

Network Working Group  
Request for Comments: 4317  
Category: Informational

A. Johnston  
Tello Corporation  
R. Sparks  
Estacado Systems  
December 2005

Session Description Protocol (SDP)  
Offer/Answer Examples

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2005).

Abstract

This document gives examples of Session Description Protocol (SDP) offer/answer exchanges. Examples include codec negotiation and selection, hold and resume, and addition and deletion of media streams. The examples show multiple media types, bidirectional, unidirectional, inactive streams, and dynamic payload types. Common Third Party Call Control (3pcc) examples are also given.

## Table of Contents

1. Overview .....	3
2. Codec Negotiation and Selection .....	3
2.1. Audio and Video 1 .....	3
2.2. Audio and Video 2 .....	4
2.3. Audio and Video 3 .....	5
2.4. Two Audio Streams .....	6
2.5. Audio and Video 4 .....	7
2.6. Audio Only 1 .....	8
2.7. Audio and Video 5 .....	9
2.8. Audio and Video 6 .....	10
3. Hold and Resume Scenarios .....	12
3.1. Hold and Unhold 1 .....	12
3.2. Hold with Two Streams .....	13
4. Addition and Deletion of Media Streams .....	15
4.1. Second Audio Stream Added .....	15
4.2. Audio, then Video Added .....	16
4.3. Audio and Video, Then Video Deleted .....	17
5. Third Party Call Control (3pcc) .....	19
5.1. No Media, Then Audio Added .....	19
5.2. Hold and Unhold 2 .....	20
5.3. Hold and Unhold 3 .....	21
6. Security Considerations .....	22
7. Informative References .....	22

## 1. Overview

This document describes offer/answer examples of Session Description Protocol (SDP) based on RFC 3264 [1]. The SDP in these examples is defined by RFC 2327 [2]. The offers and answers are assumed to be transported using a protocol such as Session Initiation Protocol (SIP) [3].

Examples include codec negotiation and selection, hold and resume, and addition and deletion of media streams. The examples show multiple media types, bidirectional, unidirectional, inactive streams, and dynamic payload types. Common Third Party Call Control (3pcc) [5] examples are also given.

The following sections contain examples in which two parties, Alice and Bob, exchange SDP offers, answers, and, in some cases, additional offers and answers. Note that the subject line (s=) contains a single space character.

## 2. Codec Negotiation and Selection

### 2.1. Audio and Video 1

This common scenario shows a video and audio session in which multiple codecs are offered but only one is accepted. As a result of the exchange shown below, Alice and Bob may send only PCMU audio and MPV video. Note: Dynamic payload type 97 is used for iLBC codec [6].

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0 8 97
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:97 iLBC/8000
m=video 51372 RTP/AVP 31 32
a=rtpmap:31 H261/90000
a=rtpmap:32 MPV/90000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49174 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 49170 RTP/AVP 32
a=rtpmap:32 MPV/90000
```

## 2.2. Audio and Video 2

Alice can support PCMU, PCMA, and iLBC codecs, but not more than one at the same time. Alice offers all three to maximize chances of a successful exchange, and Bob accepts two of them. An audio-only session is established in the initial exchange between Alice and Bob, using either PCMU or PCMA codecs (payload type in RTP packet tells which is being used). Since Alice only supports one audio codec at a time, a second offer is made with just that one codec, to limit the codec choice to just one.

Note: the version number is incremented in both SDP messages in the second exchange. After this exchange, only the PCMU codec may be used for media session between Alice and Bob.

Note: The declined video stream still present in the second exchange of SDP with ports set to zero.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0 8 97
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:97 iLBC/8000
m=video 51372 RTP/AVP 31 32
a=rtpmap:31 H261/90000
a=rtpmap:32 MPV/90000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 0 8
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
m=video 0 RTP/AVP 31
a=rtpmap:31 H261/90000
```

[Second-Offer]

```
v=0
o=alice 2890844526 2890844527 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 51372 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 0 RTP/AVP 31
a=rtpmap:31 H261/90000
```

[Second-Answer]

```
v=0
o=bob 2808844564 2808844565 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 0 RTP/AVP 31
a=rtpmap:31 H261/90000
```

### 2.3. Audio and Video 3

Alice offers three audio and two video codecs, while Bob accepts with a single audio and video codec. As a result of this exchange, Bob and Alice use iLBC for audio and H261 for video.

Note: change of dynamic payload type from 97 to 99 between the offer and the answer is OK since the same codec is referenced.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0 8 97
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:97 iLBC/8000
m=video 51372 RTP/AVP 31 32
a=rtpmap:31 H261/90000
a=rtpmap:32 MPV/90000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 99
a=rtpmap:99 iLBC/8000
m=video 51374 RTP/AVP 31
a=rtpmap:31 H261/90000
```

#### 2.4. Two Audio Streams

In this example, Alice wishes to establish separate audio streams, one for normal audio and the other for telephone-events. Alice offers two separate streams, one audio with two codecs and the other with RFC 2833 [4] tones (for DTMF). Bob accepts both audio streams choosing the iLBC codec and telephone-events.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0 97
a=rtpmap:0 PCMU/8000
a=rtpmap:97 iLBC/8000
m=audio 49172 RTP/AVP 98
a=rtpmap:98 telephone-event/8000
a=sendonly
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=audio 49174 RTP/AVP 98
a=rtpmap:98 telephone-event/8000
a=recvonly
```

## 2.5. Audio and Video 4

Alice and Bob establish an audio and video session with a single audio and video codec. In a second exchange, Bob changes his address for media and Alice accepts with the same SDP as the initial exchange (and as a result does not increment the version number).

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=video 51372 RTP/AVP 31
a=rtpmap:31 H261/90000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49174 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=video 49170 RTP/AVP 31
a=rtpmap:31 H261/90000
```

[Second-Offer]

```
v=0
o=bob 2808844564 2808844565 IN IP4 host.biloxi.example.com
s=
c=IN IP4 newhost.biloxi.example.com
t=0 0
m=audio 49178 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=video 49188 RTP/AVP 31
a=rtpmap:31 H261/90000
```

[Second-Answer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=video 51372 RTP/AVP 31
a=rtpmap:31 H261/90000
```

## 2.6. Audio Only 1

Alice wishes to establish an audio session with Bob using either PCMU codec or iLBC codec with RFC2833 tones, but not both at the same time. The offer contains these two media streams. Bob declines the first one and accepts the second one. If both media streams had been accepted, Alice would have sent a second declining one of the streams, as shown in Section 4.3.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=audio 51372 RTP/AVP 97 101
a=rtpmap:97 iLBC/8000
a=rtpmap:101 telephone-event/8000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 0 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=audio 49170 RTP/AVP 97 101
a=rtpmap:97 iLBC/8000
a=rtpmap:101 telephone-event/8000
```

## 2.7. Audio and Video 5

Alice and Bob establish an audio and video session in the first exchange with a single audio and video codec. In the second exchange, Alice adds a second video codec, which Bob accepts. This allows Alice and Bob to switch between the two video codecs without another offer/answer exchange.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 99
a=rtpmap:99 iLBC/8000
m=video 51372 RTP/AVP 31
a=rtpmap:31 H261/90000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 99
a=rtpmap:99 iLBC/8000
m=video 51374 RTP/AVP 31
a=rtpmap:31 H261/90000
```

[Second-Offer]

```
v=0
o=alice 2890844526 2890844527 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 99
a=rtpmap:99 iLBC/8000
m=video 51372 RTP/AVP 31 32
a=rtpmap:31 H261/90000
a=rtpmap:32 MPV/90000
```

[Second-Answer]

```
v=0
o=bob 2808844564 2808844565 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 99
a=rtpmap:99 iLBC/8000
m=video 51374 RTP/AVP 31 32
a=rtpmap:31 H261/90000
a=rtpmap:32 MPV/90000
```

## 2.8. Audio and Video 6

This example shows an audio and video offer that is accepted, but the answerer wants the video sent to a different address than that of the audio. This is a common scenario in conferencing where the video and audio mixing utilizes different servers. In this example, Alice offers audio and video, and Bob accepts.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0 8 97
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:97 iLBC/8000
m=video 51372 RTP/AVP 31 32
a=rtpmap:31 H261/90000
a=rtpmap:32 MPV/90000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49174 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 49172 RTP/AVP 32
c=IN IP4 otherhost.biloxi.example.com
a=rtpmap:32 MPV/90000
```

### 3. Hold and Resume Scenarios

#### 3.1. Hold and Unhold 1

Alice calls Bob, but when Bob answers he places Alice on hold. Bob then takes Alice off hold in the second offer. Alice changes port number in the second exchange. The media session between Alice and Bob is now active after Alice's second answer. Note that `a=sendrecv` could be present in both second offer and answer exchange. This is a common flow in 3pcc [5] scenarios.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0 97
a=rtpmap:0 PCMU/8000
a=rtpmap:97 iLBC/8000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 placeholder.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 97
a=rtpmap:97 iLBC/8000
a=sendonly
```

[Second-Offer]

```
v=0
o=bob 2808844564 2808844565 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

[Second-Answer]

```
v=0
o=alice 2890844526 2890844527 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49178 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

### 3.2. Hold with Two Streams

In this example, two audio streams have been established in the first offer/answer exchange. In this second offer/answer exchange, one of the audio streams is placed on hold. Alice offers two media streams, a bidirectional audio stream and a send-only telephone event stream. Bob accepts both streams. Bob then puts Alice's audio stream on hold but not the tone stream. Alice responds with identical SDP to the initial offer.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0 97
a=rtpmap:0 PCMU/8000
a=rtpmap:97 iLBC/8000
m=audio 49172 RTP/AVP 98
a=rtpmap:98 telephone-event/8000
a=sendonly
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=audio 49174 RTP/AVP 98
a=rtpmap:98 telephone-event/8000
a=recvonly
```

[Second-Offer]

```
v=0
o=bob 2808844564 2808844565 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 97
a=rtpmap:97 iLBC/8000
a=sendonly
m=audio 49174 RTP/AVP 98
a=rtpmap:98 telephone-event/8000
a=recvonly
```

[Second-Answer]

```
v=0
o=alice 2890844526 2890844527 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0 97
a=rtpmap:0 PCMU/8000
a=rtpmap:97 iLBC/8000
m=audio 49172 RTP/AVP 98
a=rtpmap:98 telephone-event/8000
a=sendonly
```

#### 4. Addition and Deletion of Media Streams

This section shows addition and deletion of media streams.

##### 4.1. Second Audio Stream Added

In this example, the first offer/answer exchange establishes a single audio stream with a single codec. The second offer/answer exchange adds a second audio stream for telephone events. The second stream is added by Bob's media server (different connection address) to receive RFC 2833 telephone-events (DTMF digits, typically) from Alice. Alice accepts. Even though the second stream is unidirectional, Alice receives RTCP packets on port 49173 from the media server.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0 97
a=rtpmap:0 PCMU/8000
a=rtpmap:97 iLBC/8000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

[Second-Offer]

```
v=0
o=bob 2808844564 2808844565 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=audio 48282 RTP/AVP 98
c=IN IP4 mediaserver.biloxi.example.com
a=rtpmap:98 telephone-event/8000
a=recvonly
```

[Second-Answer]

```
v=0
o=alice 2890844526 2890844527 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=audio 49172 RTP/AVP 98
c=IN IP4 host.atlanta.example.com
a=rtpmap:98 telephone-event/8000
a=sendonly
```

#### 4.2. Audio, then Video Added

An audio-only session is established in the initial exchange between Alice and Bob using PCMU codec. Alice adds a video stream that is accepted by Bob.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

[Second-Offer]

```
v=0
o=alice 2890844526 2890844527 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 49172 RTP/AVP 31
a=rtpmap:31 H261/90000
```

[Second-Answer]

```
v=0
o=bob 2808844564 2808844565 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 49168 RTP/AVP 31
a=rtpmap:31 H261/90000
```

#### 4.3. Audio and Video, Then Video Deleted

Alice and Bob establish an audio and video session. In a second exchange, Bob deletes the video session, resulting in an audio-only session.

## [Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=video 51372 RTP/AVP 31
a=rtpmap:31 H261/90000
```

## [Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49174 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=video 49170 RTP/AVP 31
a=rtpmap:31 H261/90000
```

## [Second-Offer]

```
v=0
o=bob 2808844564 2808844565 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49174 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=video 0 RTP/AVP 31
a=rtpmap:31 H261/90000
```

## [Second-Answer]

```
v=0
o=alice 2890844526 2890844527 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
m=video 0 RTP/AVP 31
a=rtpmap:31 H261/90000
```

## 5. Third Party Call Control (3pcc)

This section shows examples common in Third Party Call Control (3pcc) flows [5]. Call hold and resume flows are also common in 3pcc.

### 5.1. No Media, Then Audio Added

The first offer from Alice contains no media lines, so Bob accepts with no media lines. In the second exchange, Alice adds an audio stream that Bob accepts.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
```

[Second-Offer]

```
v=0
o=alice 2890844526 2890844527 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

[Second-Answer]

```
v=0
o=bob 2808844564 2808844565 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

## 5.2. Hold and Unhold 2

The first offer from Alice contains the connection address 0.0.0.0 and a random port number, which means that Bob can not send media to Alice (the media stream is "black holed" or "bh"). Bob accepts with normal SDP. In the second exchange, Alice changes the connection address, Bob accepts, and a media session is established.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 0.0.0.0
t=0 0
m=audio 23442 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

[Second-Offer]

```
v=0
o=alice 2890844526 2890844527 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

[Second-Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

### 5.3. Hold and Unhold 3

The first offer from Alice contains an audio stream, but the answer from Bob contains the connection address 0.0.0.0 and a random port number, which means that Alice can not send media to Bob (the media stream is "black holed" or "bh"). In the second exchange, Bob changes the connection address, Alice accepts, and a media session is established.

[Offer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

[Answer]

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 0.0.0.0
t=0 0
m=audio 9322 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

[Second-Offer]

```
v=0
o=bob 2808844564 2808844565 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49172 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

[Second-Answer]

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 97
a=rtpmap:97 iLBC/8000
```

## 6. Security Considerations

SDP offer and answer messages can contain private information about addresses and sessions to be established between parties. If this information needs to be kept private, some security mechanism in the protocol used to carry the offers and answers must be used. For SIP, this means using TLS transport and/or S/MIME encryption of the SDP message body.

It is important that SDP offer and answer messages be properly authenticated and authorized before they are used to establish a media session. Examples of SIP mechanisms include SIP Digest, certs, and cryptographically-verified SIP identity.

## 7. Informative References

- [1] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.
- [2] Handley, M. and V. Jacobson, "SDP: Session Description Protocol", RFC 2327, April 1998.
- [3] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.
- [4] Schulzrinne, H. and S. Petrack, "RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals", RFC 2833, May 2000.
- [5] Rosenberg, J., Peterson, J., Schulzrinne, H., and G. Camarillo, "Best Current Practices for Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP)", BCP 85, RFC 3725, April 2004.
- [6] Duric, A. and S. Andersen, "Real-time Transport Protocol (RTP) Payload Format for internet Low Bit Rate Codec (iLBC) Speech", RFC 3952, December 2004.

Authors' Addresses

Alan Johnston  
Tello Corporation  
999 Baker Way, Suite 250  
San Mateo, CA 94404

EMail: [ajohnston@tello.com](mailto:ajohnston@tello.com)

Robert J. Sparks  
Estacado Systems

EMail: [rjsparks@estacado.net](mailto:rjsparks@estacado.net)

## Full Copyright Statement

Copyright (C) The Internet Society (2005).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

## Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at [ietf-ipr@ietf.org](mailto:ietf-ipr@ietf.org).

## Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.

