IPSECME D. Palomares (Ed)
Internet-Draft D. Migault (Ed)
Intended status: Informational Orange
Expires: September 6, 2014 March 5, 2014

IKEv2/IPsec Context Definition
draft-plmrs-ipsecme-ipsec-ikev2-context-definition-01

Abstract

IKEv2/IPsec clusters are constituted of multiple nodes accessed via a single address by the end user. The traffic is then split between the nodes via specific IP load balancing policies. Once a session is assigned to a given node, IPsec makes it difficult to assign the session to another node. This makes management operations and transparent high availability for end users difficult to perform within the cluster.

This document describes the IKEv2 and IPsec contexts that MUST be transferred between nodes within a cluster so a session can be restored. This makes possible to transfer an IPsec session between different nodes.

Status of This Memo

Copyright Notice

Table of Contents

1. Requirements notation

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].

2. Introduction

Large clusters may take advantage of the multiple nodes to enhance the peer's Quality of Service by performing among others:

1) Fail-over with high availability.

2) Load balancing among cluster members.

3) Scalability for overloaded IPsec platforms.

4) Compatibility for IKEv2/IPsec context transfers among different constructors.

This document addresses transfer of an IPsec session between physically or virtually different nodes within an IKEv2/IPsec cluster. More specifically, the document describes the parameters that MUST be transmitted between the IPsec/IKEv2 nodes, so that IKEv2 and IPsec session can be restored on the other node.

Currently IPsec based services can hardly benefit from these features as IPsec Security Associations are bound to a single node and cannot be shared among different cluster members.

This draft describes the parameters that MUST be transferred in order to keep an IKEv2/IPsec session alive in conformance with the Security Architecture for the Internet Protocol [RFC4301] and the Internet Key Exchange (IKEv2) Protocol [RFC5996].

This includes information such as the cryptographic material, the algorithms and the IP addresses, among others parameters.

Note that IKEv2 and IPsec session do not need to be on the same node as IKEv2 and IPsec context are different. Note also that we do not specify in this document how the IKEv2 or IPsec context are transferred between one node to the other. This can be performed via a simple UDP session that MAY be IPsec protected, a SCP session [RFC4251] or using the context transfer protocol [RFC4067].

3. Terminology

This document uses the following terminology:

IKE_SA context: the set of parameters composing a single IKE Security Association. A bidirectional communication will need a pair of IKE_SAs, for incoming and outgoing IKE exchanges.

IPsec_SA Context: the set of parameters composing a single IPsec Security Association. A bidirectional communication will need a pair of IPsec_SAs for incoming and outgoing traffic.

4. Parameters level definition

Information related to the IKEv2 and IPsec contexts can be defined within three different levels: mandatory, optional or vendor specific. This allows classification of the parameters considering

their relevance and susceptibility to be transferred in order to maintain an IKEv2/IPsec session alive.

1) Mandatory (M): Those parameters identified with a Mandatory flag (M) are considered absolutely relevant and necessary in order to maintain an IKE_SA or an IPsec_SA alive. The absence of a parameter with a mandatory flag, results in the loss of the IKE_SA or IPsec_SA.

2) Optional (O): Those parameters identified with an optional flag (O) are considered as additional information but are NOT absolutely necessary to maintain an IKEv2/IPsec session alive.

3) Vendor Specific (V): Those parameters identified with a vendor's specific flag (V) are considered as the information related to some specific constructor. It ensures enhancement provided by certain proprietary solutions when transmitting IKEv2/IPsec contexts, however, this MUST NOT interfere the interoperability with other IKEv2 and IPsec implementations and standards.

5. IKEv2 key management

Implementations might decide to manage sending cryptographic material (a.k.a. IKEv2/IPsec session keys) in different fashions; especially IKEv2 session keys. This document specifies three different ways to exchange IKEv2 keying information as follows:

1) Case 1: The node sends the private Diffie-Hellman key, the peer's KE content and nonces. In this case, the node receiving these information will recalculate all keys from the very beginning as it usually does during any initial IKEv2 exchange. The main drawback for this case is that recalculating keys is computational expensive, especially if thousands of session keys has to be calculated (e.g. during rush hours).

2) Case 2: A cluster member sends the SKEYSEED and nonces. In such case, the node receiving the information might not recalculate all the keys since the very beginning, but it still has to compute SK_* (SK_d, SK_ai, SK_ar, SK_ei, SK_er, SK_pi, SK_pr).

3) Case : The cluster member sends all computed keys (SK_* = SK_d, SK_ai, SK_ar, SK_ei, SK_er, SK_pi, SK_pr). In this case, the node receiving the keys wont need to recalculate keys from the beginning. However, this case demands more data to be sent between cluster members. Note that sending SK_pi/SK_pr may be omitted, as these keys are only used during authentication.

6. IKEv2 Session parameters

Considering IKEv2/IPsec sessions as bidirectional, we provide a list of parameters needed to create the IKE_SAs, which are usually stored in the user-land.

6.1. MANDATORY - IKEv2 Session parameters

1) Version of IKE: in this draft we only consider version 2.

2) The initiator flag and the responder flag for the IKE_SAs.

3) Local host address and remote host address (IPv4 or IPv6).

4) The IKE_SA's SPI of both initiator and responder.

5) The outgoing and incoming Message ID's.

7) The cryptographic material for the IKE_SA (see section Section 5 for details).

8) The [SA] proposal information: encryption algorithm, length of the encryption key, integrity algorithm, length of the integrity key and the pseudo random function (prf).

9) The extensions and condition of the IKE_SA (NAT, EAP, MOBIKE...).

10) The IDs of the initiator and responder (ID_IPV4_ADDR, ID_IPV6_ADDR, ID_FQDN, ID_RFC822_ADDR, ID_DER_ASN1_DN, ID_DER_ASN1_GN or ID_KEY_ID).

11) Credentials: pre-shared keys or digital certificates.

12) The windows bitmap value.

6.2. OPTIONAL - IKEv2 Session parameters

1) The IKE lifetime.

2) Vendors ID: when a vendors ID payload has been sent during IKE_SA negotiation, it is part of the IKE_SA parameters.

6.3. VENDOR SPECIFIC - IKEv2 Session parameters

For now, there are no vendor specific parameters for IKEv2.

7. IPsec Session parameters

Once the IKE_SAs are established for securing further IKEv2 exchanges, a pair of IPsec_SAs are negotiated in order to secure the traffic flow. The following list includes the parameters needed to build an IPsec_SA:

7.1. MANDATORY - IPsec Session parameters

1) Local host and remote host addresses (IPv4 or IPv6).

2) The inbound and outbound IPsec_SA Security Parameter Indexes (SPIs).

3) The IP compression information: flag for IPcomp. If active, The IPcomp Compression Parameter Index values (CPI IN, CPI OUT) and the the IPcomp algorithm.

4) The sequence number values: SN counter and SN overflow flag

5) The anti-replay window value.

6) IPsec mode: transport or tunnel mode.

7) The SA Lifetime: a time interval or byte count after which an SA must be replaced with a new SA (and new SPI).

8) Path MTU: maximum size of an IPsec packet that can be transmitted without fragmentation.

9) Upperspec: upper-layer protocol to be used.

10) Source IP/Destination IP addresses and ports of the protected traffic.

11) The IPsec protocol ESP and/or AH, their encryption/integrity algorithms and the key lengths.

12) The cryptograhic material: KEYMAT (encryption and/or authentication keys).

7.2. OPTIONAL - IPsec Session parameters

For now, there are no optional parameters for IPsec sessions.

7.3. VENDOR SPECIFIC - IPsec Session parameters

1) Instance-id or flow-id: helps a node to identify which packet processing unit will process some IPsec traffic or which IPsec instance out of multiple IPsec processing units will process the IPsec traffic.

8. IANA Considerations

There are no IANA consideration for this document.

9. Security Considerations

Transferring an IPsec context between different SG involves sending sensitive information through the network. These pieces of information MUST be sent to an authenticated node via a secure channel.

10. Acknowledgment

IPsec cluster management is a joint work between Orange, Universite Pierre et Marie Curie / LIP6 and Institut Telecom / Telcom SudParis.

We would like to thank Maryline Laurent and Tobias Guggemos for their advises.

11. References

11.1. Normative References

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

[RFC4251] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Protocol Architecture", RFC 4251, January 2006.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC5996] Kaufman, C., Hoffman, P., Nir, Y., and P. Eronen, "Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 5996, September 2010.

11.2. Informative References

[RFC4067] Loughney, J., Nakhjiri, M., Perkins, C., and R. Koodli, "Context Transfer Protocol (CXTP)", RFC 4067, July 2005.

11.3. URIs

[1] http://tools.ietf.org/html/draft-plmrs-ipsecme-ipsec-ikev2
    -context-definition-01

[2] http://tools.ietf.org/html/draft-plmrs-ipsecme-ipsec-ikev2
    -context-definition-00

Appendix A. ANNEX A: Data structure example

Example of an IKEv2 data structure:

```
  typedef struct _IKEV2CONTEXT
            {
                 bool *initiator;
                 u_int32_t *ike_spi_i;
                 u_int32_t *ike_spi_r;
                 char *my_host;
                 char *other_host;
                 u_int16_t *enc_alg_ike;
                 u_int16_t *enc_alg_ike_len;
                 u_int16_t *int_alg_ike;
                 u_int16_t *prf_alg;
                 char *nonce_i;
                 char *nonce_r;
                 char *dh_secret;
                 u_int16_t message_id;
                 char *cert;
        } IKEV2CONTEXT;
```

Example of an IPsec session data structure:

```
     typedef struct _IPSECCONTEXT
               {
                    bool initiator;
                    char *my_host;
                    char *other_host;
                    u_int8_t ipsec_mode;
                    u_int16_t encr_alg_child;
                    u_int16_t enc_alg_len_child;
                    u_int16_t int_alg_child;
                    u_int32_t enc_key_i;
                    u_int32_t int_key_i;
                    u_int32_t enc_key_o;
                    u_int32_t int_key_o;
                    char *child_seq_i;
                    char *child_bit_i;
                    char *child_seq_o;
                    char *child_bit_o;
                    char *child_spi_i;
                    char *child_spi_o;
                    u_int16_t ts_l_fromport;
                    u_int16_t ts_l_toport;
                    u_int8_t ts_l_type;
                    u_int8_t ts_l_proto;
                    char *ts_l_fromaddress;
                    char *ts_l_toaddress;
                    u_int16_t ts_r_fromport;
                    u_int16_t ts_r_toport;
                    u_int8_t ts_r_type;
                    u_int8_t ts_r_proto;
                    char *ts_r_fromaddress;
                    char *ts_r_toaddress;
                    bool ipcomp_flag;
                    u_int32_t ipcom_algo;
                    char *ipcomp_cpi_i;
                    char *ipcomp_cpi_o;
           } IPSECCONTEXT;
```

Appendix B. Document Change Log

[RFC Editor: This section is to be removed before publication]

draft-plmrs-ipsecme-ipsec-ikev2-context-definition-01 [1]
Added missing information as part of the IPsec and IKEv2 contexts
Worked on the text
Include mandatory, optional and vendor specific flags
Added three different ways send keys session keys

draft-plmrs-ipsecme-ipsec-ikev2-context-definition-00 [2]

initial draft.

Authors' Addresses

Daniel Palomares
Orange
38 rue du General Leclerc
92794 Issy-les-Moulineaux Cedex 9
France

Phone: +33 1 45 29 51 16
Email: danielpalomares.ietf@gmail.com

Daniel Migault
Orange
38 rue du General Leclerc
92794 Issy-les-Moulineaux Cedex 9
France

Phone: +33 1 45 29 60 52
Email: daniel.migault@orange.com