SDP for the WebRTC
draft-nandakumar-rtcweb-sdp-02

Abstract

The Web Real-Time Communication (WebRTC) **[WEBRTC]** working group is charged to provide protocol support for direct interactive rich communication using audio,video and data between two peers' web browsers. With in the WebRTC framework, Session Description protocol (SDP) **[RFC4566]** is used for negotiating session capabilities between the peers. Such a negotiataion happens based on the SDP Offer/Answer exchange mechanism described in the RFC 3264 **[RFC3264]**.

This document serves a introductory purpose in describing the role of SDP for the most common WebRTC use-cases.

This SDP examples provided in this document is still a work in progress, but aims to align closest to the evolving standards.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

Javascript Session Exchange Protocol(JSEP) **[JSEP]** specifies a generic protocol needed to generate **[RFC3264]** offers and answers negotiated between the WebRTC peers for setting up, updating and tearing down a multimedia session. For this purpose, SDP is used to construct **[RFC3264]** offers/answers for describing (media and non-media) streams as appropriate for recipients of a session description to participate in the session.

The remainder of this document is organized as follows: Section 3 and 4 provides an overview of SDP and the Offer/Answer exchange mechanism. Section 5 and 6 provide sample SDP usages for the most common WebRTC use-cases.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in **[RFC2119]**.

3. SDP and the WebRTC

The purpose of this section is to provide a general overview of SDP and its components. For a more in-depth understanding, the readers are advised to refer to **[RFC4566]**.

The Session Description Protocol (SDP) **[RFC4566]** describes multimedia sessions,which can be audio,video,whiteboards,fax,modem, and other streams. It provides a general purpose,standard representation to describe various aspects of multimedia session such as media capabilities,transport addresses and related metadata in a transport agnostic manner,for the purposes of session announcement,session invitation and parameter negotiation.

As of today SDP is widely used in the context of Session Initiation Protocol, Real-time Transport Protocol and Real-time Streaming Protocol.

Below figure introduces high-level breakup of SDP into components that semantically describe a multimedia session,in our case,say,a WebRTC session **[WEBRTC]**. It by no means captures everything about SDP and hence, should be used for informational purposes only.
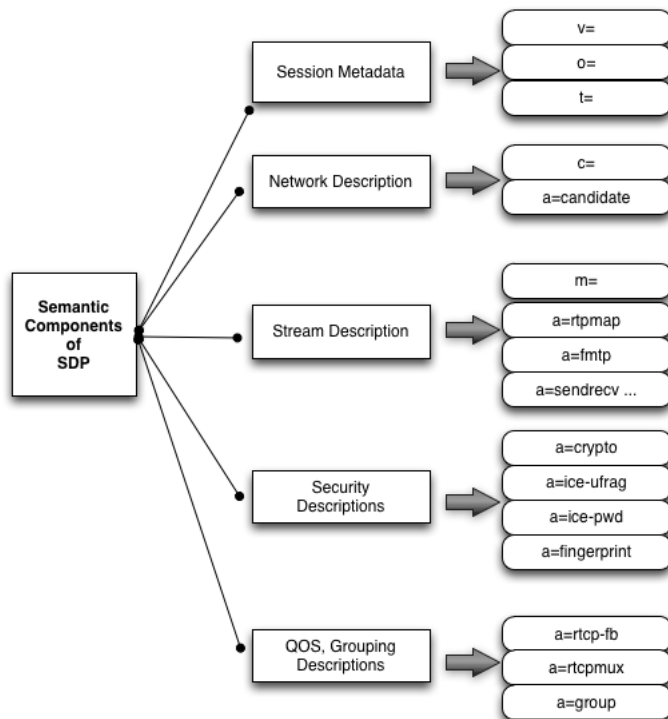
[WEBRTC] proposes JavaScript application to fully control the signaling plane of a multimedia session as described in the JSEP specification [JSEP]. JSEP provides mechanisms to create session characterisation and media definition information to conduct the session based on SDP exchanges.

In this context,SDP serves two purposes:

Provide grammatical structure syntatically

Semantically convey partipant's intention and capabilities.

---

4. Offer/Answer and the WebRTC

This section introduces SDP Offer/Answer Exchange mechanism mandated by WebRTC for negotitating session capabilities while setting up,updating and tearing down a WebRTC session. This section is intentionally brief in nature and interested readers are recommended to [RFC3264] for specific details on the protocol operation.

The Offer/Answer [RFC3264] model specifies rule for the bilateral exchange of Session Description Protocol (SDP) messages for creation of multimedia streams. It defines protocol with involved participants exchanging desired session characteristics from each others perspective modelled on SDP to negotiate the session between them.

In the most basic form,the protocol operation begins by one of the participants sending an initial SDP offer describing its intent to start a multimedia communication session. The participant receiving the offer MAY generate an SDP answer accepting the offer or it MAY reject the offer. If the session is accepted the Offer/Answer model guarantees a common view of the multimedia session between the participants.

At any time, either participant MAY generate a new SDP offer that updates the session in progress.

With in the context of WebRTC, the Offer/Answer model defines the state-machinery for WebRTC peers to negotiate session descriptions between them during the initial setup stages as well as for eventual session updates. Javascript Session Establishment Protocol specification [JSEP] for WebRTC provides the mechanism for generating [RFC3264] SDP offers and answers in order for both sides of the session to agree upon details such as list of media formats to be sent/received, bandwidth information, crypto parameters, transport parameters, for example.

The following sections provide samples of SDP message details and exchanges for the most common WebRTC usecases.

---

5. WebRTC Session Description Examples

A typical web based real-time multimedia communication session can be characterized as below:

It has zero or more Audio only,Video only or Audio/Video Media streams

MAY contain zero or more non-media data streams

All the streams are secured with DTLS/SRTP

ICE processing for NAT Traversal

Sessions over IPv4-only, IPv6-only, dual-stack based clients.

As mentioned earlier [WEBRTC] proposes using SDP based Offer/Answer model to negotiate multimedia session between peers' browsers Building on the concepts from the previous sections, the following subsections attempts to describe the usage of SDP for the most common WebRTC use-cases.

In all the use-cases, Alice and Bob are assumed to be the WebRTC peers unless mentioned otherwise. Pointers to appropriate RFCs and notes are provided, wherever necessary,against the SDP lines.

5.1. Secure Two-Way Audio,Video and Data with RTCP Feedback

This use-case allows two users to participate in a two-way communication session securely on their WebRTC enabled Web browsers.

Two-Way Secure Audio,Video, Data with RTCP Feedback
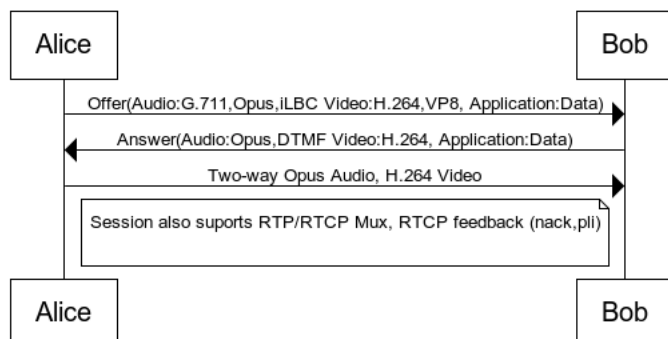
More specifically, this use-case demonstrates following aspects of a WebRTC session

- SRTP with DTLS based encryption
- RTP and RTCP Muxing
- RTCP based feedback and reduced size support
- ICE processing for NAT Traversal
- Audio Codec Offered : PCMU, Opus, iLBC
- Audio Codec Answered : Opus
- Video Codecs Offered: H.264, VP8
- Video Codecs Answered: H.264
- Data Channel Support

The tables (5.1 and 5.2) below capture in detail, the initial SDP Offer and Answer messages exchanged.

The exact SDP parameters specified for Data-Channel is still under the WG discussion and is expected to be updated once the final decision is reached..

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20518 0 IN IP4 0.0.0.0	**[RFC4566]** - Session Origin Information
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]** - Session Level ICE parameter
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]** - Session Level ICE parameter
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]** - Session DTLS Fingerprint for SRTP
a=rtcp-rsize	**[RFC5506]** - Alice intends to use reduced size RTCP for this session
m=audio 54609 RTP/SAVPF 0 109 98	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=rtpmap:0 PCMU/8000	**[RFC3551]** G.711 Audio Codec
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]** - Opus Codec 48khz, 2 channels
a=ptime:20	**[draft-ietf-payload-rtp-opus]** - Opus packetization of 20ms
a=rtpmap:98 iLBC/8000	**[RFC3952]** - Internet Low Bitrate Codec
a=fmtp:98 mode=20	**[RFC3952]**
a=sendrecv	**[RFC3264]** - Alice can send and recv audio

SDP Contents	RFC#/Notes
a=rtcp-mux	**[RFC5761]** - Alice can perform RTP/RTCP Muxing on port 54609
b=AS:64	**[RFC4566]** - Audio Session B/W of 64kbps
b=RS:800	**[RFC3556]** - RTCP b/w allocated to active data senders
b=RR:2400	**[RFC3556]** - RTCP b/w allocated to receivers
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]** - Host ICE Candidate
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]** - Server Reflexive ICE Candidate for the above host candidate
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]** - Second Host Candidate
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]** - Server Reflexive Candidate for the Second Host Candidate
a=rtcp-fb:109 nack	**[RFC5104]** - Indicates NACK RTCP feedback support
m=video 62537 RTP/SAVPF 99 120	**[RFC4566]**
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=rtpmap:99 H264/90000	**[RFC3984]** - H.264 Video Codec
a=fmtp:99 profile-level-id=4d0028;packetization-mode=1	**[RFC3984]**
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]** - VP8 video codec
a=sendrecv	**[RFC3264]** - Alice can send and recv video
a=rtcp-mux	**[RFC5761]** - Alice can perform RTP/RTCP Muxing on port 62537
b=AS:256	**[RFC4566]** - Audio Session B/W of 256kbps
b=RS:800	**[RFC3556]** - RTCP b/w allocated to active data senders
b=RR:2400	**[RFC3556]** - RTCP b/w allocated to receivers
a=candidate:0 1 UDP 2113667327 192.168.1.4 62537 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 24.23.204.141 62537 typ srflx raddr 192.168.1.4 rport 62537	**[RFC5245]**
a=candidate:0 2 2113667326 192.168.1.4 54721 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 54721 typ srflx raddr 192.168.1.4 rport 54721	**[RFC5245]**
a=rtcp-fb:99 nack pli	**[RFC5104]** - Indicates support for Picture loss Indication and NACK
a=rtcp-fb:99 ccm fir	**[RFC5104]** - Full Intra Frame Request-Codec Control Message support
a=rtcp-fb:120 nack pli	**[RFC5104]** - Indicates support for Picture loss Indication and NACK
a=rtcp-fb:120 ccm fir	**[RFC5104]** - Full Intra Frame Request-Codec Control Message support
m=application 56966 DTLS/SCTP 5000	**[draft-ietf-rtcweb-data-channel]**
c= IN IP4 24.23.204.141	**[RFC4566]**
a=sctpmap:5000 webrtc-Datachannel 1	**[draft-ietf-mmusic-sctp-sdp]** - One data stream of type chat
a=webrtc-Datachannel:5000 stream=1;label="channel 1";subprotocol="chat";	**[draft-ietf-mmusic-sctp-sdp]**
a=sendrecv	**[RFC3264]** - Alice can send and recv non-media data
a=candidate:0 1 UDP 2113667327 192.168.1.7 56966 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 24.23.204.141 56966 typ srflx raddr 192.168.1.7 rport 56966	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 51641 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 51641 typ srflx raddr 192.168.1.7 rport 51641	**[RFC5245]**

Table 1: 5.1 SDP Offer

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=bob 16833 0 IN IP4 0.0.0.0	**[RFC4566]** - Session Origin Information
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:c300d85b	**[RFC5245]** - Session Level ICE username frag
a=ice-pwd:de4e99bd291c325921d5d47efbabd9a2	**[RFC5245]** - Session Level ICE password

a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]** - Session DTLS Fingerprint for SRTP
a=rtcp-rsize	**[RFC5506]** - Alice intends to use reduced size RTCP for this session
m=audio 49203 RTP/SAVPF 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 98.248.92.77	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]** - Bob accepts only Opus Codec
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendrecv	**[RFC3264]** - Bob can send and recv audio
a=rtcp-mux	**[RFC5761]** - Bob can perform RTP/RTCP Muxing on port 49203
b=AS:64	**[RFC4566]** - Audio Session b/w of 64kbps
b=RS:800	**[RFC3556]** - RTCP b/w allocated to active data senders
b=RR:2400	**[RFC3556]** - RTCP b/w allocated to receivers
a=candidate:0 1 UDP 2113667327 192.168.1.7 49203 typ host	**[RFC5245]** - Host ICE Candidate for Opus Stream
a=ccandidate:1 1 UDP 1694302207 98.248.92.77 49203 typ srflx raddr 192.168.1.7 rport 49203	**[RFC5245]** - Server Reflexive ICE Candidate for the above host candidate
a=candidate:0 2 UDP 2113667326 192.168.1.7 60065 typ host	**[RFC5245]** - Second Host Candidate
a=candidate:1 2 UDP 1694302206 98.248.92.77 60065 typ srflx raddr 192.168.1.7 rport 60065	**[RFC5245]** - Server Reflexive Candidate for the Second Host Candidate
m=video 63130 RTP/SAVPF 99	**[RFC4566]**
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 98.248.92.771	**[RFC4566]**
a=rtpmap:99 H264/90000	**[RFC3984]** - Bob accepts H.264 Video Codec.
a=fmtp:99 profile-level-id=4d0028;packetization-mode=1	**[RFC3984]**
a=sendrecv	**[RFC3264]** - Bob can send and recv video
a=rtcp-mux	**[RFC5761]** - Bob can perform RTP/RTCP Muxing on port 63130
b=AS:256	**[RFC4566]** - Audio Session B/W of 256kbps
b=RS:800	**[RFC3556]** - RTCP b/w allocated to active data senders
b=RR:2400	**[RFC3556]** - RTCP b/w allocated to receivers
a=candidate:0 1 UDP 2113667327 192.168.1.7 63130 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 98.248.92.77 63130 typ srflx raddr 192.168.1.7 rport 63130	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 56607 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 98.248.92.77 56607 typ srflx raddr 192.168.1.7 rport 56607	**[RFC5245]**
a=rtcp-fb:99 nack pli	**[RFC5104]** - Indicates support for Picture loss Indication and NACK
a=rtcp-fb:99 ccm fir	**[RFC5104]** - Full Intra Frame Request-Codec Control Message support
m=application 55700 DTLS/SCTP 5000	**[draft-ietf-mmusic-sctp-sdp]**
c= IN IP4 98.248.92.771	**[RFC4566]**
a=sctpmap:5000 webrtc-Datachannel 1	**[draft-ietf-mmusic-sctp-sdp]**
a=webrtc-Datachannel:5000 stream=1;label="channel 1";subprotocol="chat";	**[draft-ietf-mmusic-sctp-sdp]**
a=sendrecv	**[RFC3264]** - Bob can send and recv non-media data
a=candidate:0 1 UDP 2113667327 192.168.1.7 55700 typ host	**[RFC5245]** - Refer 4.1 SDP Offer
a=candidate:1 1 UDP 1694302207 98.248.92.77 55700 typ srflx raddr 192.168.1.7 rport 55700	**[RFC5245]** Refer 4.1 SDP Offer
a=candidate:0 2 UDP 2113667326 192.168.1.7 58137 typ host	**[RFC5245]** Refer 4.1 SDP Offer
a=candidate:1 2 UDP 1694302206 98.248.92.77 58137 typ srflx raddr 192.168.1.7 rport 581371	**[RFC5245]** Refer 4.1 SDP Offer

Table 2: 5.1 SDP Answer

5.2. Secure Two-way Audio,Video,Data and remove data stream

This scenario builds upon from the usecase in the section 5.1 It extends by Alice removing data-stream once the session is in

progress.

There is an ongoing discussion with in the working group to allow addition and deletion of streams using partial Offer/Answer exchanges based on m=lines. Once a final decision is reached, the following example shall be updated to reflect the same.

WebRTC Session (Audio,Video,Datachannel) - Drop Datachannel
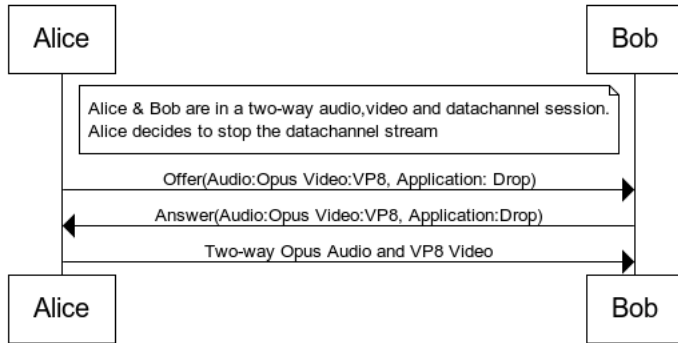
As a precondition, A Two-Way Audio,Video and Data Session is already setup.

SDP Contents	RFC#/Notes
v=1	**[RFC4566]** Incremented version to indicate the update
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=rtcp-rsize	**[RFC5506]**
m=audio 54609 RTP/SAVPF 0 109 98	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=rtpmap:98 iLBC/8000	**[RFC3952]** - Internet Low Bitrate Codec
a=fmtp:98 mode=20	**[RFC3952]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
a=rtcp-fb:109 nack	**[RFC5104]**
m=video 62537 RTP/SAVPF 99 120	**[RFC4566]**
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=rtpmap:99 H264/90000	**[RFC3984]**
a=fmtp:99 profile-level-id=4d0028;packetization-mode=1	**[RFC3984]**
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**

SDP Contents	RFC#/Notes
a=candidate:0 1 UDP 2113667327 192.168.1.4 62537 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 24.23.204.141 62537 typ srflx raddr 192.168.1.4 rport 62537	**[RFC5245]**
a=candidate:0 2 2113667326 192.168.1.4 54721 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 54721 typ srflx raddr 192.168.1.4 rport 54721	**[RFC5245]**
a=rtcp-fb:99 nack pli	**[RFC5104]**
a=rtcp-fb:99 ccm fir	**[RFC5104]**
a=rtcp-fb:120 nack pli	**[RFC5104]**
a=rtcp-fb:120 ccm fir	**[RFC5104]**
m=application 0 DTLS/SCTP 5000	**[draft-ietf-mmusic-sctp-sdp]** - Port 0 indicates dropping data stream
c= IN IP4 24.23.204.141	**[RFC4566]**
a=sctmap:5000 webrtc-Datachannel 1	**[draft-ietf-mmusic-sctp-sdp]**
a=webrtc-Datachannel:5000 stream=1;label="channel 1";subprotocol="chat";	**[draft-ietf-mmusic-sctp-sdp]**
a=sendrecv	**[RFC3264]**

Table 3: 5.2 SDP Updated Offer w/DataChannel Drop

SDP Contents	RFC#/Notes
v=1	**[RFC4566]**
o=bob 16833 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:c300d85b	**[RFC5245]**
a=ice-pwd:de4e99bd291c325921d5d47efbabd9a2	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=rtcp-rsize	**[RFC5506]**
m=audio 49203 RTP/SAVPF 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 98.248.92.77	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.7 49203 typ host	**[RFC5245]**
a=ccandidate:1 1 UDP 1694302207 98.248.92.77 49203 typ srflx raddr 192.168.1.7 rport 49203	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 60065 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 98.248.92.77 60065 typ srflx raddr 192.168.1.7 rport 60065	**[RFC5245]**
m=video 63130 RTP/SAVPF 99	**[RFC4566]**
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 98.248.92.771	**[RFC4566]**
a=rtpmap:99 H264/90000	**[RFC3984]**
a=fmtp:99 profile-level-id=4d0028;packetization-mode=1	**[RFC3984]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.7 63130 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 98.248.92.77 63130 typ srflx raddr 192.168.1.7 rport 63130	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 56607 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 98.248.92.77 56607 typ srflx raddr 192.168.1.7 rport 56607	**[RFC5245]**
a=rtcp-fb:99 nack pli	**[RFC5104]**
a=rtcp-fb:99 ccm fir	**[RFC5104]**
m=application 0 DTLS/SCTP 5000	**[draft-ietf-mmusic-sctp-sdp]** Bob accepts dropping the data stream

c= IN IP4 98.248.92.771	**[RFC4566]**
a=sctpmap:5000 webrtc-Datachannel 1	**[draft-ietf-mmusic-sctp-sdp]**
a=webrtc-Datachannel:5000 stream=1;label="channel 1";subprotocol="chat";	**[draft-ietf-mmusic-sctp-sdp]**
a=sendrecv	**[RFC3264]**

Table 4: 5.2 SDP Updated Answer

5.3. Secure Two-Way Audio,Video with BUNDLE Support Unknown

This use-case demonstrates a successfull audio and video media streams multiplexing scenario using SDP BUNDLE negotiation framework **[draft-ietf-mmusic-sdp-bundle-negotiation]**

Since Alice is unsure of the Bob's support for BUNDLE framework, the SDP example below shows

- An SDP Offer, in which the Alice assigns unique addresses to each "m=" line in the BUNDLE group, and requests the Answerer to select the Offerer's BUNDLE address.
- An SDP Answer, in which the Bob selects the BUNDLE address for the Offerer, and assigns its own local BUNDLE address to each "m=" line in the BUNDLE group.
- A subsequent SDP Offer from Alice, which is used to perform a BUNDLE Address Synchronization (BAS).

Once the Offer/Answer exchange completes, both Alice and Bob each end up using single RTP Session for both the Media Streams.

WebRTC Session - 2-Way Secure Audio,Video with BUNDLE

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20518 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=rtcp-rsize	**[RFC5506]**
a=group:BUNDLE foo bar	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 109	**[RFC4566]**
a=mid:foo	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.141	**[RFC4566]**
b=AS:200	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:11111	**[RFC5576]**

a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 62537 RTP/SAVPF 120	**[RFC4566]**
a=mid:bar	**[RFC5888]** Video m=line part of the Bundle group with a unique port number
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.141	**[RFC4566]**
b=AS:1756	**[RFC4566]**
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:22222	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 62537 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 24.23.204.141 62537 typ srflx raddr 192.168.1.4 rport 62537	**[RFC5245]**
a=candidate:0 2 2113667326 192.168.1.4 54721 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 54721 typ srflx raddr 192.168.1.4 rport 54721	**[RFC5245]**

Table 5: 5.3 SDP Offer w/BUNDLE

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=bob 16833 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:c300d85b	**[RFC5245]**
a=ice-pwd:de4e99bd291c325921d5d47efbabd9a2	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=rtcp-rsize	**[RFC5506]**
a=group:BUNDLE foo bar	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Bob supports BUNDLE semantics.
m=audio 49203 RTP/SAVPF 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
a=mid:foo	**[RFC5888]** Audio m=line part of the BUNDLE group
c= IN IP4 98.248.92.77	**[RFC4566]**
b=AS:200	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:33333	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.7 49203 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 98.248.92.77 49203 typ srflx raddr 192.168.1.7 rport 49203	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 60065 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 98.248.92.77 60065 typ srflx raddr 192.168.1.7 rport 60065	**[RFC5245]**
m=video 49203 RTP/SAVPF 120	**[RFC4566]**
a=mid:bar	**[RFC5888]** Video m=line part of the BUNDLE group with the port from audio line repeated
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
b=AS:1756	**[RFC4566]**
c= IN IP4 98.248.92.77	**[RFC4566]**

a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:44444	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.7 49203 typ host	**[RFC5245]** - Candidate lines identical with the audio m-line.
a=candidate:1 1 UDP 1694302207 98.248.92.77 49203 typ srflx raddr 192.168.1.7 rport 49203	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 60065 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 98.248.92.77 60065 typ srflx raddr 192.168.1.7 rport 60065	**[RFC5245]**

Table 6: 5.3 SDP Answer w/BUNDLE

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20518 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=rtcp-rsize	**[RFC5506]**
a=group:BUNDLE foo bar	**[draft-ietf-mmusic-sdp-bundle-negotiation]**
m=audio 54609 RTP/SAVPF 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
a=mid:foo	**[RFC5888]** - Port number finalized as Bundle Address.
c= IN IP4 24.23.204.141	**[RFC4566]**
b=AS:200	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:11111	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 54609 RTP/SAVPF 120	**[RFC4566]**
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
a=mid:bar	**[RFC5888]** - Same Bundle address from the audio m=line
c= IN IP4 24.23.204.141	**[RFC4566]**
b=AS:1756	**[RFC4566]**
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:22222	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]** - Candidate lines identical with the audio m-line.
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**

Table 7: 5.3 SDP Offer for BAS

5.4. Secure Two-Way Audio,Video w/BUNDLE Support Known

This use-case is a successful audio and video stream multiplexing scenario, with Alice and Bob aware of each others support for SDP BUNDLE framework **[draft-ietf-mmusic-sdp-bundle-negotiation]**.

WebRTC Session - 2-Way Secure Audio,Video with BUNDLE Support Known

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20518 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=rtcp-rsize	**[RFC5506]**
a=group:BUNDLE foo bar	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics.
m=audio 10000 RTP/SAVPF 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
a=mid:foo	**[RFC5888]** - Audio m=line part of BUNDLE group
c= IN IP4 24.23.204.141	**[RFC4566]**
b=AS:200	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:11111	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 10000 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 10000 typ srflx raddr 192.168.1.4 rport 10000	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 10000 RTP/SAVPF 120	**[RFC4566]**
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
a=mid:bar	**[RFC5888]** - Video m=line with Bundle address same as the audio m=line
c= IN IP4 24.23.204.141	**[RFC4566]**
b=AS:1000	**[RFC4566]**
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:22222	**[RFC5576]**

a=candidate:0 1 UDP 2113667327 192.168.1.4 10000 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 10000 typ srflx raddr 192.168.1.4 rport 10000	**[RFC5245]**
a=candidate:0 2 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**

Table 8: 5.4 SDP Offer w/BUNDLE

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=bob 16833 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:c300d85b	**[RFC5245]**
a=ice-pwd:de4e99bd291c325921d5d47efbabd9a2	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=rtcp-rsize	**[RFC5506]**
a=group:BUNDLE foo bar	**[draft-ietf-mmusic-sdp-bundle-negotiation]** - Bob supports BUNDLE semantics.
m=audio 20000 RTP/SAVPF 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
a=mid:foo	**[RFC5888]** - Audio m=line part of the BUNDLE group
b=AS:200	**[RFC4566]**
c= IN IP4 98.248.92.77	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:33333	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.7 20000 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 98.248.92.77 20000 typ srflx raddr 192.168.1.7 rport 20000	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 60065 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 98.248.92.77 60065 typ srflx raddr 192.168.1.7 rport 60065	**[RFC5245]**
m=video 20000 RTP/SAVPF 120	**[RFC4566]**
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
a=mid:bar	**[RFC5888]** - Video m=line with Bundle address same as the audio m=line
b=AS:1000	**[RFC4566]**
c= IN IP4 98.248.92.77	**[RFC4566]**
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:44444	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.7 20000 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 98.248.92.77 20000 typ srflx raddr 192.168.1.7 rport 20000	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 60065 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 98.248.92.77 60065 typ srflx raddr 192.168.1.7 rport 60065	**[RFC5245]**

Table 9: 5.4 SDP Answer w/BUNDLE

5.5. Secure Two-Way Audio,Video w/BUNDLE Unsupported
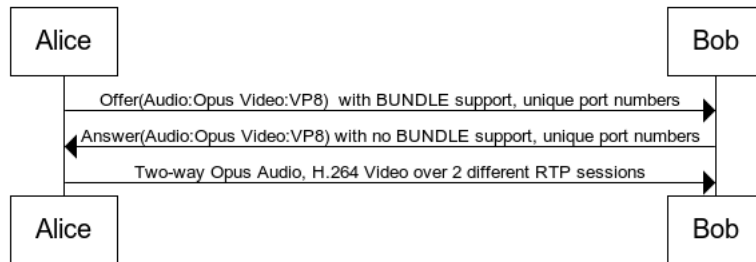
This use-case illustrates SDP Offer/Answer exchange when the far-end (Bob) either doesn't support media bundling or doesn't

want to group m=lines over a single 5-tuple.

On successful Offer/Answer exchange, Alice and Bob each end up using separate RTP sessions for audio and video media streams respectively.

WebRTC Session - 2-Way Secure Audio,Video with BUNDLE Unsupported

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20518 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=rtcp-rsize	**[RFC5506]**
a=group:BUNDLE foo bar	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 55232 RTP/SAVPF 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
a=mid:foo	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
c= IN IP4 24.23.204.141	**[RFC4566]**
b=AS:200	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:11111	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 55232 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 55232 typ srflx raddr 192.168.1.4 rport 55232	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 54332 RTP/SAVPF 120	**[RFC4566]**
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
a=mid:bar	**[RFC5888]** Video m=line part of the BUNDLE group with a unique port number
c= IN IP4 24.23.204.141	**[RFC4566]**
b=AS:1000	**[RFC4566]**
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:22222	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54332 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 24.23.204.141 54332 typ	

srflx raddr 192.168.1.4 rport 54332	**[RFC5245]**
a=candidate:0 2 2113667326 192.168.1.4 54721 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 54721 typ srflx raddr 192.168.1.4 rport 54721	**[RFC5245]**

Table 10: 5.5 SDP Offer w/BUNDLE

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=bob 16833 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:c300d85b	**[RFC5245]**
a=ice-pwd:de4e99bd291c325921d5d47efbabd9a2	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=rtcp-rsize	**[RFC5506]**
m=audio 53214 RTP/SAVPF 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
b=AS:200	**[RFC4566]**
c= IN IP4 98.248.92.77	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendrecv	**[RFC3264]**
a=candidate:0 1 UDP 2113667327 192.168.1.7 53214 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 98.248.92.77 53214 typ srflx raddr 192.168.1.7 rport 53214	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 60065 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 98.248.92.77 60065 typ srflx raddr 192.168.1.7 rport 60065	**[RFC5245]**
m=video 58679 RTP/SAVPF 120	**[RFC4566]**
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
b=AS:2000	**[RFC4566]**
c= IN IP4 98.248.92.77	**[RFC4566]**
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=sendrecv	**[RFC3264]**
a=candidate:0 1 UDP 2113667327 192.168.1.7 58679 typ host	**[RFC5245]**
a=candidate:1 1 UDP 1694302207 98.248.92.77 58679 typ srflx raddr 192.168.1.7 rport 58679	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 56607 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 98.248.92.77 56607 typ srflx raddr 192.168.1.7 rport 56607	**[RFC5245]**

Table 11: 5.5 SDP Answer without BUNDLE

5.6. Successful One Way Session with 2 Video Streams

In this scenario Alice and Bob engage in one-way multimedia session with Bob receiving two video streams, one corresponding to Alice's video and other corresponding to her presentation slides.

1 Way Audio & Video w/2 Video Streams

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=rtcp-rsize	**[RFC5506]**
m=audio 54609 RTP/SAVPF 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]** - Send only audio stream
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5104]**
m=video 62537 RTP/SAVPF 120	**[RFC4566]**
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=content:speaker	**[RFC4796]** - Stream 1 for Alice's video
a=sendonly	**[RFC3264]** - Send only video stream
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 62537 typ host	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 54721 typ host	**[RFC5245]**
m=video 62539 RTP/SAVPF 120	**[RFC4566]**
a=msid:mb ta	Identifies RTCMediaStream ID (mb) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.141	**[RFC4566]**

SDP Contents	RFC#/Notes
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=content:slides	**[RFC4796]** - Stream 2 for Alice's slides
a=sendonly	**[RFC3264]** - Send only video stream
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 62539 typ host	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 54723 typ host	**[RFC5245]**

Table 12: 5.6 SDP Offer

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=bob 16833 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:c300d85b	**[RFC5245]**
a=ice-pwd:de4e99bd291c325921d5d47efbabd9a2	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=rtcp-rsize	**[RFC5506]**
m=audio 49203 RTP/SAVPF 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 98.248.92.77	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=recvonly	**[RFC3264]** - Receive only audio stream
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.7 49203 typ host	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 60065 typ host	**[RFC5245]**
m=video 63130 RTP/SAVPF 120	**[RFC4566]**
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 98.248.92.771	**[RFC4566]**
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=content:speaker	**[RFC4796]** - Stream 1 for Alice's Video
a=recvonly	**[RFC3264]** - Receive Only Video Stream 1
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.7 63130 typ host	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 56607 typ host	**[RFC5245]**
m=video 63133 RTP/SAVPF 120	**[RFC4566]**
a=msid:mb ta	Identifies RTCMediaStream ID (mb) and RTCMediaStreamTrack ID (ta)
c= IN IP4 98.248.92.771	**[RFC4566]**
a=rtpmap:120 VP8/90000	**[draft-ietf-payload-vp8]**
a=content:slides	**[RFC4796]** - Stream 2 for Alice's Slides
a=recvonly	**[RFC3264]** - Receive Only Video Stream 2
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.7 63133 typ host	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.7 56609 typ host	**[RFC5245]**

Table 13: 5.6 SDP Answer

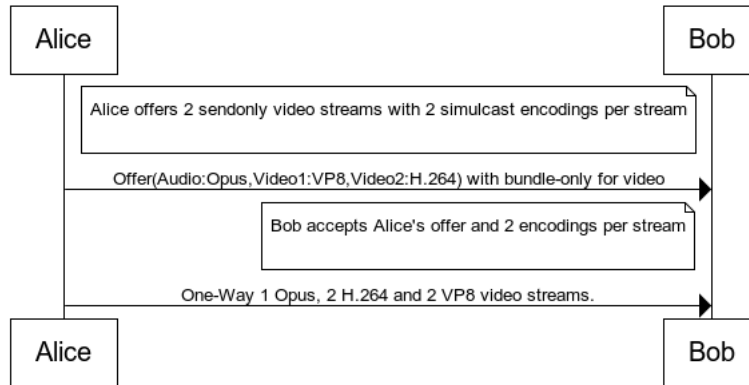### 5.7. Sendonly Simulcast w/2 cameras and 2 encodings per camera

This SDP below shows Offer/Answer exchange with audio and two video streams each of which can send two different resolutions.

One video stream supports VP8, while the other supports H.264.

bundle-only framework is used along with BUNDLE grouping framework to enable multiplexing of all the 5 streams (audio stream + 4 video streams) over a single RTP Session.

1 Way Successful Simulcast

Alice	Bob

Alice offers 2 sendonly video streams with 2 simulcast encodings per stream

Offer(Audio:Opus,Video1:VP8,Video2:H.264) with bundle-only for video

Bob accepts Alice's offer and 2 encodings per stream

One-Way 1 Opus, 2 H.264 and 2 VP8 video streams.

Alice	Bob

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=group:BUNDLE m0 m1 m2	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 0 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m0	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:11111	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 0 RTP/SAVPF 98 100	bundle-only video line with port number set to zero
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m1	**[RFC5888]** Video m=line part of BUNDLE group
a=rtpmap:98 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:100 VP8/90000	**[draft-ietf-payload-vp8]**
a=imageattr:98 [x=1280,y=720]	**[RFC6236]**Camera-1,Encoding-1 Resolution
a=fmtp:98 max-fr=30	**[RFC4566]**
a=imageattr:100 [x=640,y=480]	**[RFC6236]** Camera-1,Encoding-2 Resolution
a=fmtp:100 max-fr=15	**[RFC4566]**
a=ssrc-group:SIMULCAST 12345 45678	**[RFC5576]**
a=ssrc:12345	**[RFC5576]**

a=ssrc:45678	**[RFC5576]**
a=ssrc:12345 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-1 SSRC with Session CNAME
a=ssrc:45678 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-2 SSRC with Session CNAME
a=sendonly	**[RFC3264]** - Send only video stream
a=rtcp-mux	**[RFC5761]**
a=bundle-only	Add reference to unified plan when available
m=video 0 RTP/SAVPF 98 100	bundle-only video line with port number set to zero
a=msid:ma tc	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tc)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m2	**[RFC5888]** Video m=line part of BUNDLE group
a=rtpmap:98 H264/90000	**[RFC3984]**
a=rtpmap:100 H264/90000	**[RFC3984]**
a=fmtp:98 profile-level-id=4d0028;packetization-mode=1;max-fr=30	**[RFC3984]**Camera-2,Encoding-1 Resolution
a=fmtp:100 profile-level-id=4d0028;packetization-mode=1;max-fr=15	**[RFC3984]**Camera-1,Encoding-2 Resolution
a=ssrc:67890	**[RFC5576]**
a=ssrc:56789	**[RFC5576]**
a=ssrc-group:SIMULCAST 67890 56789	**[RFC5576]**
a=ssrc:67890 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-1 SSRC with Session CNAME
a=ssrc:56789 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-2 SSRC with Session CNAME
a=sendonly	**[RFC3264]** - Send only video stream
a=rtcp-mux	**[RFC5761]**
a=bundle-only	Add reference to unified plan when available

Table 14: 5.8 SDP Offer

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:ufrag:c300d85b	**[RFC5245]**
a=ice-pwd:de4e99bd291c325921d5d47efbabd9a2	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=group:BUNDLE m0 m1 m2	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 0 109	**[RFC4566]**
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m0	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:22222	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 54609 RTP/SAVPF 98 100	BUNDLE accepted with port repeated from the audio port

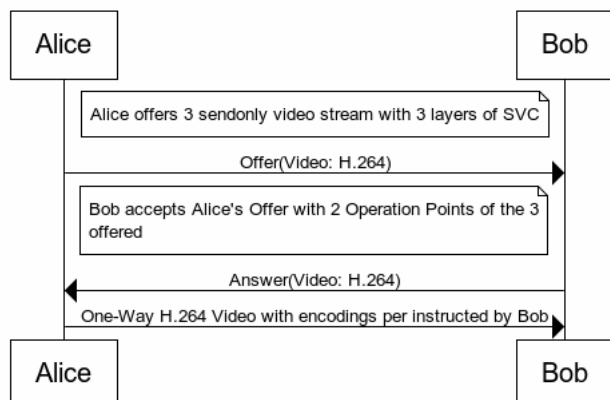|	
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m1	**[RFC5888]** Video m=line part of BUNDLE group
a=rtpmap:98 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:100 VP8/90000	**[draft-ietf-payload-vp8]**
a=imageattr:98 [x=1280,y=720]	**[RFC6236]** Camera-1,Encoding-1 Resolution
a=fmtp:98 max-fr=30	**[RFC4566]**
a=imageattr:100 [x=640,y=480]	**[RFC6236]** Camera-1,Encoding-2 Resolution
a=fmtp:100 max-fr=15	**[RFC4566]**
a=ssrc-group:SIMULCAST 54321 87654	**[RFC5576]**
a=ssrc:54321	**[RFC5576]**
a=ssrc:87654	**[RFC5576]**
a=ssrc:54321 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-1 SSRC with Session CNAME
a=ssrc:87654 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-2 SSRC with Session CNAME
a=sendonly	**[RFC3264]** - Send only video stream
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
a=rtcp-mux	**[RFC5576]**
a=bundle-only	Add reference to unified plan when available
m=video 54609 RTP/SAVPF 98 100	BUNDLE accepted with port repeated from the audio port
c= IN IP4 24.23.204.141	**[RFC4566]**
a=msid:ma tc	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tc)
a=mid:m2	**[RFC5888]** Video m=line part of BUNDLE group
a=rtpmap:98 H264/90000	**[RFC3984]**
a=rtpmap:10 H264/90000	**[RFC3984]**
a=fmtp:98 profile-level-id=4d0028;packetization-mode=1;max-fr=30	**[RFC3984]** Camera-2,Encoding-1 Resolution
a=fmtp:100 profile-level-id=4d0028;packetization-mode=1;max-fr=15	**[RFC3984]** Camera-1,Encoding-2 Resolution
a=ssrc:90876	**[RFC5576]**
a=ssrc:89754	**[RFC5576]**
a=ssrc-group:SIMULCAST 90876 89754	**[RFC5576]**
a=ssrc:90876 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-1 SSRC with Session CNAME
a=ssrc:89754 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-2 SSRC with Session CNAME
a=sendonly	**[RFC3264]** - Send only video stream
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
a=rtcp-mux	**[RFC5576]**
a=bundle-only	Add reference to unified plan when available

Table 15: 5.8 SDP Answer

5.8. Successful SVC Video Stream

This section shows an SDP Offer/Answer for a session with an audio and a single video stream. The video stream is layered coding at 3 different resolutions based on **[RFC5583]**. The video m=line shows 3 streams with last stream (payload 100) dependent on streams with payload 96 and 97 for decoding.

2 way SVC Video

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=group:BUNDLE m0 m1	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 0 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m0	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 0 RTP/SAVPF 96 97 100	bundle-only video line with port number set to zero
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tc)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m1	**[RFC5888]** Audio m=line part of BUNDLE group
a=msid:ma tb	Add appropriate reference when available
b=AS:2500	**[RFC4566]**
a=rtpmap:96 H264/90000	**[RFC3984]**
a=fmtp:96 profile-level-id=4d0028;packetization-mode=1;max-fr=30;max-fs=8040	**[RFC3984]**H.264 Layer 1
a=rtpmap:97 H264/90000	**[RFC3984]**
a=fmtp:97 profile-level-id=4d0028;packetization-mode=1;max-fr=15;max-fs=1200	**[RFC3984]** H.264 Layer 2
a=rtpmap:100 H264-SVC/90000	**[RFC3984]**
a=fmtp:100 profile-level-id=4d0028;packetization-	

mode=1;max-fr=30;max-fs=8040	**[RFC3984]**
a=depend:100 lay m1:96,97;	**[RFC5583]**Layer 3 dependent on layers 1 and 2
a=sendonly	**[RFC3264]** - Send only video stream
a=rtcp-mux	**[RFC5761]**
a=bundle-only	Add reference to unified plan when available

Table 16: 5.9 SDP Offer with SVC

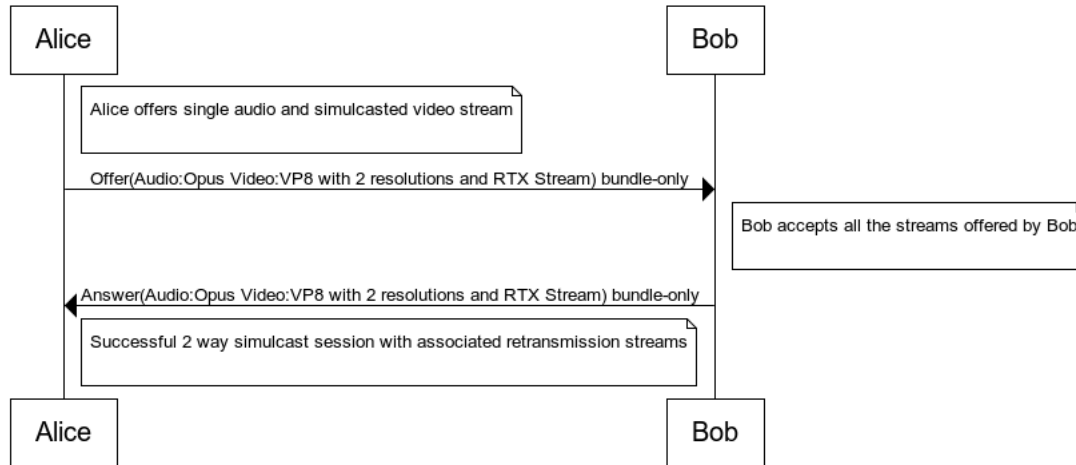| SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=group:BUNDLE m0 m1	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 0 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.142	**[RFC4566]**
a=mid:m0	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.5 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.142 54609 typ srflx raddr 192.168.1.5 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.5 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.142 64678 typ srflx raddr 192.168.1.5 rport 64678	**[RFC5245]**
m=video 54609 RTP/SAVPF 96 100	BUNDLE accepted Bundle address same as audio m=line.
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.142	**[RFC4566]**
a=mid:m1	**[RFC5888]** Video m=line part of BUNDLE group
b=AS:2500	**[RFC4566]**
a=rtpmap:96 H264/90000	**[RFC3984]**
a=fmtp:96 profile-level-id=4d0028;packetization-mode=1;max-fr=30;max-fs=8040	**[RFC3984]**H.264 Layer 1
a=rtpmap:100 H264-SVC/90000	**[RFC3984]**
a=fmtp:100 profile-level-id=4d0028;packetization-mode=1;max-fr=30;max-fs=8040	**[RFC3984]**
a=depend:100 lay m1:96;	**[RFC5583]** Bob chooses 2 Codec Operation points
a=candidate:0 1 UDP 2113667327 192.168.1.5 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.142 54609 typ srflx raddr 192.168.1.5 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.5 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.142 64678 typ srflx raddr 192.168.1.5 rport 64678	**[RFC5245]**
a=recvonly	**[RFC3264]** - Receive only video stream
a=rtcp-mux	**[RFC5761]**
a=bundle-only	Add reference to unified plan when available

Table 17: 5.9 SDP Answer with SVC

5.9. Successful Simulcast Video Streams with Retransmisson

This section shows an SDP Offer/Answer exchange for a simulcast scenario with 2 two resolutions and has **[RFC4588]** style re-transmission flows.

Simulcast Streams with Retransmission

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=group:BUNDLE m0 m1	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 0 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m0	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:11111	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 0 RTP/SAVPF 98 100 101 103	bundle-only video line with port number set to zero
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m1	**[RFC5888]** Audio m=line part of BUNDLE group
b=AS:1756	**[RFC4566]**

SDP Contents	RFC#/Notes
a=rtpmap:98 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:100 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:101 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:103 VP8/90000	**[draft-ietf-payload-vp8]**
a=fmtp:98 max-fr=30;max-fs=8040	**[RFC4566]**
a=fmtp:100 max-fr=15;max-fs=1200	**[RFC4566]**
a=fmtp:101 apt=98;rtx-time=3000	**[RFC4588]**
a=fmtp:103 apt=100;rtx-time=3000	**[RFC4588]**
a=ssrc-group:SIMULCAST 12345 78990	Simulcast group
a=ssrc-group:FID 12345 34567	**[RFC5888]**
a=ssrc-group:FID 78990 90887	**[RFC5888]**
a=ssrc:12345	**[RFC5576]**
a=ssrc:78990	**[RFC5576]**
a=ssrc:34567	**[RFC5576]**
a=ssrc:90887	**[RFC5576]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=bundle-only	Add reference to unified plan when available

Table 18: 5.10 SDP Offer w/Simulcast, RTX

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=group:BUNDLE m0 m1	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 0 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.142	**[RFC4566]**
a=mid:m0	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:33333	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.5 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.142 54609 typ srflx raddr 192.168.1.5 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.5 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.142 64678 typ srflx raddr 192.168.1.5 rport 64678	**[RFC5245]**
m=video 54609 RTP/SAVPF 98 100 101 103	BUNDLE accepted with Bundle address identical to audio m-line
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.142	**[RFC4566]**
a=mid:m1	**[RFC5888]** Video m=line part of BUNDLE group
b=AS:1756	**[RFC4566]**
a=rtpmap:98 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:100 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:101 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:103 VP8/90000	**[draft-ietf-payload-vp8]**
a=fmtp:98 max-fr=30;max-fs=8040	**[RFC4566]**

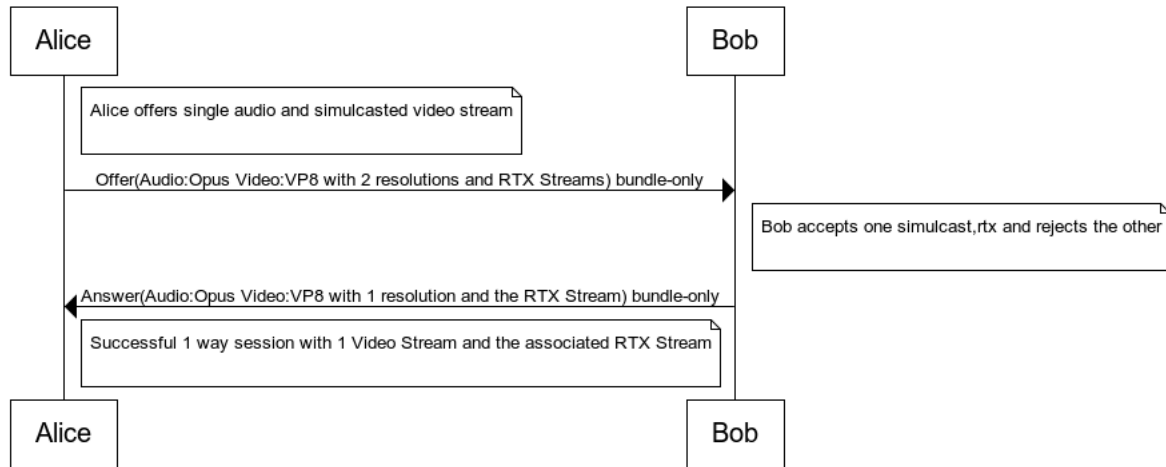|	
a=fmtp:100 max-fr=15;max-fs=1200	**[RFC4566]**
a=fmtp:101 apt=98;rtx-time=3000	**[RFC4588]**
a=fmtp:103 apt=100;rtx-time=3000	**[RFC4588]**
a=candidate:0 1 UDP 2113667327 192.168.1.5 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.142 54609 typ srflx raddr 192.168.1.5 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.5 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.142 64678 typ srflx raddr 192.168.1.5 rport 64678	**[RFC5245]**
a=ssrc-group:SIMULCAST 54321 77656	Simulcast group
a=ssrc-group:FID 54321 88776	**[RFC5888]**
a=ssrc-group:FID 77656 12908	**[RFC5888]**
a=ssrc:54321	**[RFC5576]**
a=ssrc:77656	**[RFC5576]**
a=ssrc:88776	**[RFC5576]**
a=ssrc:12908	**[RFC5576]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=bundle-only	Add reference to unified plan when available

Table 19: 5.10 SDP Answer w/Simulcast, RTX

5.10. Successful 1-way Simulcast with 2 resolutions and RTX - One resolution rejected

This section shows an SDP Offer/Answer exchange for a simulcast scenario with 2 two resolutions and has **[RFC4588]** style re-transmission flows.

It also showcases when Bob rejects one of the Simulcast Video Stream which results in the rejection of the associated repair stream implicitly

Simulcast Streams with RTX Rejected

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**

SDP Contents	RFC#/Notes
a=group:BUNDLE m0 m1	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 0 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m0	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:11111	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 0 RTP/SAVPF 98 100 101 103	bundle-only video line with port number set to zero
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m1	**[RFC5888]** Audio m=line part of BUNDLE group
b=AS:1756	**[RFC4566]**
a=rtpmap:98 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:100 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:101 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:103 VP8/90000	**[draft-ietf-payload-vp8]**
a=fmtp:98 max-fr=30;max-fs=8040	**[RFC4566]**
a=fmtp:100 max-fr=15;max-fs=1200	**[RFC4566]**
a=fmtp:101 apt=98;rtx-time=3000	**[RFC4588]**
a=fmtp:103 apt=100;rtx-time=3000	**[RFC4588]**
a=ssrc-group:SIMULCAST 12345 78990	Simulcast group
a=ssrc-group:FID 12345 34567	**[RFC5888]**
a=ssrc-group:FID 78990 90887	**[RFC5888]**
a=ssrc:12345	**[RFC5576]**
a=ssrc:78990	**[RFC5576]**
a=ssrc:34567	**[RFC5576]**
a=ssrc:90887	**[RFC5576]**
a=sendonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=bundle-only	Add reference to unified plan when available

Table 20: 5.11 SDP Offer w/Simulcast, RTX

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=group:BUNDLE m0 m1	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 0 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)

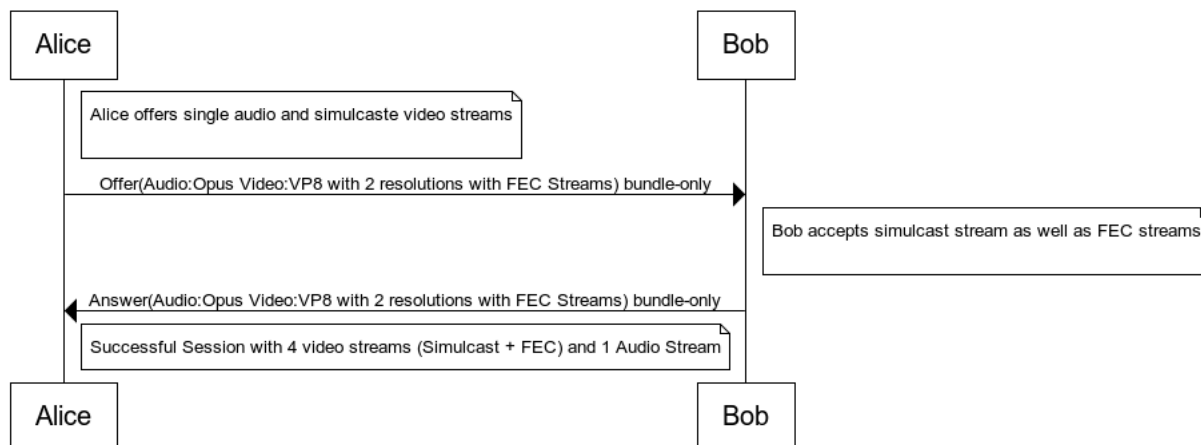|	
c= IN IP4 24.23.204.142	**[RFC4566]**
a=mid:m0	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=recvonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:33333	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.5 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.142 54609 typ srflx raddr 192.168.1.5 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.5 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.142 64678 typ srflx raddr 192.168.1.5 rport 64678	**[RFC5245]**
m=video 54609 RTP/SAVPF 98 101	BUNDLE accepted with Bundle address identical to audio m-line
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.142	**[RFC4566]**
a=mid:m1	**[RFC5888]** Video m=line part of BUNDLE group
b=AS:1756	**[RFC4566]**
a=rtpmap:98 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:101 VP8/90000	**[draft-ietf-payload-vp8]**
a=fmtp:98 max-fr=30;max-fs=8040	**[RFC4566]**
a=fmtp:101 apt=98;rtx-time=3000	**[RFC4588]**
a=candidate:0 1 UDP 2113667327 192.168.1.5 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.142 54609 typ srflx raddr 192.168.1.5 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.5 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.142 64678 typ srflx raddr 192.168.1.5 rport 64678	**[RFC5245]**
a=ssrc-group:FID 54321 88776	**[RFC5888]**
a=ssrc:54321	**[RFC5576]**
a=ssrc:88776	**[RFC5576]**
a=recvonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=bundle-only	Add reference to unified plan when available

Table 21: 5.11 SDP Answer no Simulcast

5.11. Simulcast Video Stream with Forward Error Correction

This section shows an SDP Offer/Answer exchange for Simulcast video stream at two resolutions and and has **[RFC5956]** style FEC flows.

On completion of Offer/Answer exchange, one audio stream, 2 simulcast video streams and 2 associated FEC streams are sent over a single 5-Tuple as part of bundle-only and BUNDLE framework.

Simulcast Streams with Forward Error Correction

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=group:BUNDLE m0 m1	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 0 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m0	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:11111	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 0 RTP/SAVPF 98 100 101 103	bundle-only video line with port number set to zero
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m1	**[RFC5888]** Video m=line part of BUNDLE group
b=AS:2500	**[RFC4566]**
a=rtpmap:98 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:100 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:101 1d-interleaved-parityfec/90000	**[RFC5956]**
a=rtpmap:103 1d-interleaved-parityfec/90000	**[RFC5956]**
a=fmtp:98 max-fr=30;max-fs=8040	**[RFC4566]**

a=fmtp:100 max-fr=15;max-fs=1200	**[RFC4566]**
a=fmtp:101 L=5; D=10; repair-window=200000	**[RFC5956]**
a=fmtp:103 L=5; D=10; repair-window=200000	**[RFC5956]**
a=ssrc-group:SIMULCAST 12345 78990	Simulcast group
a=ssrc-group:FEC-FR 12345 34567	**[RFC5888]**
a=ssrc-group:FEC-FR 78990 90887	**[RFC5888]**
a=ssrc:12345	**[RFC5576]**
a=ssrc:78990	**[RFC5576]**
a=ssrc:34567	**[RFC5576]**
a=ssrc:90887	**[RFC5576]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=bundle-only	Add reference to unified plan when available

Table 22: 5.12 SDP Offer

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
a=group:BUNDLE m0 m1	**[draft-ietf-mmusic-sdp-bundle-negotiation]** Alice supports grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 0 109	**[RFC4566]**
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m0	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=ssrc:33333	**[RFC5576]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 54609 RTP/SAVPF 98 100 101 103	BUNDLE accepted with Bundle Address identical to audio m=line.
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m1	**[RFC5888]** Video m=line part of BUNDLE group
b=AS:2500	**[RFC4566]**
a=rtpmap:98 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:100 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:101 1d-interleaved-parityfec/90000	**[RFC5956]**
a=rtpmap:103 1d-interleaved-parityfec/90000	**[RFC5956]**
a=fmtp:98 max-fr=30;max-fs=8040	**[RFC4566]**
a=fmtp:100 max-fr=15;max-fs=1200	**[RFC4566]**
a=fmtp:101 L=5; D=10; repair-window=200000	**[RFC5956]**
a=fmtp:103 L=5; D=10; repair-window=200000	**[RFC5956]**
a=ssrc-group:SIMULCAST 54321 77656	Simulcast group
a=ssrc-group:FEC-FR 54321 88776	**[RFC5888]**

a=ssrc-group:FEC-FR 77656 12908	**[RFC5888]**
a=ssrc:54321	**[RFC5576]**
a=ssrc:77656	**[RFC5576]**
a=ssrc:88776	**[RFC5576]**
a=ssrc:12908	**[RFC5576]**
a=sendrecv	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=bundle-only	Add reference to unified plan when available

Table 23: 5.12 SDP Offer

6. WebRTC <-> Legacy Interop Examples

In this section, we attempt to provide session descriptions showcasing inter-operability between a WebRTC end-point and a Legacy VOIP end-point. The ideas included in here are not fully baked into the standards yet.
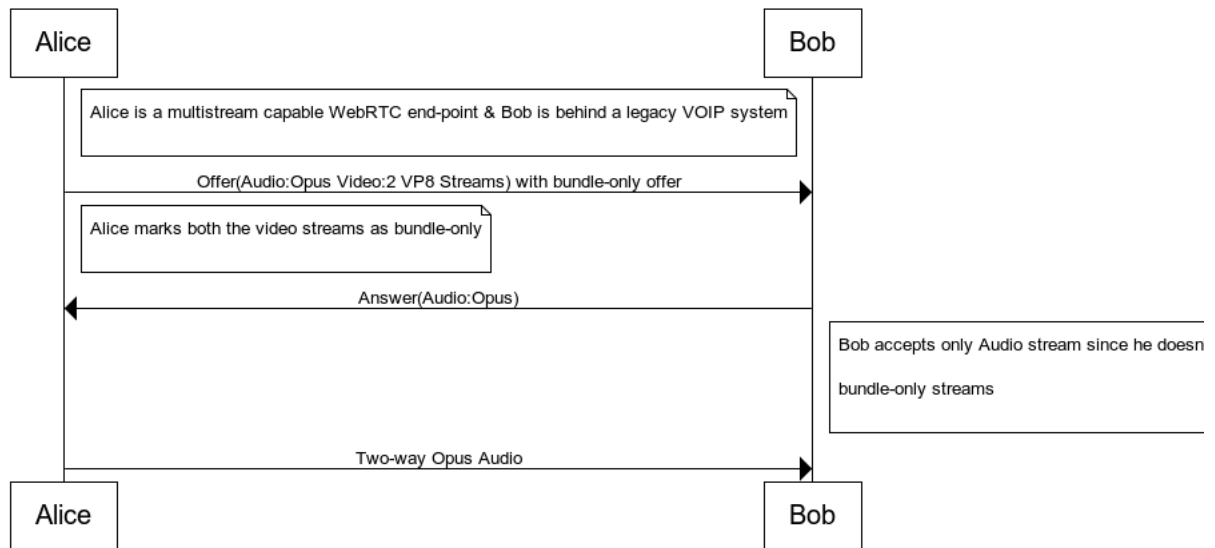
6.1. Successful legacy Interop Fallaback with bundle-only

In the scenario described below, Alice is a multi-stream capable WebRTC endpoint while Bob is a legacy VOIP end-point. The SDP Offer/Answer exchange demonstrates successful session setup with fallback to audio only stream negotiated via bundle-only framework between the end-points. Specifically,

- Offer from Alice describes 2 cameras via 2 video m=lines with both marked as bundle-only.
- Since Bob doesnot recognize either BUNDLE mechanism or bundle-only attribute, he accepts only the audio stream from Alice.

Successful 2-Way WebRTC <-> VOIP Interop

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**
s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:074c6550	**[RFC5245]**
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a068	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
	[draft-ietf-mmusic-sdp-bundle-negotiation] Alice supports

SDP Contents	RFC#/Notes
a=group:BUNDLE m0 m1 m2	grouping of m=lines under BUNDLE semantics
m=audio 54609 RTP/SAVPF 0 109	**[RFC4566]**
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m0	**[RFC5888]** Audio m=line part of BUNDLE group with a unique port number
a=msid:ma ta	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (ta)
a=rtpmap:0 PCMU/8000	**[RFC3551]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]**
a=rtcp-mux	**[RFC5761]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 0 RTP/SAVPF 98 100	bundle-only video line with port number set to zero
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m1	**[RFC5888]** Video m=line part of BUNDLE group
a=msid:ma tb	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tb)
a=rtpmap:98 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:100 VP8/90000	**[draft-ietf-payload-vp8]**
a=imageattr:98 [x=1280,y=720]	**[RFC6236]** Camera-1,Encoding-1 Resolution
a=fmtp:98 max-fr=30	**[RFC4566]**
a=imageattr:100 [x=640,y=480]	**[RFC6236]** Camera-1,Encoding-2 Resolution
a=fmtp:100 max-fr=15	**[RFC4566]**
a=ssrc-group:SIMULCAST 12345 45678	**[RFC5576]**
a=ssrc:12345	**[RFC5576]**
a=ssrc:45678	**[RFC5576]**
a=ssrc:12345 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-1 SSRC with Session CNAME
a=ssrc:45678 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-2 SSRC with Session CNAME
a=bundle-only	Add reference to unified plan when available
m=video 0 RTP/SAVPF 101 103	bundle-only video line with port number set to zero
c= IN IP4 24.23.204.141	**[RFC4566]**
a=mid:m2	**[RFC5888]** Video m=line part of BUNDLE group
a=msid:ma tc	Identifies RTCMediaStream ID (ma) and RTCMediaStreamTrack ID (tc)
a=rtpmap:101 H264/90000	**[RFC3984]**
a=rtpmap:103 H264/90000	**[RFC3984]**
a=fmtp:101 profile-level-id=4d0028;packetization-mode=1;max-fr=30	**[RFC3984]** Camera-2,Encoding-1 Resolution
a=fmtp:100 profile-level-id=4d0028;packetization-mode=1;max-fr=15	**[RFC3984]** Camera-1,Encoding-2 Resolution
a=ssrc:67890	**[RFC5576]**
a=ssrc:56789	**[RFC5576]**
a=ssrc-group:SIMULCAST 67890 56789	**[RFC5576]**
a=ssrc:67890 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-1 SSRC with Session CNAME
a=ssrc:56789 cname:axzo1278npDlAzM73	**[RFC5576] [draft-rescorla-avtcore-6222bis]** Camera-1,Encoding-2 SSRC with Session CNAME
a=bundle-only	Add reference to unified plan when available

Table 24: 6.1 SDP Simulcast bundle-only

SDP Contents	RFC#/Notes
v=0	**[RFC4566]**
o=alice 20519 0 IN IP4 0.0.0.0	**[RFC4566]**

s=	**[RFC4566]**
t=0 0	**[RFC4566]**
a=ice-ufrag:ufrag:c300d85b	**[RFC5245]**
a=ice-pwd:de4e99bd291c325921d5d47efbabd9a2	**[RFC5245]**
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:70:9d:1f:66:79:a8:07	**[RFC5245]**
m=audio 54609 RTP/SAVPF 109	**[RFC4566]**
c= IN IP4 24.23.204.141	**[RFC4566]**
a=rtpmap:109 opus/48000/2	**[draft-ietf-payload-rtp-opus]**
a=ptime:20	**[draft-ietf-payload-rtp-opus]**
a=sendonly	**[RFC3264]**
a=candidate:0 1 UDP 2113667327 192.168.1.4 54609 typ host	**[RFC5245]**
a=candidate:1 1 UDP 694302207 24.23.204.141 54609 typ srflx raddr 192.168.1.4 rport 54609	**[RFC5245]**
a=candidate:0 2 UDP 2113667326 192.168.1.4 64678 typ host	**[RFC5245]**
a=candidate:1 2 UDP 1694302206 24.23.204.141 64678 typ srflx raddr 192.168.1.4 rport 64678	**[RFC5245]**
m=video 0 RTP/SAVPF 98 100	Bob doesn't recognize bundle-only and hence rejects the video stream
c= IN IP4 24.23.204.141	**[RFC4566]**
a=rtpmap:98 VP8/90000	**[draft-ietf-payload-vp8]**
a=rtpmap:100 VP8/90000	**[draft-ietf-payload-vp8]**
a=imageattr:98 [x=1280,y=720]	**[RFC6236]**Camera-1,Encoding-1 Resolution
a=fmtp:98 max-fr=30	**[RFC4566]**
a=imageattr:100 [x=640,y=480]	**[RFC6236]** Camera-1,Encoding-2 Resolution
a=fmtp:100 max-fr=15	**[RFC4566]**
m=video 0 RTP/SAVPF 98 100	Bob doesn't recognize bundle-only and hence rejects the video stream
c= IN IP4 24.23.204.141	**[RFC4566]**
a=rtpmap:101 H264/90000	**[RFC3984]**
a=rtpmap:103 H264/90000	**[RFC3984]**
a=fmtp:101 profile-level-id=4d0028;packetization-mode=1;max-fr=30	**[RFC3984]**Camera-2,Encoding-1 Resolution
a=fmtp:100 profile-level-id=4d0028;packetization-mode=1;max-fr=15	**[RFC3984]**Camera-1,Encoding-2 Resolution

Table 25: 6.2 SDP Answer

---

7. IANA Considerations

This document requires no actions from IANA.

---

8. Change Log

[RFC EDITOR NOTE: Please remove this section when publishing]

Changes from draft-nandakumar-rtcweb-sdp-01

- Updated references to OPUS RTP Payload Specification.
- Updated BUNDLE examples based on the latest draft-ietf-mmusic-sdp-bundle-negotiation.
- Added examples for multiple audio and video flows based on Unified Plan.
- Added new examples for RTX and FEC streams
- Updated Simulcast and SVC examples

Changes from draft-nandakumar-rtcweb-sdp-00

- Fixed editorial comments on the mailing list.
- Updated Data-channel SDP information based on draft-ietf-mmusic-sctp-sdp.
- Updated BUNDLE examples based on draft-ietf-mmusic-sdp-bundle-negotiation.
- Added examples for few more BUNDLE variants
- Added new examples for Simulcast and SVC

---

9. References

9.1. Normative References

[RFC3264]   Rosenberg, J. and H. Schulzrinne, "**An Offer/Answer Model with Session Description Protocol (SDP)**," RFC 3264, June 2002 (**TXT**).

[RFC4566]   Handley, M., Jacobson, V., and C. Perkins, "**SDP: Session Description Protocol**," RFC 4566, July 2006 (**TXT**).

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

9.2. Informative References

[RFC5245]   Rosenberg, J., "**Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols**," RFC 5245, July 2006 (**TXT**).

[WEBRTC]   W3C, "WebRTC 1.0: Real-time Communication Between Browsers," <http://dev.w3.org/2011/webrtc/editor/webrtc.html> .

[JSEP]   Uberti, J. and C. Jennigs, "**Javascript Session Establishment Protocol**," draft-ietf-rtcweb-jsep-01 (work in progress), December 2012 (**TXT**).

[RFC5506]   Johansson, I., "**Support for Reduced-Size Real-Time Transport Control Protocol (RTCP): Opportunities and Consequences**," RFC 5506, April 2009 (**TXT**).

[RFC3551]   Schulzrinne, H. and S. Casner, "**RTP Profile for Audio and Video Conferences with Minimal Control**," RFC 3551, July 2003 (**TXT**).

[RFC3952]   Duric, A. and S. Andersen, "**Real-time Transport Protocol (RTP) Payload Format for internet Low Bit Rate Codec (iLBC) Speech**," RFC 3952, December 2004 (**TXT**).

[RFC4796]   Hautakorpi, J. and G. Camarillo, "**The Session Description Protocol (SDP) Content Attribute**," RFC 4796, February 2007 (**TXT**).

[RFC5761]   Perkins, C. and M. Westerlund, "**Multiplexing RTP Data and Control Packets on a Single Port**," RFC 5761, April 2010 (**TXT**).

[RFC3556]   Casner, S., "**Session Description Protocol (SDP) Bandwidth Modifiers for RTP Control Protocol (RTCP) Bandwidth**," RFC 3556, July 2003 (**TXT**).

[RFC5104]   Wenger, S., Chandra, U., Westerlund, M., and B. Burman, "**Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF)**," RFC 5104, February 2008 (**TXT**).

[RFC4588]   Rey, J., Leon, D., Miyazaki, A., Varsa, V., and R. Hakenberg, "**RTP Retransmission Payload Format**," RFC 4588, July 2006 (**TXT**).

[RFC5956]   Begen, A., "**Forward Error Correction Grouping Semantics in the Session Description Protocol**," RFC 5956, September 2010 (**TXT**).

[RFC5888]   Camarillo, G. and H. Schulzrinne, "**RTP Payload Format for H.264 Video**," RFC 5888, June 2010 (**TXT**).

[RFC6236]   Johansson, I. and K. Jung, "**Negotiation of Generic Image Attributes in the Session Description Protocol (SDP)**," RFC 6236, May 2011 (**TXT**).

[draft-ietf-payload-rtp-opus]   Spittka, J., Vos, K., and JM. Valin, "**RTP Payload Format for Opus Speech and Audio Codec**," draft-ietf-payload-rtp-opus-00 (work in progress), July 2012 (**TXT**).

[draft-ietf-payload-vp8]   Westin, P., Lundin, H., Glover, M., Uberti, J., and F. Galligan, "**RTP Payload Format for VP8 Video**," draft-ietf-payload-vp8-05 (work in progress), August 2012 (**TXT**).

[RFC3984]   Wenger, S., Hannuksela, M., Stockhammer, T., Westerlund, M., and D. Singer, "**RTP Payload Format for H.264 Video**," RFC 3984, February 2005 (**TXT**).

[RFC5583]   Schierl, T. and S. Wenger, "**Signaling Media Decoding Dependency in the Session Description Protocol (SDP)**," RFC 5583, July 2009 (**TXT**).

[RFC5576]   Lennox, J., Ott, J., and T. Schierl, "**Source-Specific Media Attributes in the Session Description Protocol (SDP)**," RFC 5576, June 2009 (**TXT**).

[draft-ietf-rtcweb-data-channel]   Jesup, R., Loreto, S., and M. Tuexen, "**RTCWeb Datagram Connection**," draft-ietf-rtcweb-data-channel-01 (work in progress), September 2012 (**TXT**).

[draft-ietf-mmusic-sctp-sdp]   Loreto, S. and G. Camarillo, "**Stream Control Transmission Protocol (SCTP)-Based Media Transport in the Session Description Protocol (SDP)**," draft-ietf-mmusic-sctp-sdp-03 (work in progress), September 2012 (**TXT**).

[draft-ietf-mmusic-sdp-bundle-negotiation]   Holmberg, C., Alvestrand, H., Glover, M., and C. Jennings, "**Multiplexing Negotiation Using Session Description Protocol (SDP) Port Numbers**," draft-ietf-mmusic-sdp-bundle-negotiation-04 (work in progress), February 2013 (**TXT**).

[draft-lennox-mmusic-sdp-source-selection]   Lennox, J. and H. Schulzrinne, "**Multiplexing Negotiation Using Session Description Protocol (SDP) Port Numbers**," draft-lennox-mmusic-sdp-source-selection-05 (work in progress), October 2012 (**TXT**).

[draft-rescorla-avtcore-6222bis]   Rescorla, E. and A. Begen, "**Guidelines for Choosing RTP Control Protocol (RTCP) Canonical Names (CNAMEs)**," draft-rescorla-avtcore-6222bis-00 (work in progress), October 2012 (**TXT**).

Authors' Addresses

Suhas Nandakumar
Cisco
170 West Tasman Drive
San Jose, CA 95134
USA
Email: snandaku@cisco.com

Cullen Jennings
Cisco
170 West Tasman Drive
San Jose, CA 95134
USA
Phone: +1 408 421-9990
Email: fluffy@cisco.com