Network Working Group
Internet-Draft
Intended status: Informational
Expires: 20 May 2021

R. Barnes
Cisco
R. Robert
Wire

16 November 2020

Using Messaging Layer Security (MLS) to Provide Keys for SFrame
draft-barnes-sframe-mls-00

Abstract

Secure Frames (SFrame) defines a compact scheme for encrypting real-time media. In order for SFrame to address cases where media are exchanged among many participants (e.g., real-time conferencing), it needs to be augmented with a group key management protocol. The Messaging Layer Security (MLS) protocol provides continuous group authenticated key exchange, allowing a group of participants in a media session to authenticate each other and agree on a group key. This document defines how the group keys produced by MLS can be used with SFrame to secure real-time sessions for groups.

Discussion Venues

This note is to be removed before publishing as an RFC.

Source for this draft and an issue tracker can be found at https://github.com/bifurcation/sframe-mls.

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 20 May 2021.

Table of Contents

1. Introduction

Secure Frames (SFrame) defines a compact scheme for encrypting real-time media. In order for SFrame to address cases where media are exchanged among many participants (e.g., real-time conferencing), it needs to be augmented with a group key management protocol. The Messaging Layer Security (MLS) protocol [!I-D.ietf-mls-protocol] provides continuous group authenticated key exchange. MLS provides several important security properties [!I-D.ietf-mls-arch]:

* Group Key Exchange: All members of the group at a given time know a secret key that is inaccessible to parties outside the group.

* Authentication of group members: Each member of the group can authenticate the other members of the group.

* Group Agreement: The members of the group all agree on the identities of the participants in the group.

* Forward Secrecy: There are protocol events such that if a member's state is compromised after the event, group secrets created before the event are safe.

* Post-compromise Security: There are protocol events such that if a member's state is compromised before the event, the group secrets created after the event are safe.

When a real-time session uses MLS as the basis for SFrame keys, these security properties apply to real-time media as well. In the remainder of this document, we define how to use the secrets produced by MLS to generate the keys required by SFrame.

[[OPEN ISSUE: We could define an MLS extension that would provide negotiation of SFrame parameters, notably the ciphersuite and the value E defined below.]]

2. SFrame Key Management

MLS creates a linear sequence of keys, each of which is shared among the members of a group at a given point in time. When a member joins or leaves the group, a new key is produced that is known only to the augmented or reduced group. Each step in the lifetime of the group is know as an "epoch", and each member of the group is assigned an "index" that is constant for the time they are in the group.

In SFrame, we derive per-sender "base_key" values from the group secret for an epoch, and use the KID field to signal the epoch and sender index. First, we use the MLS exporter to compute a shared SFrame secret for the epoch.

```
sframe_epoch_secret = MLS-Exporter("SFrame 10 MLS", "", AEAD.Nk)

sender_base_key[index] = HKDF-Expand(sframe_epoch_secret,
                           encode_big_endian(index, 4), AEAD.Nk)
```

[[OPEN ISSUE: MLS has its own "secret tree" that provides better forward secrecy properties within an epoch. (This scheme provides none.) An alternative approach would be to re-use the MLS secret tree, either directly or as a data structure.]]

The Key ID (KID) field in the SFrame header provides the epoch and index values that are needed to generate the appropriate key from the MLS key schedule.

```
KID = (sender_index << E) + (epoch % (1 << E))
```

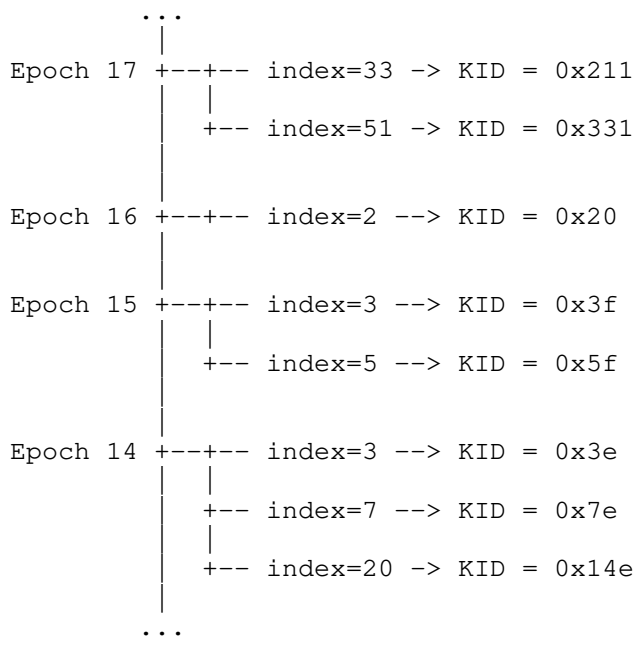
For compactness, do not send the whole epoch number. Instead, we send only its low-order E bits. The participants in the group MUST agree on the value of E for a given session, through some negotiation not specified here.

Note that E effectively defines a re-ordering window, since no more than 2^E epoch can be active at a given time. The better the participants are in sync with regard to key roll-over, and the less reordering of SFrame-protected payloads by the network, the fewer bits of epoch are necessary.

Receivers MUST be prepared for the epoch counter to roll over, removing an old epoch when a new epoch with the same E lower bits is introduced.

[[OPEN ISSUE: There might be some considerations for new joiners. Some trial decryption might be necessary to detect whether you're in epoch N or in epoch N + 1 << E.]]

Once an SFrame stack has been provisioned with the "sframe_epoch_secret" for an epoch, it can compute the required KIDs and "sender_base_key" values on demand, as it needs to encrypt/ decrypt for a given member.



MLS also provides an authenticated signing key pair for each participant. When SFrame uses signatures, these are the keys used to generate SFrame signatures.

3. Security Considerations

The security properties provided by MLS are discussed in detail in [!I-D.ietf-mls-arch] and [!I-D.ietf-mls-protocol]. This document extends those guarantees to SFrame.

It should be noted that the per-sender keys derived here do not provide per-sender authentication, since any member of the group could derive the same keys (as indeed they must in order to decrypt the protected payload). Per-sender keys are derived only to avoid nonce collision among multiple unsynchronized senders. So the authentication limitations of SFrame remain: There is per-sender authentication only when signatures are used. Otherwise, SFrame only authenticates membership in the group, and members are free to impersonate each other.

4. IANA Considerations

This document makes no request of IANA.

Appendix A. Acknowledgements

TODO

Authors' Addresses

Richard Barnes
Cisco

Email: rlb@ipv.sx

Raphael Robert
Wire

Email: raphael@wire.com