SFrame
E2EE for Video Conferencing

IETF 109

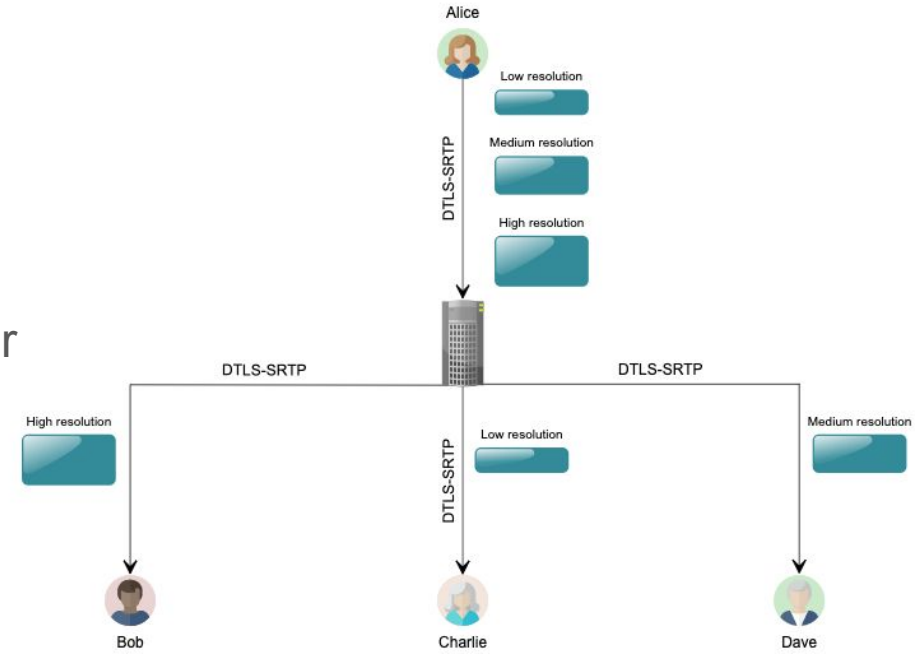
emadomara@google.com

Goals

- Goals
  - Security
  - Simplicity
  - Efficiency
  - Transport agnostic
- Non Goals
  - Signaling
  - Metadata payload format
  - Key exchange

Conference Calls System Overview

- Endpoints sends multiple media streams to a central media server
- These streams are encrypted to the server HBH like DTLS-SRTP
- The server routes the streams to other endpoints in the call
- The server **has access** to the entire media contents

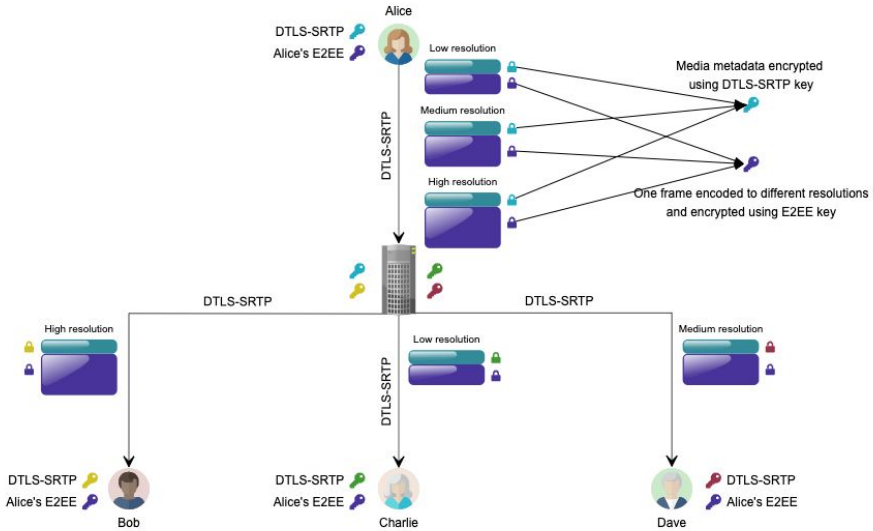


Secure Frame

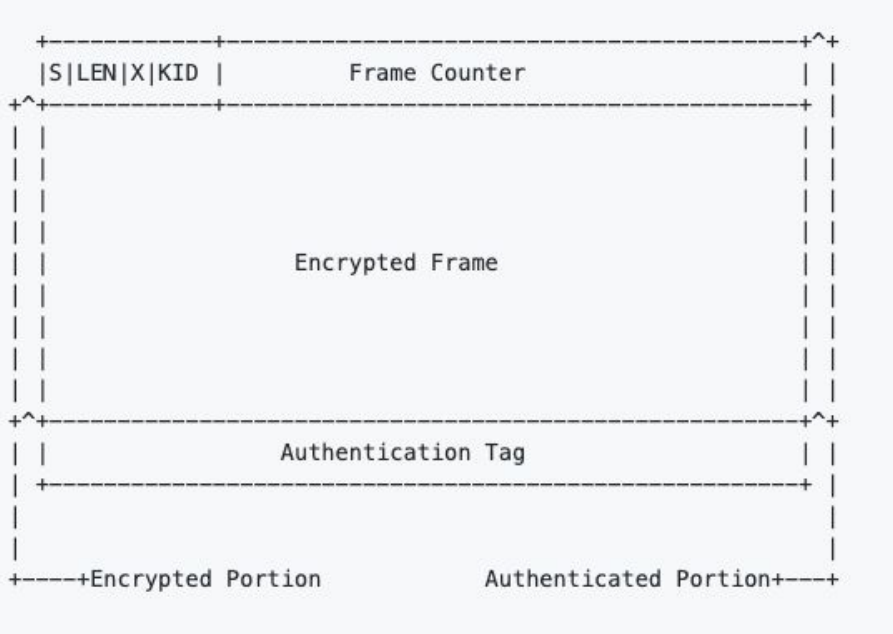
- A new protocol to end-to-end encrypt video conferences
- Encrypt the entire media frame instead of per packet encryption to reduce the overhead
- Transport agnostic as the encryption happens before packetization
- Simple to implement by the client and easy to adopt by existing media backends
- Compatible with existing packets fixing schemas like FEC

SFrame

- Mechanism to efficiently encrypt RTC traffic end to end
  - Encrypts the entire media frame rather than individual packets to minimize the overhead
  - Exposes only the metadata needed by the server to route the streams
  - Individual packets are still HBH encrypted
- SFrame keys are exchanged securely out of band between the endpoints
  - Each user has their own key to encrypt their outgoing traffic
  - Can be used with any KMS like Signal or MLS
  - Keys are exchanged via the signaling channel at the call setup and when the call participants changes
- The server can only access the media metadata but **can not access** the media contents



Wire Format



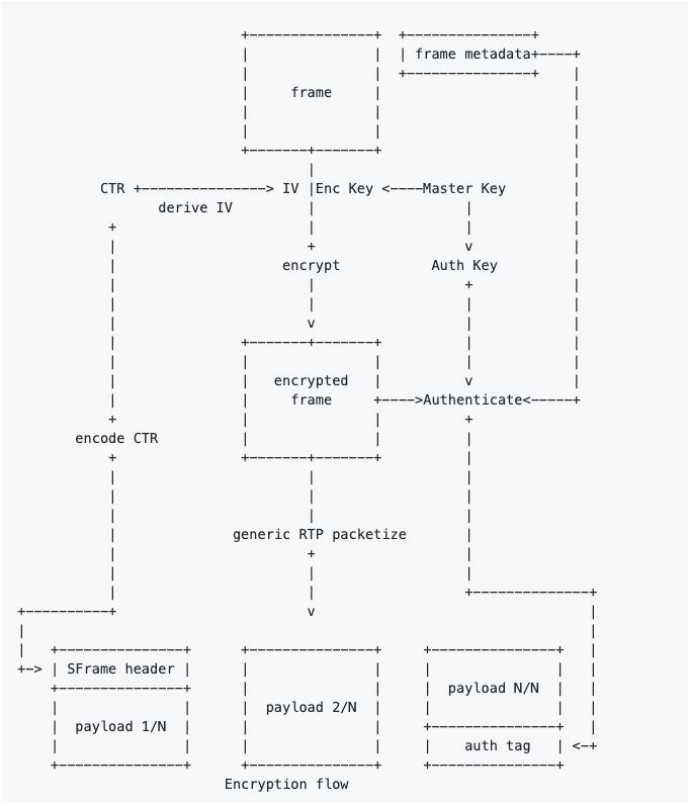
SFrame payload

SFrame short header

SFrame long header

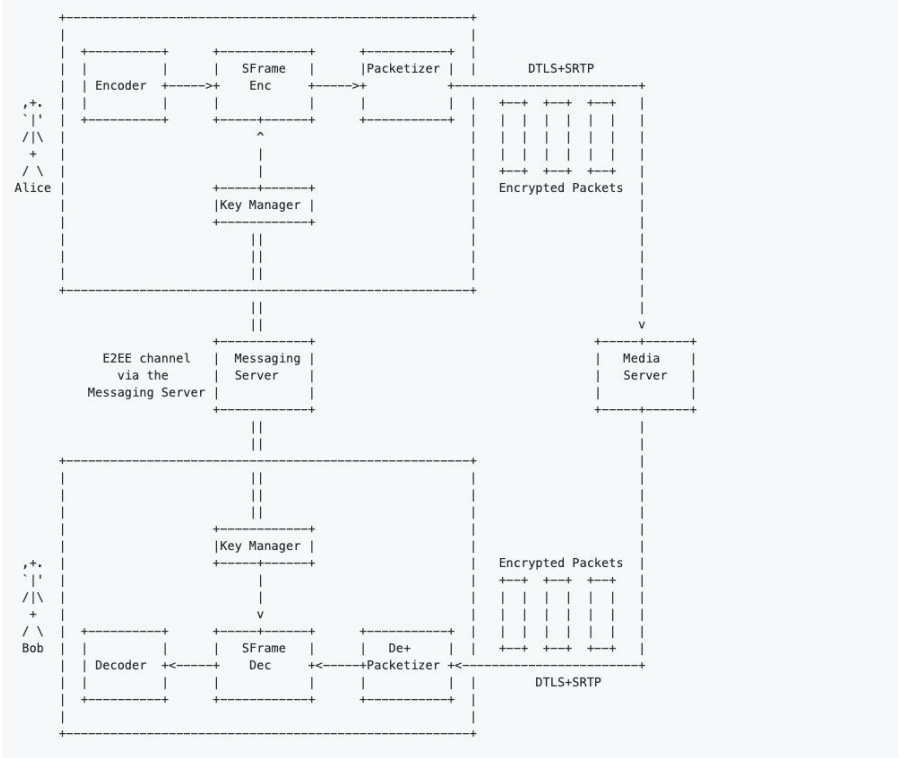
Encryption Schema

- Each endpoint creates and securely exchange their master key
- From the master key, SFrame derives 3 keys
  - Encryption key to encrypt the media frame
  - Authentication key to authenticate the encrypted frame. SFrame header and the media metadata
  - Salt key to derive the IV
- The entire payload is then split into smaller packets



SFrame in WebRTC

- SFrame works with existing RTC frameworks like WebRTC
- The encryptor in injected after the frame is encoded and before it is packetized
- Media metadata are passed to the server using a special RTP header extension
- The server can construct the encrypted frame without access the contents



SFrame in WebRTC

- Changes needed from other WebRTC WG
  - Signaling SFrame
  - RTP payload type
  - Frame metadata RTP header extension

Current Status

- Specs
  - SFrame protocol draft
    - The core protocol is done
    - Needs some minor tweaks to support subframes
  - Other documents needed
    - SFrame architecture document
      - Defines the overall system architecture
      - Defines the changes needed by other protocols (like WebRTC) to integrate with SFrame
    - MLS-SFrame
      - KMS integration document
- Implementation
  - Implemented and launched in Google Duo since April 2019

Questions ?

Please submit your questions to

sframe@ietf.org