ipsecme
Internet-Draft
Intended status: Standards Track
Expires: September 6, 2018

D. Schinazi
Apple Inc.

March 5, 2018

Privacy Addition to the Internet Key Exchange Protocol Version 2 (IKEv2) IKE_SA_INIT Exchange

draft-dschinazi-ipsecme-sa-init-privacy-addition-00

Abstract

The Internet Key Exchange Protocol version 2 (IKEv2) provides strong security and privacy properties to both endpoints once they have authenticated each other. However, before an endpoint has validated the peer's AUTH payload, it could be divulging information to an untrusted host. An example of such information is the Identification payload of the initiator. Another example is the fact that a host is running an IKEv2 responder. This document introduces a new "Initialization Authentication Code" notify payload that can be included in IKE_SA_INIT messages to increase their trustworthiness. This new protection is meant to be used in addition to current IKEv2 mechanisms and is not meant to replace the AUTH payload in any way.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on September 6, 2018.

Copyright Notice

Table of Contents

1. Introduction

The Internet Key Exchange Protocol version 2 (IKEv2) [RFC7296] provides strong security and privacy properties to both endpoints once they have authenticated each other. However, before an endpoint has validated the peer's AUTH payload, it could be divulging information to an untrusted host. Examples include:

- The Identification payload of the initiator is sent with the initiator's first IKE_AUTH request. This payload can be used to track the owner of the device initiating IKE.

- Some IKEv2 servers may wish to hide their very existence to avoid being blacklisted by entities that resent the privacy properties an IKEv2/IPsec tunnel can provide to users. If the IKEv2 server is accessible over TLS on a TCP port [RFC8229] that is shared with another protocol, responding to the initiator's IKE_SA_INIT can disclose the server's existence.

This document introduces a new "Initialization Authentication Code" (IAC) notify payload that can be included in IKE_SA_INIT messages to increase their trustworthiness. This new protection is meant to be used in addition to current IKEv2 mechanisms and is not meant to replace the AUTH payload in any way.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

1.2. Terminology

This document uses the following terms:

Endpoint One of the two hosts that are involved in an IKE exchange.

Initiator The endpoint that sends the first IKE_SA_INIT request of the IKE exchange being discussed.

Responder The endpoint that is not the initiator.

Peer When discussing an endpoint, its peer is the other endpoint participating in the IKE exchange.

IASS Initialization Authentication Shared Secret, a shared secret included in the IKEv2 configuration. It is separate from any shared secret used for computation of the AUTH payload.

MAC Message Authentication Code, a cryptographic means of ensuring integrity and authenticity of a message.

IAC Initialization Authentication Code, a MAC of IKE_SA_INIT nonces with the IASS.

PRF Pseudo-Random Function, a function used to compute the IAC.

2. Attack Vectors

This document only attempts to address the following attack vectors.

2.1. On-Path Attacker Targeting Initiator

This attack vector assumes the presence of an active on-path attacker that can block and forge any packets between both endpoints. Without the mechanism described in this document, the attacker can forge an IKE_SA_INIT reply and get the initiator to send it its IKE_AUTH request encrypted with the ephemeral shared secret computed between the initiator and the attacker. This leaks the identity of the initiator (IDi) and can leak the identity of the responder (IDr) if the initiator also sent it.

2.2. Off-Path Attacker Targeting Responder

Some network middleboxes may wish block to block IKEv2 negotiation. This is often done by blocking UDP traffic which can be worked around using IKEv2 TCP encapsulation [RFC8229]. This obfuscation can even be improved by encapsulating IKEv2 and IPsec inside TLS. However, a more persevering middlebox can establish a TLS connection to the responder and try to send an IKE_SA_INIT to probe the server for IKEv2 support. Without the mechanism described in this document, the responder has to send an IKE_SA_INIT reply before it's established any initiator identity, leaking the presence of the IKEv2 server.

3. Initialization Authentication

3.1. Computing Initialization Authentication

Each endpoint configuration will include both an IASS and a PRF for this endpoint, and also IASS and PRF of the peer. It will commonly be the case (but it is not required) that the same IASS and the same PRF is used in both directions.

The peers authenticate the IKE_SA_INIT messages by having each MAC nonces using a padded shared secret as the key. The IAC is computed as follows:

IAC_i = prf_i(prf_i(IASS_i, "Initialization Authentication Key Pad for IKEv2 Initiator"), Ni)

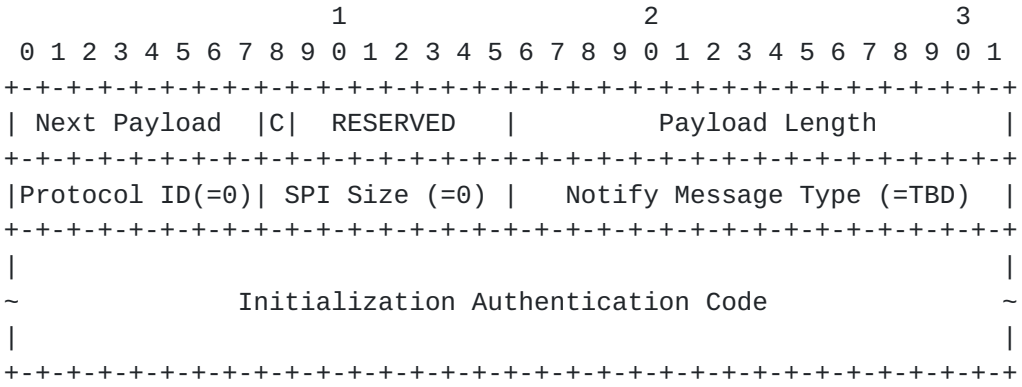
IAC_r = prf_r(prf_r(IASS_r, "Initialization Authentication Key Pad for IKEv2 Responder"), Ni | Nr)

Where IAC_i and IAC_r are the Initialization Authentication Codes of the initiator and responder respectively. Ni and Nr are the nonces sent in the IKE_SA_INIT messages that contain the IAC. The strings are 57 ASCII characters without null termination. prf_i() and prf_r() denote the PRFs selected in the initiator and responder configurations respectively. IASS_i and IASS_r denote the initialization authentication shared secret in the initiator and responder configurations respectively.

The pad strings are added so that if the IASS are derived from a password, the IKE implementation need not store the password in cleartext, which could not be used as a password equivalent for protocols other than IKEv2. Using different pad strings for each direction limits the information leakage about the IASS if IASS_i and IASS_r are equal. IAC_r is based on both Ni and Nr to prevent replay attacks on the IKE_SA_INIT reply while also preventing a MAC oracle on the responder, since the responder controls the random generation of Nr.

3.2. Initialization Authentication Notify Payload

The Initialization Authentication Notify Payload is defined as follows:



The 'Next Payload', 'C', 'RESERVED', 'Payload Length', 'Protocol ID', 'SPI Size', and 'Notify Message Type' fields are the same as described in Section 3 of [RFC7296]. The Critical ('C') bit MUST be set to 0. The 'SPI Size' field MUST be set to 0 to indicate that the SPI is not present in this message. The 'Protocol ID' MUST be set to 0, since the notification is specific to this IKE_SA_INIT message. The 'Payload Length' field is set to the length in octets of the entire payload, including the generic payload header. The 'Notify Message Type' field is set to indicate INITIALIZATION_AUTHENTICATION_CODE (TBD). The Initialization Authentication Code field has a variable length, and is computed according to Section 3.1.

3.3. Receiving Initialization Authentication

When the responder receives the initiator's IKE_SA_INIT request, it has not yet established the identity of the initiator, as the identity payload will come later. If the responder has distributed the same initialization authentication shared secret for all of its clients, it can easily verify that incoming IKE_SA_INIT requests come from clients that possess the shared secret. If the responder uses different initialization authentication shared secrets per client, it will have to iterate all of them to find a match since there is no identity sent with the IKE_SA_INIT request. Care should be taken with regards to the timing of the IKE_SA_INIT reply to avoid leaking information. If the responder cannot find a (IASS, PRF) combination in its configuration that matches the IAC in the incoming IKE_SA_INIT request, it MUST silently ignore the incoming packet. Not responding at all is crucial to hiding the fact that the responder is running an IKEv2 server. The responder SHOULD log the failure to facilitate debugging.

When the initiator receives the responder's IKE_SA_INIT reply, it knows the identity of the responder it is trying to establish a security association with. It can therefore use the (IASS, PRF) from its configuration to validate the IAC on the reply. If the IAC in the reply does not match what was computed from the configuration, the initiator treats this similarly to receiving and error on the reply and MUST fail the exchange and MUST NOT send the IKE_AUTH message it would have normally sent. This is crucial to protect the initiator identity (IDi) from an active on-path attacker. The initiator SHOULD log the failure to facilitate debugging.

4. Security Considerations

This document attempts to resolve the attacks described in Section 2 and no other attacks on IKEv2.

4.1. Timing Attacks

An IKEv2 responder wishing to stay hidden needs to ensure it doesn't leak information via the timing of its responses. In general if it receives an IKE_SA_INIT message whose IAC does not match, it simply does not respond. However if IKEv2 is running over TCP, the timing of when the responder closes the TCP connection can leak information. Implementors of hidden IKEv2 responders should ensure that they reply to bad input and to invalid IAC in similar time. In particular, if the server is also running another application protocol on the same port, it SHOULD reply to an invalid or missing IAC the same way as it would reply to an invalid request on that other protocol.

4.2. Replay Attacks

The initiator's IKE_SA_INIT message is sent unencrypted and can be replayed. The mechanism described in this document is still vulnerable to replays of the IKE_SA_INIT message. Note however that an obfuscated IKEv2 server running over TLS can leverage TLS to ensure the absence of on-path attackers inside the TLS channel between both endpoints.

The responder's IKE_SA_INIT message is also sent unencrypted and can also be replayed. However, the Initialization Authentication Code takes Ni as input so replaying a previous responder IKE_SA_INIT for a different IKEv2 exchange will have a different IAC and will be ignored.

4.3. Denial of Service Attacks

An IKEv2 responder implementing this specification opens themselves to computing more MACs for IKE_SA_INIT messages. We believe that downside is negligible compared to other DOS attacks on IKEv2.

5. IANA Considerations

If approved, this document defines a new payload in the IANA "IKEv2 Notify Message Types - Status Types" registry [IKEV2IANA]:

NOTIFY messages: status types	Value
INITIALIZATION_AUTHENTICATION_CODE	TBD

6. Normative References

[IKEV2IANA] "IANA, Internet Key Exchange Version 2 (IKEv2) Parameters", <https://www.iana.org/assignments/ikev2-parameters/>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8229] Pauly, T., Touati, S., and R. Mantha, "TCP Encapsulation of IKE and IPsec Packets", RFC 8229, DOI 10.17487/RFC8229, August 2017, <https://www.rfc-editor.org/info/rfc8229>.

Author's Address

David Schinazi
Apple Inc.
1 Infinite Loop
Cupertino, California 95014
US

Email: dschinazi@apple.com