

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 18, 2013

S. Nandakumar
C. Jennings
Cisco
October 15, 2012

SDP for the WebRTC
draft-nandakumar-rtcweb-sdp-00

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. Within the WebRTC framework, Session Description protocol (SDP) [[RFC4566](#)] is used for negotiating session capabilities between the peers. Such a negotiation happens based on the SDP Offer/Answer exchange mechanism described in the [RFC 3264](#) [[RFC3264](#)].

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

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 <http://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 April 18, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1. Introduction](#) [3](#)
- [2. Terminology](#) [3](#)
- [3. SDP and the WebRTC](#) [3](#)
- [4. Offer/Answer and the WebRTC](#) [4](#)
- [5. WebRTC Session Description Examples](#) [5](#)
 - [5.1. Secure Two-Way Audio,Video and Data with RTCP Feedback . .](#) [5](#)
 - [5.2. Secure Two-way Audio,Video,Data and remove data stream . .](#) [11](#)
 - [5.3. Secure Two-Way Audio,Video w/Bundle](#) [15](#)
 - [5.4. Successful One Way Session with 2 Video Streams](#) [19](#)
 - [5.5. Add New Media \(video\)](#) [23](#)
- [6. WebRTC <-> Legacy Interop Examples](#) [28](#)
 - [6.1. Secure Two-Way Audio,Video w/Feedback - WebRTC <-> Legacy Interop](#) [28](#)
- [7. IANA Considerations](#) [33](#)
- [8. References](#) [33](#)
 - [8.1. Normative References](#) [33](#)
 - [8.2. Informative References](#) [33](#)
- [Authors' Addresses](#) [34](#)

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 provide 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 the [[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 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 attempt 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.

```

title WebRTC Session - 2-Way Secure Audio, Video with RTCP Feedback
Alice->Bob: Offer(Audio:G.711,Opus,iLBC Video:H.264,VP8)
Bob->Alice: Answer(Audio:Opus,DTMF Video:H.264)
Alice->Bob: Two-way Opus Audio, H.264 Video
note right of Alice
  Session also supports RTP/RTCP Mux, RTCP feedback (nack,pli)
end note

```

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 (4.1 and 4.2) below capture in detail, the initial SDP Offer and Answer messages exchanged.

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:c7:7 0:9d: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]
c= IN IP4 24.23.204.141	[RFC4566]
a=rtpmap:0 PCMU/8000	[RFC3551]
a=rtpmap:109 opus/48000	[draft-spittka-payload -rtp-opus]
a=ptime:20	[draft-spittka-payload -rtp-opus]
a=rtpmap:98 iLBC/8000 a=fmtp:98 mode=20	[RFC3952]
a=sendrecv	[RFC3264] - Alice can send and recv audio
a=rtcp-mux	[RFC5761] - Alice can perform RTP/RTCP Muxing on port 54609
b=AS:256	[RFC4566]
b=RS:0	[RFC3556]
b=RR:0	[RFC3556]
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]
c= IN IP4 24.23.204.141	[RFC4566]
a=rtpmap:99 H264/90000	[RFC3984]
a=fmtp:99	[RFC3984]
profile-level-id=4d0028;packetization-mode=1	
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp8]
	8]
a=sendrecv	[RFC3264] - Alice can
	send and recv video
a=rtcp-mux	[RFC5761] - Alice can
	perform RTP/RTCP
	Muxing on port 62537
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.4 62537 typ host	
a=candidate:1 1 UDP 1694302207	[RFC5245]
24.23.204.141 62537 typ srflx raddr	
192.168.1.4 rport 62537	
a=candidate:0 2 2113667326 192.168.1.4	[RFC5245]
54721 typ host	
a=candidate:1 2 UDP 1694302206	[RFC5245]
24.23.204.141 54721 typ srflx raddr	
192.168.1.4 rport 54721	
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 SCTP/DTLS 5000	[draft-ietf-rtcweb-dat]
	a-channel]
c= IN IP4 24.23.204.141	[RFC4566]
a=fmtp:5000	[draft-ietf-rtcweb-dat]
protocol=webrtc-datachannel;streams=16	a-channel]

a=sendrecv	[RFC3264] - Alice can
	send and recv
	non-media data
a=rtcp-mux	[RFC5761] - Alice can
	perform RTP/RTCP
	Muxing on port 56966
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.7 56966 typ host	
a=candidate:1 1 UDP 1694302207	[RFC5245]
24.23.204.141 56966 typ srflx raddr	
192.168.1.7 rport 56966	
a=candidate:0 2 UDP 2113667326	[RFC5245]
192.168.1.7 51641 typ host	
a=candidate:1 2 UDP 1694302206	[RFC5245]
24.23.204.141 51641 typ srflx raddr	
192.168.1.7 rport 51641	
a=rtcp-fb:5000 nack	[RFC5104] - Indicates
	NACK feedback support
	for the data-channel
+-----+-----+	

Table 1: 4.1 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-199:41:49:83:4a: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]
c= IN IP4 98.248.92.77	[RFC4566]
a=rtpmap:109 opus/48000	[draft-spittka-payload -rtp-opus]
a=ptime:20	[draft-spittka-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
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: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]
c= IN IP4 98.248.92.771	[RFC4566]
a=rtpmap:99 H264/90000	[RFC3984]

a=fmtp:99	[RFC3984]	
profile-level-id=4d0028;packetization-mo		
de=1		
a=sendrecv	[RFC3264] - Bob can	
	send and recv video	
a=rtcp-mux	[RFC5761] - Bob can	
	perform RTP/RTCP	
	Muxing on port 63130	
a=candidate:0 1 UDP 2113667327	[RFC5245]	
192.168.1.7 63130 typ host		
a=candidate:1 1 UDP 1694302207	[RFC5245]	
98.248.92.77 63130 typ srflx raddr		
192.168.1.7 rport 63130		
a=candidate:0 2 UDP 2113667326	[RFC5245]	
192.168.1.7 56607 typ host		
a=candidate:1 2 UDP 1694302206	[RFC5245]	
98.248.92.77 56607 typ srflx raddr		
192.168.1.7 rport 56607		
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 SCTP/DTLS 5000	[draft-ietf-rtcweb-dat]	
	a-channel]	
c= IN IP4 98.248.92.771	[RFC4566]	
a=fmtp:5000	[draft-ietf-rtcweb-dat]	
protocol=webrtc-datachannel;streams=16	a-channel]	
a=sendrecv	[RFC3264] - Bob can	
	send and recv	
	non-media data	
a=rtcp-mux	[RFC5761] - Bob can	
	perform RTP/RTCP	
	Muxing on port 55700	
a=candidate:0 1 UDP 2113667327	[RFC5245] - Refer 4.1	
192.168.1.7 55700 typ host	SDP Offer	
a=candidate:1 1 UDP 1694302207	[RFC5245] Refer 4.1	
98.248.92.77 55700 typ srflx raddr	SDP Offer	
192.168.1.7 rport 55700		
a=candidate:0 2 UDP 2113667326	[RFC5245] Refer 4.1	
192.168.1.7 58137 typ host	SDP Offer	
a=candidate:1 2 UDP 1694302206	[RFC5245] Refer 4.1	
98.248.92.77 58137 typ srflx raddr	SDP Offer	
192.168.1.7 rport 581371		

a=rtcp-fb:5000 nack	[RFC5104] - Indicates
	NACK feedback support
	for the data-channel
+-----+-----+	

Table 2: 4.1 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.

```

title WebRTC Session (Audio,Video,Datachannel) - Drop Datachannel
note right of Alice
    Alice & Bob are in a two-way audio,video and datachannel session.
    Alice decides to stop the datachannel stream
end note
Alice->Bob: Offer(Audio:Opus Video:VP8, Application: Drop)
Bob->Alice: Answer(Audio:Opus Video:VP8, Application:Drop)
Alice->Bob: Two-way Opus Audio and VP8 Video
    
```

As a precondition, A Two-Way Audio,Video and Data Session is already setup.

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:c7:7 0:9d:1f:66:79:a8:07	[RFC5245]
a=rtcp-rsize	[RFC5506]
m=audio 54609 RTP/SAVPF 0 109 98	[RFC4566]
c= IN IP4 24.23.204.141	[RFC4566]
a=rtpmap:0 PCMU/8000	[RFC3551]
a=rtpmap:109 opus/48000	[draft-spittka-payload]
	-rtp-opus]
a=ptime:20	[draft-spittka-payload]
	-rtp-opus]
a=rtpmap:98 iLBC/8000 a=fmtp:98 mode=20	[RFC3952]
a=sendrecv	[RFC3264]

a=rtcp-mux	[RFC5761]	
a=candidate:0 1 UDP 2113667327	[RFC5245]	
192.168.1.4 54609 typ host		
a=candidate:1 1 UDP 694302207	[RFC5245]	
24.23.204.141 54609 typ srflx raddr		
192.168.1.4 rport 54609		
a=candidate:0 2 UDP 2113667326	[RFC5245]	
192.168.1.4 64678 typ host		
a=candidate:1 2 UDP 1694302206	[RFC5245]	
24.23.204.141 64678 typ srflx raddr		
192.168.1.4 rport 64678		
a=rtcp-fb:109 nack	[RFC5104]	
m=video 62537 RTP/SAVPF 99 120	[RFC4566]	
c= IN IP4 24.23.204.141	[RFC4566]	
a=rtpmap:99 H264/90000	[RFC3984]	
a=fmtp:99	[RFC3984]	
profile-level-id=4d0028;packetization-mode=1		
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp	
	8]	
a=sendrecv	[RFC3264]	
a=rtcp-mux	[RFC5761]	
a=candidate:0 1 UDP 2113667327	[RFC5245]	
192.168.1.4 62537 typ host		
a=candidate:1 1 UDP 1694302207	[RFC5245]	
24.23.204.141 62537 typ srflx raddr		
192.168.1.4 rport 62537		
a=candidate:0 2 2113667326 192.168.1.4	[RFC5245]	
54721 typ host		
a=candidate:1 2 UDP 1694302206	[RFC5245]	
24.23.204.141 54721 typ srflx raddr		
192.168.1.4 rport 54721		
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 SCTP/DTLS 5000	[draft-ietf-rtcweb-dat	
	a-channel] - Port 0	
	indicates dropping	
	data stream	
c= IN IP4 24.23.204.141	[RFC4566]	
a=fmtp:5000	[draft-ietf-rtcweb-dat	
protocol=webrtc-datachannel;streams=16	a-channel]	
a=sendrecv	[RFC3264]	
a=rtcp-mux	[RFC5761]	
a=candidate:0 1 UDP 2113667327	[RFC5245]	
192.168.1.7 56966 typ host		

a=candidate:1 1 UDP 1694302207	[RFC5245]	
24.23.204.141 56966 typ srflx raddr		
192.168.1.7 rport 56966		
a=candidate:0 2 UDP 2113667326	[RFC5245]	
192.168.1.7 51641 typ host		
a=candidate:1 2 UDP 1694302206	[RFC5245]	
24.23.204.141 51641 typ srflx raddr		
192.168.1.7 rport 51641		
a=rtcp-fb:5000 nack	[RFC5104]	
+-----+-----+-----+-----+-----+		

Table 3: 4.2 SDP Updated Offer w/DataChannel Drop

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:de4e99bd291c325921d5d47efbabd9	[RFC5245]
a2	
a=fingerprint:sha-1	[RFC5245]
99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:7	
0:9d:1f:66:79:a8:07	
a=rtcp-rsize	[RFC5506]
m=audio 49203 RTP/SAVPF 109	[RFC4566]
c= IN IP4 98.248.92.77	[RFC4566]
a=rtpmap:109 opus/48000	[draft-spittka-payload
	-rtp-opus]
a=ptime:20	[draft-spittka-payload
	-rtp-opus]
a=sendrecv	[RFC3264]
a=rtcp-mux	[RFC5761]
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.7 49203 typ host	
a=ccandidate:1 1 UDP 1694302207	[RFC5245]
98.248.92.77 49203 typ srflx raddr	
192.168.1.7 rport 49203	
a=candidate:candidate:0 2 UDP 2113667326	[RFC5245]
192.168.1.7 60065 typ host	
a=candidate:1 2 UDP 1694302206	[RFC5245]
98.248.92.77 60065 typ srflx raddr	
192.168.1.7 rport 60065	
m=video 63130 RTP/SAVPF 99 120	[RFC4566]
c= IN IP4 98.248.92.771	[RFC4566]
a=rtpmap:99 H264/90000	[RFC3984]
a=fmtp:99	[RFC3984]
profile-level-id=4d0028;packetization-mo	
de=1	
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp
	8]
a=sendrecv	[RFC3264]
a=rtcp-mux	[RFC5761]
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.7 63130 typ host	
a=candidate:1 1 UDP 1694302207	[RFC5245]
98.248.92.77 63130 typ srflx raddr	
192.168.1.7 rport 63130	

a=candidate:0 2 UDP 2113667326	[RFC5245]	
192.168.1.7 56607 typ host		
a=candidate:1 2 UDP 1694302206	[RFC5245]	
98.248.92.77 56607 typ srflx raddr		
192.168.1.7 rport 56607		
a=rtcp-fb:99 nack pli	[RFC5104]	
a=rtcp-fb:99 ccm fir	[RFC5104]	
m=application 0 SCTP/DTLS 5000	[draft-ietf-rtcweb-dat	
	a-channel] Bob accepts	
	dropping the data	
	stream	
c= IN IP4 98.248.92.771	[RFC4566]	
a=fmtp:5000	[draft-ietf-rtcweb-dat	
protocol=webrtc-datachannel;streams=16	a-channel]	
a=sendrecv	[RFC3264]	
a=rtcp-mux	[RFC5761]	
a=candidate:0 1 UDP 2113667327	[RFC5245]	
192.168.1.7 55700 typ host		
a=candidate:1 1 UDP 1694302207	[RFC5245]	
98.248.92.77 55700 typ srflx raddr		
192.168.1.7 rport 55700		
a=candidate:0 2 UDP 2113667326	[RFC5245]	
192.168.1.7 58137 typ host		
a=candidate:1 2 UDP 1694302206	[RFC5245]	
98.248.92.77 58137 typ srflx raddr		
192.168.1.7 rport 581371		
a=rtcp-fb:5000 nack	[RFC5104]	
+-----+-----+-----+		

Table 4: 4.2 SDP Updated Answer

5.3. Secure Two-Way Audio, Video w/Bundle

This use-case demonstrates a successful audio and video multiplexing scenario with SDP Bundle negotiation.

The semantics of group:BUNDLE attribute would be defined for two different scenarios as follows

Receiver Supports Bundle: In this case, the receiver places a group:BUNDLE line in the answer and it sends all the media to the mids in the BUNDLE group to the port number identified for the m-line corresponding to the first mid in the BUNDLE group. Thus, Alice receives Audio and Video multiplexed onto the port 54609 as per the example below.

Receiver Doesn't Support Bundle: In this case, the receiver doesn't include a group:BUNDLE attribute and audio/video RTP streams are sent to the appropriate ports specified in the offer.

title WebRTC Session - 2-Way Secure Audio,Video with Bundle

Alice->Bob: Offer(Audio:Opus Video:VP8)

Bob->Alice: Answer(Audio:Opus Video:VP8)

Alice->Bob: Two-way Opus Audio, H.264 Video

note right of Alice

Session also supports RTP/RTCP Mux, RTCP feedback (nack,pli) with multiplexing of audio and video RTP data.

end note

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:c7:7 0:9d:1f:66:79:a8:07	[RFC5245]
a=rtcp-rsize	[RFC5506]
a=group:BUNDLE foo bar	[RFC5888]Alice supports grouping of m= lines
m=audio 54609 RTP/SAVPF 109	[RFC4566]
c= IN IP4 24.23.204.141	[RFC4566]
a=mid:foo	[RFC5888]Audio m= line part of Bundle group
a=rtpmap:109 opus/48000	[draft-spittka-payload -rtp-opus]
aptime:20	[draft-spittka-payload -rtp-opus]
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 120	[RFC4566]

c= IN IP4 24.23.204.141	[RFC4566]	
a=mid:bar	[RFC5888]	Video m=line
		part of the Bundle
		group
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp	
	8]	
a=sendrecv	[RFC3264]	
a=rtcp-mux	[RFC5761]	
a=candidate:0 1 UDP 2113667327	[RFC5245]	
192.168.1.4 62537 typ host		
a=candidate:1 1 UDP 1694302207	[RFC5245]	
24.23.204.141 62537 typ srflx raddr		
192.168.1.4 rport 62537		
a=candidate:0 2 2113667326 192.168.1.4	[RFC5245]	
54721 typ host		
a=candidate:1 2 UDP 1694302206	[RFC5245]	
24.23.204.141 54721 typ srflx raddr		
192.168.1.4 rport 54721		
a=rtcp-fb:120 nack pli	[RFC5104]	
a=rtcp-fb:120 ccm fir	[RFC5104]	
+-----+-----+		

Table 5: 4.3 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:de4e99bd291c325921d5d47efbabd9	[RFC5245]
a2	
a=fingerprint:sha-1	[RFC5245]
99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:7	
0:9d:1f:66:79:a8:07	
a=rtcp-rsize	[RFC5506]
a=group:BUNDLE foo bar	Bob supports grouping of m= lines and indicates his interest in the same as well
	[RFC5888]
m=audio 49203 RTP/SAVPF 109	[RFC4566]
c= IN IP4 98.248.92.77	[RFC4566]
a=mid:foo	[RFC5888]Audio m=line part of the Bundle group
a=rtpmap:109 opus/48000	[draft-spittka-payload-rtp-opus]
a=ptime:20	[draft-spittka-payload-rtp-opus]
a=sendrecv	[RFC3264]
a=rtcp-mux	[RFC5761]
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.7 49203 typ host	
a=candidate:1 1 UDP 1694302207	[RFC5245]
98.248.92.77 49203 typ srflx raddr	
192.168.1.7 rport 49203	
a=candidate:candidate:0 2 UDP 2113667326	[RFC5245]
192.168.1.7 60065 typ host	
a=candidate:1 2 UDP 1694302206	[RFC5245]
98.248.92.77 60065 typ srflx raddr	
192.168.1.7 rport 60065	
m=video 63130 RTP/SAVPF 120	[RFC4566]
c= IN IP4 98.248.92.771	[RFC4566]
a=mid:bar	[RFC5888]Video m=line part of the Bundle group
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp8]
a=sendrecv	[RFC3264]

a=rtcp-mux	[RFC5761]	
a=candidate:0 1 UDP 2113667327	[RFC5245]	
192.168.1.7 63130 typ host		
a=candidate:1 1 UDP 1694302207	[RFC5245]	
98.248.92.77 63130 typ srflx raddr		
192.168.1.7 rport 63130		
a=candidate:0 2 UDP 2113667326	[RFC5245]	
192.168.1.7 56607 typ host		
a=candidate:1 2 UDP 1694302206	[RFC5245]	
98.248.92.77 56607 typ srflx raddr		
192.168.1.7 rport 56607		
+-----+-----+-----+		

Table 6: 4.3 SDP Answer w/Bundle

5.4. 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.

```

title 1 Way Audio & Video w/2 Video Streams
note right of Alice
Alice offers 2 sendonly video streams
one for her video feed and other for her presentation slides.
end note
Alice->Bob: Offer(Audio:Opus, Video1,2: VP8)
note right of Bob
Bob accepts Alice's offer
end note
Bob->Alice: Answer(Audio:Opus, Video1,2: VP8)
Alice->Bob: One-way Opus Audio, VP8 Video
note right of Alice
Bob can hear Alice and see her video feed as well
as her presentation slides.
end note

```


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:c7:7 0:9d:1f:66:79:a8:07	[RFC5245]
a=rtcp-rsize	[RFC5506]
m=audio 54609 RTP/SAVPF 109	[RFC4566]
c= IN IP4 24.23.204.141	[RFC4566]
a=rtpmap:109 opus/48000	[draft-spittka-payload
	-rtp-opus]
a=ptime:20	[draft-spittka-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	[RFC5245]
a=rtcp-fb:109 nack	[RFC5104]
m=video 62537 RTP/SAVPF 120	[RFC4566]
c= IN IP4 24.23.204.141	[RFC4566]
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp
	8]
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]
a=rtcp-fb:120 nack pli	[RFC5104]
a=rtcp-fb:120 ccm fir	[RFC5104]
m=video 62539 RTP/SAVPF 120	[RFC4566]
c= IN IP4 24.23.204.141	[RFC4566]
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp
	8]

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	[RFC5245]	
192.168.1.4 62539 typ host		
a=candidate:0 2 UDP 2113667326	[RFC5245]	
192.168.1.4 54723 typ host		
a=rtcp-fb:120 nack pli	[RFC5104]	
a=rtcp-fb:120 ccm fir	[RFC5104]	
+-----+-----+		

Table 7: 4.4 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:de4e99bd291c325921d5d47efbabd9	[RFC5245]
a2	
a=fingerprint:sha-1	[RFC5245]
99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:7	
0:9d:1f:66:79:a8:07	
a=rtcp-rsize	[RFC5506]
m=audio 49203 RTP/SAVPF 109	[RFC4566]
c= IN IP4 98.248.92.77	[RFC4566]
a=rtpmap:109 opus/48000	[draft-spittka-payload
	-rtp-opus]
a=ptime:20	[draft-spittka-payload
	-rtp-opus]
a=recvonly	[RFC3264] - Receive
	only audio stream
a=rtcp-mux	[RFC5761]
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.7 49203 typ host	
a=candidate:0 2 UDP 2113667326	[RFC5245]
192.168.1.7 60065 typ host	
m=video 63130 RTP/SAVPF 120	[RFC4566]
c= IN IP4 98.248.92.771	[RFC4566]
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp
	8]
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	[RFC5245]
192.168.1.7 63130 typ host	
a=candidate:0 2 UDP 2113667326	[RFC5245]
192.168.1.7 56607 typ host	
a=rtcp-fb:120 nack pli	[RFC5104]
a=rtcp-fb:120 ccm fir	[RFC5104]
m=video 63133 RTP/SAVPF 120	[RFC4566]
c= IN IP4 98.248.92.771	[RFC4566]
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp
	8]
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	[RFC5245]	
192.168.1.7 63133 typ host		
a=candidate:0 2 UDP 2113667326	[RFC5245]	
192.168.1.7 56609 typ host		
a=rtcp-fb:120 nack pli	[RFC5104]	
a=rtcp-fb:120 ccm fir	[RFC5104]	
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+		

Table 8: 4.4 SDP Answer

5.5. Add New Media (video)

This scenario describes the message exchanges when Alice decides to add video to an existing audio-only session

```

title Add New Media(Video)
Alice->Bob: Offer(Audio:G.711,Opus,iLBC)
Bob->Alice: Answer(Audio:Opus)
Alice->Bob: Two-way Opus Audio
note right of Alice
Alice decides to add Video
end note
Alice->Bob: Offer(Audio:G.711,Opus,iLBC Video:VP8)
Bob->Alice: Answer(Audio:Opus, Video:VP8)
Alice->Bob: Two-way Opus Audio, VP8 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:c7:7 0:9d:1f:66:79:a8:07	[RFC5245]
a=rtcp-rsize	[RFC5506]
m=audio 54609 RTP/SAVPF 0 109 98	[RFC4566]
c= IN IP4 24.23.204.141	[RFC4566]
a=rtpmap:0 PCMU/8000	[RFC3551]
a=rtpmap:109 opus/48000	[draft-spittka-payload]
	-rtp-opus]
a=ptime:20	[draft-spittka-payload]
	-rtp-opus]
a=rtpmap:98 iLBC/8000 a=fmtp:98 mode=20	[RFC3952]
a=sendrecv	[RFC3264]
a=rtcp-mux	[RFC5761]
b=AS:256	[RFC4566]
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]

Table 9: 4.5 SDP Initial Audio Only 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:de4e99bd291c325921d5d47efbabd9	[RFC5245]
a2	
a=fingerprint:sha-1	[RFC5245]
99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:7	
0:9d:1f:66:79:a8:07	
a=rtcp-rsize	[RFC5506]
m=audio 49203 RTP/SAVPF 109	[RFC4566]
c= IN IP4 98.248.92.77	[RFC4566]
a=rtpmap:109 opus/48000	[draft-spittka-payload
	-rtp-opus]
a=ptime:20	[draft-spittka-payload
	-rtp-opus]
a=sendrecv	[RFC3264]
a=rtcp-mux	[RFC5761]
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.7 49203 typ host	
a=ccandidate:1 1 UDP 1694302207	[RFC5245]
98.248.92.77 49203 typ srflx raddr	
192.168.1.7 rport 49203	
a=candidate:candidate:0 2 UDP 2113667326	[RFC5245]
192.168.1.7 60065 typ host	
a=candidate:1 2 UDP 1694302206	[RFC5245]
98.248.92.77 60065 typ srflx raddr	
192.168.1.7 rport 60065	

Table 10: 4.5 SDP Answer- Audio Only

Alice decides to add Video to the current session

SDP Contents	RFC#/Notes
v=0	[RFC4566]
o=alice 20520 0 IN IP4 0.0.0.0	[RFC4566] - Increased Version Number
s=	[RFC4566]
t=0 0	[RFC4566]
a=ice-ufrag:074c6550	[RFC5245]
a=ice-pwd:a28a397a4c3f31747d1ee3474af08a 068	[RFC5245]
a=fingerprint:sha-1 99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:7 0:9d:1f:66:79:a8:07	[RFC5245]
a=rtcp-rsize	[RFC5506]
m=audio 54609 RTP/SAVPF 0 109 98	[RFC4566]
c= IN IP4 24.23.204.141	[RFC4566]
a=rtpmap:0 PCMU/8000	[RFC3551]
a=rtpmap:109 opus/48000	[draft-spittka-payload -rtp-opus]
a=ptime:20	[draft-spittka-payload -rtp-opus]
a=rtpmap:98 iLBC/8000 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/SAVPF120	[RFC4566]
c= IN IP4 24.23.204.141	[RFC4566]
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp 8]
a=sendrecv	[RFC3264]
a=rtcp-mux	[RFC5761]
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	[RFC5245]	
54721 typ host		
a=candidate:1 2 UDP 1694302206	[RFC5245]	
24.23.204.141 54721 typ srflx raddr		
192.168.1.4 rport 54721		
a=rtcp-fb:120 nack pli	[RFC5104]	
a=rtcp-fb:120 ccm fir	[RFC5104]	
+-----+-----+-----+		

Table 11: 4.5 SDP Updated Offer w/Video

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:de4e99bd291c325921d5d47efbabd9	[RFC5245]	
a2		
a=fingerprint:sha-1	[RFC5245]	
99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:7		
0:9d:1f:66:79:a8:07		
a=rtcp-rsize	[RFC5506]	
m=audio 49203 RTP/SAVPF 109	[RFC4566]	
c= IN IP4 98.248.92.77	[RFC4566]	
a=rtpmap:109 opus/48000	[draft-spittka-payload	
	-rtp-opus]	
a=ptime:20	[draft-spittka-payload	
	-rtp-opus]	
a=sendrecv	[RFC3264]	
a=rtcp-mux	[RFC5761]	
a=candidate:0 1 UDP 2113667327	[RFC5245]	
192.168.1.7 49203 typ host		
a=ccandidate:1 1 UDP 1694302207	[RFC5245]	
98.248.92.77 49203 typ srflx raddr		
192.168.1.7 rport 49203		
a=candidate:candidate:0 2 UDP 2113667326	[RFC5245]	
192.168.1.7 60065 typ host		
a=candidate:1 2 UDP 1694302206	[RFC5245]	
98.248.92.77 60065 typ srflx raddr		
192.168.1.7 rport 60065		
m=video 63130 RTP/SAVPF 99 120	[RFC4566]	
c= IN IP4 98.248.92.771	[RFC4566]	
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp	
	8]	
a=sendrecv	[RFC3264]	

a=rtcp-mux	[RFC5761]	
a=candidate:0 1 UDP 2113667327	[RFC5245]	
192.168.1.7 63130 typ host		
a=candidate:1 1 UDP 1694302207	[RFC5245]	
98.248.92.77 63130 typ srflx raddr		
192.168.1.7 rport 63130		
a=candidate:0 2 UDP 2113667326	[RFC5245]	
192.168.1.7 56607 typ host		
a=candidate:1 2 UDP 1694302206	[RFC5245]	
98.248.92.77 56607 typ srflx raddr		
192.168.1.7 rport 56607		
+-----+-----+-----+		

Table 12: 4.5 SDP Updated Answer

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 and might be controversial in nature. The hope here is to demonstrate a plausible SDP composition to enhance seamless inter-operability between the aforementioned communication systems.

6.1. Secure Two-Way Audio, Video w/Feedback - WebRTC <-> Legacy Interop

In the scenario described below, Alice sends [[RFC3264](#)] Offer with two sets of media descriptions per media type.

One set that corresponds to [[WEBRTC](#)] Compliant RTP/SAVPF based audio and video descriptions.

Another set with RTP/AVP based audio and video descriptions for the legacy Interop purposes.

Also to note, Alice includes session level DTLS information and media level RTCP feedback information as applicable to both the sets of media descriptions

On the other hand, Bob being a Legacy VOIP end-point, recognizes only the media descriptions with RTP/AVP as the application protocol. The security and feedback requirements for the session are either handled by an intermediate gateway or with some combination of Bob's capabilities and the intermediate gateway.


```

title Successful 2-Way WebRTC <-> VOIP Interop
note right of Alice
Alice is on a WebRTC end-point & Bob is behind a legacy VOIP system
end note
Alice->Bob: Offer(Audio:Opus Video:VP8)
note right of Alice
  Alice includes 2 copies of media descriptions
  1. WebRTC compliant media description (RTP/SAVPF)
  2. Legacy compliant media description (RTP/AVP)
end note
Bob->Alice: Answer(Audio:Opus Video:VP8)
note right of Bob
  Bob recognizes"legacy compliant" media description from Alice.
  and accepts the same.
end note
Alice->Bob: Two-way Opus Audio, VP8 Video
note right of Alice
  Session also supports RTP/RTCP Mux, RTCP feedback (nack,pli)
end note

```

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:c7:7 0:9d:1f:66:79:a8:07	[RFC5245]
a=rtcp-rsize	[RFC5506]
m=audio 54609 RTP/SAVPF 109	[RFC4566]
c= IN IP4 24.23.204.141	[RFC4566]
a=rtpmap:109 opus/48000	[draft-spittka-payload
	-rtp-opus]
a=ptime:20	[draft-spittka-payload
	-rtp-opus]
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	[RFC5245]
192.168.1.4 64678 typ host	
a=candidate:1 2 UDP 1694302206	[RFC5245]
24.23.204.141 64678 typ srflx raddr	
192.168.1.4 rport 64678	
a=rtcp-fb:109 nack	[RFC5104]
m=video 62537 RTP/SAVPF 120	[RFC4566]
c= IN IP4 24.23.204.141	[RFC4566]
a=rtptime:120 VP8/90000	[draft-ietf-payload-vp]
	8]
a=sendrecv	[RFC3264]
a=rtcp-mux	[RFC5761]
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.4 62537 typ host	
a=candidate:1 1 UDP 1694302207	[RFC5245]
24.23.204.141 62537 typ srflx raddr	
192.168.1.4 rport 62537	
a=candidate:0 2 2113667326 192.168.1.4	[RFC5245]
54721 typ host	
a=candidate:1 2 UDP 1694302206	[RFC5245]
24.23.204.141 54721 typ srflx raddr	
192.168.1.4 rport 54721	
a=rtcp-fb:120 nack pli	[RFC5104]
a=rtcp-fb:120 ccm fir	[RFC5104]
-----	These set of media
	descriptions are for
	Legacy Inter-op
	purposes
m=audio 54732 RTP/AVP 109	[RFC4566]Alice
	includes RTP/AVP audio
	stream description
c= IN IP4 24.23.204.141	[RFC4566]
a=fingerprint:sha-1	[RFC5245]
99:41:49:83:4a:97:0e:1f:7f:7d:f9:c9:c7:7	
0:9d:1f:66:79:a8:07	
a=rtptime:109 opus/48000	[draft-spittka-payload]
	-rtp-opus]
a=ptime:20	[draft-spittka-payload]
	-rtp-opus]
a=sendrecv	[RFC3264]
a=rtcp-mux	[RFC5761]Alice still
	includes RTP/RTCP Mux
	support
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.4 54732 typ host	
a=candidate:1 1 UDP 694302207	[RFC5245]
24.23.204.141 54732 typ srflx raddr	
192.168.1.4 rport 54732	

a=candidate:0 2 UDP 2113667326	[RFC5245]
192.168.1.4 64678 typ host	
a=candidate:1 2 UDP 1694302206	[RFC5245]
24.23.204.141 64678 typ srflx raddr	
192.168.1.4 rport 64678	
a=rtcp-fb:109 nack	[RFC5104]She adds her
	intent for NACK RTCP
	feedback support
m=video 62445 RTP/AVP 120	[RFC4566]Alice
	includes RTP/AVP video
	stream description
c= IN IP4 24.23.204.141	[RFC4566]
a=fingerprint:sha-1	[RFC5245]
99:41:49:83:4a:97:0e:1f:ef:7d:f7:c9:c7:7	
0:9d:1f:66:79:a8:07	
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp
	8]
a=sendrecv	[RFC3264]
a=rtcp-mux	[RFC5761]Alice intends
	to perform RTP/RTCP
	Mux
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.4 62445 typ host	
a=candidate:1 1 UDP 1694302207	[RFC5245]
24.23.204.141 62537 typ srflx raddr	
192.168.1.4 rport 62445	
a=candidate:0 2 2113667326 192.168.1.4	[RFC5245]
54721 typ host	
a=candidate:1 2 UDP 1694302206	[RFC5245]
24.23.204.141 54721 typ srflx raddr	
192.168.1.4 rport 54721	
a=rtcp-fb:120 nack pli	[RFC5104] Alice
	indicates support for
	Picture loss
	Indication and NACK
	RTCP feedback
a=rtcp-fb:120 ccm fir	[RFC5104]

Table 13: 5.1 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:de4e99bd291c325921d5d47efbabd9	[RFC5245]
a2	
a=fingerprint:sha-1	[RFC5245]
99:41:49:83:4a:97:0e:1f:ef:6d:f7:c9:c7:7	
0:9d:1f:66:79:a8:07	
m=audio 49203 RTP/AVP 109	[RFC4566] Bob accepts
	RTP/AVP based audio
	stream
c= IN IP4 98.248.92.77	[RFC4566]
a=rtpmap:109 opus/48000	[draft-spittka-payload
	-rtp-opus]
a=ptime:20	[draft-spittka-payload
	-rtp-opus]
a=sendrecv	[RFC3264]
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.7 49203 typ host	
a=ccandidate:1 1 UDP 1694302207	[RFC5245]
98.248.92.77 49203 typ srflx raddr	
192.168.1.7 rport 49203	
a=candidate:candidate:0 2 UDP 2113667326	[RFC5245]
192.168.1.7 60065 typ host	
a=candidate:1 2 UDP 1694302206	[RFC5245]
98.248.92.77 60065 typ srflx raddr	
192.168.1.7 rport 60065	
m=video 63130 RTP/SAVP 120	[RFC4566] Bob accepts
	RTP/AVP based video
	stream
c= IN IP4 98.248.92.771	[RFC4566]
a=rtpmap:120 VP8/90000	[draft-ietf-payload-vp
	8]
a=sendrecv	[RFC3264]
a=candidate:0 1 UDP 2113667327	[RFC5245]
192.168.1.7 63130 typ host	
a=candidate:1 1 UDP 1694302207	[RFC5245]
98.248.92.77 63130 typ srflx raddr	
192.168.1.7 rport 63130	
a=candidate:0 2 UDP 2113667326	[RFC5245]
192.168.1.7 56607 typ host	

a=candidate:1 2 UDP 1694302206	[RFC5245]	
98.248.92.77 56607 typ srflx raddr		
192.168.1.7 rport 56607		
+-----+-----+		

Table 14: 5.1 SDP Answer

7. IANA Considerations

This document requires no actions from IANA.

8. References

8.1. Normative References

- [RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", [RFC 3264](#), June 2002.
- [RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", [RFC 4566](#), July 2006.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

8.2. 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.
- [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.
- [RFC5506] Johansson, I., "Support for Reduced-Size Real-Time Transport Control Protocol (RTCP): Opportunities and Consequences", [RFC 5506](#), April 2009.
- [RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", [RFC 3551](#),

July 2003.

- [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.
- [RFC4796] Hautakorpi, J. and G. Camarillo, "The Session Description Protocol (SDP) Content Attribute", [RFC 4796](#), February 2007.
- [RFC5761] Perkins, C. and M. Westerlund, "Multiplexing RTP Data and Control Packets on a Single Port", [RFC 5761](#), April 2010.
- [RFC3556] Casner, S., "Session Description Protocol (SDP) Bandwidth Modifiers for RTP Control Protocol (RTCP) Bandwidth", [RFC 3556](#), July 2003.
- [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.
- [RFC5888] Camarillo, G. and H. Schulzrinne, "RTP Payload Format for H.264 Video", [RFC 3984](#), June 2010.

[\[draft-spittka-payload-rtp-opus\]](#)

Spittka, J., Vos, K., and JM. Valin, "RTP Payload Format for Opus Speech and Audio Codec", [draft-spittka-payload-rtp-opus-01](#) (work in progress), July 2012.

[\[draft-ietf-payload-vp8\]](#)

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.

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

[\[draft-ietf-rtcweb-data-channel\]](#)

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

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

