach
is needed to inspect traffic where at least one of the endpoints
has not been modified. In other words, intermediate nodes are
expected to support both approaches in order to achieve good
security and performance during the transition period.

2. Wrapped ESP (WESP) Header format

Wrapped ESP encapsulation (WESP) uses protocol number (TBD via
IANA). Accordingly, the (outer) protocol header (IPv4, IPv6, or
Extension) that immediately precedes the WESP header SHALL
contain the value (TBD via IANA) in its Protocol (IPv4) or Next
Header (IPv6, Extension) field. WESP provides additional
attributes in each packet to assist in differentiating between
encrypted and non-encrypted data, and to aid parsing of the
packet. WESP follows RFC 4303 for all IPv6 and IPv4
considerations (e.g., alignment considerations).

This extension essentially acts as a wrapper to the existing ESP
protocol and provides an additional 4 octets at the front of the
existing ESP packet.

This may 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Wrapped ESP Header                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Existing ESP Encapsulation                  |
~                                                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1 WESP Packet Format

By preserving the body of the existing ESP packet format, a
compliant implementation can simply add in the new header,
without needing to change the body of the packet. The value of
the new protocol used to identify this new header is TBD via
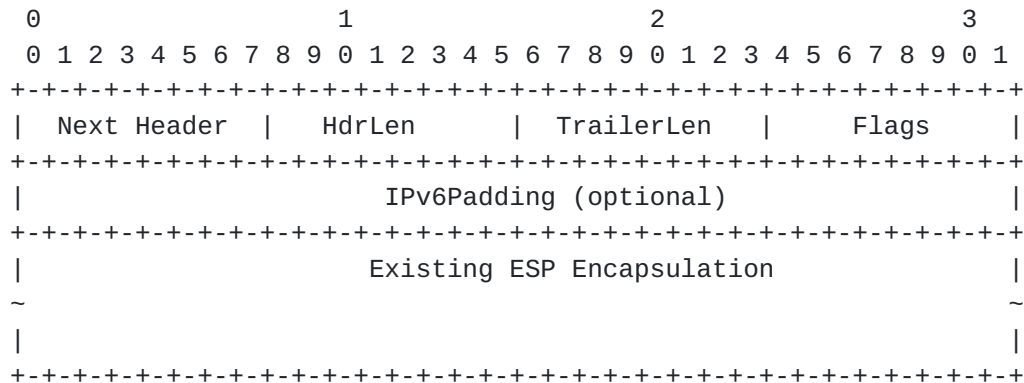IANA. Further details are shown below:

```
 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     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      IPv6Padding (optional)                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Existing ESP Encapsulation                 |
~                                                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2 Detailed WESP Packet Format

Where:

Next Header, 8 bits: This field MUST be the same as the Next
Header field in the ESP trailer when using ESP in the Integrity
only mode. When using ESP with encryption, the "Next Header"
field looses this name and semantics and becomes an empty field
which MUST be initialized to all zeros. The receiver MUST do
some sanity checks before the WESP packet is accepted. The
receiver MUST ensure that the Next Header field in the WESP
header and the Next Header field in the ESP trailer match when
using ESP in the Integrity only mode. The packet MUST be dropped
if the two do not match. Similarly, the receiver MUST ensure
that the Next Header field in the WESP header is an empty field
initialized to zero if using WESP with encryption. The WESP
flags dictate if the packet is encrypted.

HdrLen, 8 bits: Offset from the beginning of the WESP header to
the beginning of the Rest of Payload Data (i.e., past the IV, if
present) within the encapsulated ESP header, in octets. HdrLen
MUST be set to zero when using ESP with encryption. When using
integrity-only ESP, the following HdrLen values are invalid: any
value less than 12; any value that is not a multiple of 4; any
value that is not a multiple of 8 when using IPv6. The receiver
MUST ensure that this field matches with the header offset
computed from using the negotiated SA and MUST drop the packet
in case it does not match.

TrailerLen, 8 bits: TrailerLen contains the size of the ICV
being used by the negotiated algorithms within the IPsec SA.
TrailerLen MUST be set to zero when using ESP with encryption.
The receiver MUST only accept the packet if this field matches
with the value computed from using the negotiated SA. This
insures that sender is not deliberately setting this value to
obfuscate a part of the payload from examination by a trusted
intermediary device.

Flags, 8 bits: The bits are defined most-significant-bit (MSB)
first, so bit 0 is the most significant bit of the flags octet.

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|V V|E|X| Rsvd  |
+-+-+-+-+-+-+-+-+
```

Figure 3 Flags format

    Version (V), 2 bits: MUST be sent as 0 and checked by the
receiver. If the version is different than an expected version
number (e.g. negotiated via the control channel), then the
packet MUST be dropped by the receiver. Future modifications to
the WESP header may require a new version number. Intermediate
nodes dealing with unknown versions are not necessarily able to
parse the packet correctly. Intermediate treatment of such
packets is policy-dependent (e.g., it may dictate dropping such
packets).

    Encrypted Payload (E), 1 bit: Setting the Encrypted Payload
bit to 1 indicates that the WESP (and therefore ESP) payload is
protected with encryption. If this bit is set to 0, then the
payload is using integrity-only ESP. Setting or clearing this
bit also impacts the value in the WESP Next Header field, as
described above. The recipient MUST ensure consistency of this
flag with the negotiated policy and MUST drop the incoming
packet otherwise.

    Extended header (X), 1 bit: If set (value 1), the 4 octet padding
is present. If not set (value 0), the 4 octet padding is absent. This
padding SHOULD be used with IPv6 in order to preserve IPv6 8-octet
IPv6 alignment. This padding SHOULD NOT be used with IPv4, as it is
not needed to guarantee 4-octet IPv4 alignment.

    Rsvd, 4 bits: Reserved for future use. The reserved bits
MUST be sent as 0, and ignored by the receiver. Future documents
defining any of these bits MUST NOT affect the distinction
between encrypted and unencrypted packets. Intermediate nodes

dealing with unknown reserved bits are not necessarily able to

   parse the packet correctly. Intermediate treatment of such
   packets is policy-dependent (e.g., it may dictate dropping such
   packets).

   Future versions of this protocol may change the Version number
   and/or the reserved bits sent, possibly by negotiating them over
   the control channel.

   As can be seen, the WESP format extends the standard ESP header
   by the first 4 octets for IPv4 and by 8 octets for IPv6. The
   WESP header is integrity protected, along with all the fields
   specified for ESP in RFC 4303.

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, RFC4306],
   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)              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Length              |         Checksum              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Protocol Identifier (value = 0x00000002)              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Next Header  |  HdrLen       |  TrailerLen   |    Flags       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     IPv6Padding (optional)                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Existing ESP Encapsulation                 |
~                                                              ~
|                                                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
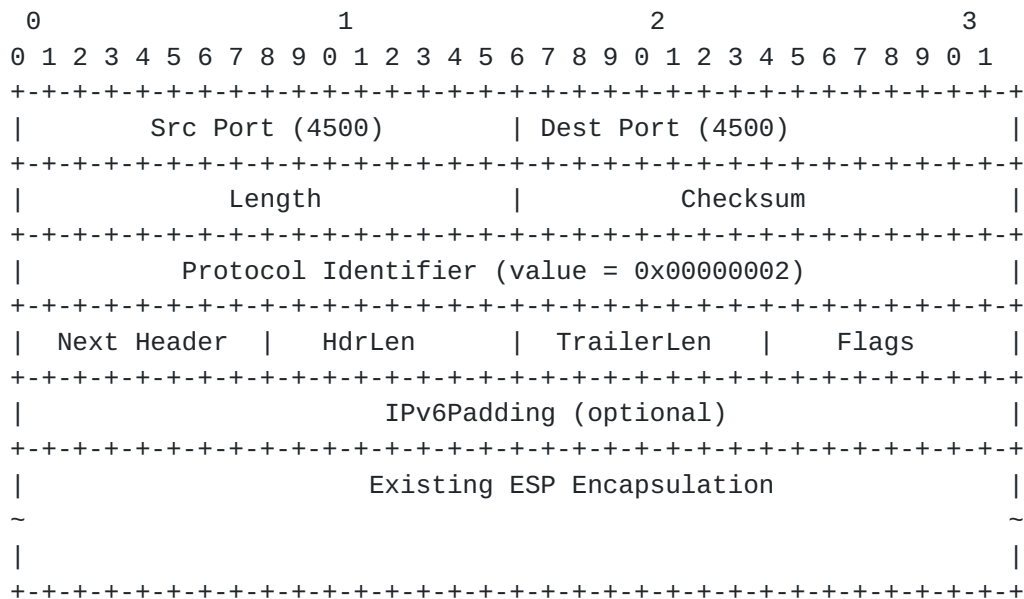
                Figure 4 UDP-Encapsulated WESP Header

   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
the UDP encapsulation in this specification.

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 greater then 255 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 one of these reserved
values 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 above.

2.2. Transport and Tunnel Mode Considerations

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

In the diagrams below, "WESP ICV" refers to the ICV computation as
modified by this specification. Namely, the ESP ICV computation is
augmented to include the four octets that constitute the WESP header.
Otherwise, the ICV computation is as specified by ESP [RFC4303].

2.2.1. Transport Mode Processing

In transport mode, ESP is inserted after the IP header and before a
next layer protocol, e.g., TCP, UDP, ICMP, etc. The following
diagrams illustrate how WESP is applied to the ESP transport mode for
a typical packet, on a "before and after" basis.

```
BEFORE APPLYING WESP - IPv4
       ----------------------------
      |orig IP hdr  |     |       |
      |(any options)| TCP | Data  |
       ----------------------------
```

AFTER APPLYING WESP - IPv4

```
        -----------------------------------------------------
        |orig IP hdr  | WESP | ESP |     |        |   ESP   |WESP|
        |(any options)| Hdr  | Hdr | TCP | Data | Trailer | ICV|
        -----------------------------------------------------
                                  |<---- encryption ---->|
                         |<---------- integrity ---------->|
```

BEFORE APPLYING WESP - IPv6

```
        -----------------------------------------
        | orig |hop-by-hop,dest*,|dest|    |    |
        |IP hdr|routing,fragment.|opt*|TCP|Data|
        -----------------------------------------
```

AFTER APPLYING WESP - IPv6

```
     ---------------------------------------------------------------
     | orig |hop-by-hop,dest*,|    |    |dest|    |    | ESP   |WESP|
     |IP hdr|routing,fragment.|WESP|ESP|opt*|TCP|Data|Trailer| ICV|
     ---------------------------------------------------------------
                                   |<---- encryption --->|
                          |<-------- integrity --------->|
```

```
            * = if present, could be before WESP, after ESP, or both
```

All other considerations are as per RFC 4303.

2.2.2. Tunnel Mode Processing

In tunnel mode, ESP is inserted after the new IP header and before
the original IP header, as per RFC 4303. The following diagram
illustrates how WESP is applied to the ESP tunnel mode for a typical
packet, on a "before and after" basis.

```
BEFORE APPLYING WESP - IPv4
      --------------------------
      | orig IP hdr*  |   |    |
      | (any options) |TCP|Data|
      --------------------------

AFTER APPLYING WESP - IPv4
      ------------------------------------------------------------------
      |new IP hdr*  |   |   | orig IP hdr*  |   |    | ESP   |WESP|
      |(any options)|WESP|ESP| (any options) |TCP|Data|Trailer| ICV|
      ------------------------------------------------------------------
                            |<--------- encryption --------->|
                   |<-------------- integrity ------------->|

BEFORE APPLYING WESP - IPv6
        --------------------------
        | orig*|orig ext |   |    |
        |IP hdr| hdrs *  |TCP|Data|
        --------------------------

AFTER APPLYING WESP - IPv6
----------------------------------------------------------------------
| new* |new ext |   |   | orig*|orig ext |   |    | ESP   |WESP|
|IP hdr| hdrs*  |WESP|ESP|IP hdr| hdrs *  |TCP|Data|Trailer| ICV|
----------------------------------------------------------------------
                         |<--------- encryption ---------->|
                |<-------------- integrity -------------->|
```

   * = if present, construction of outer IP hdr/extensions and

   modification of inner IP hdr/extensions is discussed in

   the Security Architecture document.

   All other considerations are as per RFC 4303.

2.3. IKE Considerations

   This document assumes that WESP negotiation is performed using
   IKEv2. In order to negotiate the new format of ESP encapsulation
   via IKEv2 [RFC4306], both parties need to agree to use the new
   packet format. This can be achieved using a notification method
   similar to USE_TRANSPORT_MODE defined in RFC 4306.

   The notification, USE_WESP_MODE (value TBD) MAY be included in a
   request message that also includes an SA payload requesting a

CHILD_SA using ESP. It requests that the CHILD_SA use WESP mode
rather than ESP for the SA created. If the request is accepted,
the response MUST also include a notification of type
USE_WESP_MODE. If the responder declines the request, the
CHILD_SA will be established using ESP, as per RFC 4303.   If
this is unacceptable to the initiator, the initiator MUST delete
the SA. Note: Except when using this option to negotiate WESP
mode, all CHILD_SAs will use standard ESP.

Negotiation of WESP in this manner preserves all other
negotiation parameters, including NAT-T [RFC3948]. NAT-T is
wholly compatible with this wrapped 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. 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 IKEv2, 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.

ESP is end-to-end and it will be impossible for the intermediate
devices to verify that all the fields in the WESP header are
correct. It is thus possible to modify the WESP header so that
the packet sneaks past a firewall if the fields in the WESP
header are set to something that the firewall will allow. The
endpoint thus must verify the sanity of the WESP header before
accepting the packet. In an extreme case, someone colluding with
the attacker, could change the WESP fields back to the original
values so that the attack goes unnoticed. However, this is not a
new problem and it already exists IPsec.

4. IANA Considerations

The WESP protocol number is assigned by IANA out of the IP
Protocol Number space (and as recorded at the IANA web page at
http://www.iana.org/assignments/protocol-numbers) is: TBD.

The USE_WESP_MODE notification number is assigned out of the "IKEv2 Notify Message Types - Status Types" registry's 16384-40959 (Expert Review) range: TBD.

The SPI value of 2 is assigned by IANA out of the reserved SPI range from the SPI values registry to indicate use of the WESP protocol within a UDP encapsulated, NAT-T environment.

This specification requests that IANA create a new registry for "WESP Flags" to be managed as follows:

The first 2 bits are the WESP Version Number. The value 0 is assigned to the version defined in this specification. Further assignments of the WESP Version Number are to be managed via the IANA Policy of "Standards Action" [RFC5226]. For WESP version numbers, the unassigned values are 1, 2 and 3. The Encrypted Payload bit is used to indicate if the payload is encrypted or using integrity-only ESP. The extended header bit is used to signal the use of padding required to preserve IPv6 alignment. The remaining 4 bits of the WESP Flags are undefined and future assignment is to be managed via the IANA Policy of "Specification Required".

5. Acknowledgments

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.

This document was prepared using 2-Word-v2.0.template.doc.

6. References

6.1. Normative References

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

[RFC4543] McGrew, D. and Viega J., "The Use of Galois Message Authentication Code (GMAC) in IPsec ESP and AH", RFC 4543, May 2006.

[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.

6.2. Informative References

   [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.

   [RFC5226] Narten, T., Alverstrand, H., "Guidelines for Writing
             an IANA Considerations Section in RFCs", RFC 5226,
             May 2008.

   [Heuristics I-D] Kivinen, T., McDonald, D., "Heuristics for Detecting
             ESP-NULL packets", Internet Draft, April 2009.

Author's 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

Manav Bhatia
Alcatel-Lucent
Bangalore
India

Phone:
Email: manav@alcatel-lucent.com