RFC 8856
Session Description Protocol (SDP) Format for Binary Floor Control Protocol (BFCP) Streams

Abstract

This document defines the Session Description Protocol (SDP) offer/answer procedures for negotiating and establishing Binary Floor Control Protocol (BFCP) streams.

This document obsoletes RFC 4583.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8856.

Copyright Notice

Table of Contents

1. Introduction

As discussed in the BFCP (Binary Floor Control Protocol) specification [RFC8855], a given BFCP client needs a set of data in order to establish a BFCP connection to a floor control server. This data includes the transport address of the server, the conference identifier, and the user identifier.

One way for clients to obtain this information is to use a Session Description Protocol (SDP) offer/ answer exchange [RFC3264]. This document specifies how to encode this information in the SDP session descriptions that are part of such an offer/answer exchange.

User agents typically use the offer/answer model to establish a number of media streams of different types. Following this model, a BFCP connection is described as any other media stream by using an SDP "m=" line, possibly followed by a number of SDP lines that also apply to the BFCP connection.

Section 4 defines how the field values in an "m=" line representing a BFCP connection are set.

Section 5 defines SDP attributes that are used when negotiating a BFCP connection.

Section 6 defines multiplexing considerations for a BFCP connection.

Section 7 defines procedures for managing a BFCP connection.

Section 8 defines TLS and DTLS considerations when negotiating a BFCP connection.

Section 9 defines considerations regarding Interactive Connectivity Establishment (ICE) [RFC8445] when negotiating a BFCP connection.

Section 10 defines the SDP offer/answer procedures for negotiating a BFCP connection.

This document obsoletes RFC 4583 [RFC4583]. Section 14 summarizes the changes from RFC 4583.

2. Conventions

The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**NOT RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Floor Control Roles

When two endpoints establish a BFCP stream, they need to determine which of them acts as a floor control client and which acts as a floor control server.

Once the roles have been determined, the roles will apply to all BFCP-controlled streams associated with the BFCP stream.

4. Fields in the "m=" Line

According to the SDP specification [RFC4566], the "m=" line format is as follows:

```
m=<media> <port> <proto> <fmt> ...
```

This section describes how to generate an "m=" line of an SDP Media Description ("m=" section) describing a BFCP stream.

The media field **MUST** have a value of "application".

Depending on the value of the proto field, the port field is set as explained below. A port field value of zero has the standard SDP meaning (i.e., rejection of the media stream), regardless of the proto field.

- When TCP is used as the transport, the port field is set following the rules in [RFC4145]. Depending on the value of the 'setup' attribute (defined in [RFC4145] and discussed in Section 7.1), the port field contains the port to which the remote endpoint will direct BFCP messages, or in the case where the endpoint will initiate the connection towards the remote endpoint, should be set to a value of 9.
- When UDP is used as the transport, the port field contains the port to which the remote endpoint will direct BFCP messages, regardless of the value of the 'setup' attribute.

This document defines five values for the proto field: 'TCP/BFCP', 'TCP/DTLS/BFCP', 'TCP/TLS/BFCP', 'UDP/BFCP', and 'UDP/TLS/BFCP'.

The proto values are used as described below:

- 'TCP/BFCP' is used for TCP transport of BFCP without TLS encryption and is backward compatible with endpoints that are compliant with RFC 4583.
- 'TCP/TLS/BFCP' is used for TCP transport of BFCP with TLS encryption and is backward compatible with endpoints that are compliant with RFC 4583 and support TLS.
- 'UDP/BFCP' is used for UDP transport of BFCP without DTLS encryption [RFC6347].
- 'UDP/TLS/BFCP' is used for UDP transport of BFCP with DTLS encryption. This is one of the options when ICE is used (Section 9). It can also be used without ICE when backward compatibility with endpoints compliant with RFC 4583 is not required.
- 'TCP/DTLS/BFCP' is used for TCP transport of BFCP with DTLS encryption, running on top of TCP using the framing method defined in [RFC4571], with DTLS packets being sent and received instead of RTP / RTP Control Protocol (RTCP) packets, such that the LENGTH field defined in RFC 4571 precedes each DTLS message. This is one of the options when ICE is used (Section 9). It can also be used without ICE when backward compatibility with endpoints compliant with RFC 4583 is not required.

The fmt (format) list is not applicable to BFCP. The fmt list of "m=" lines in the case of any proto field value related to BFCP **MUST** contain a single "*" character. If the fmt list contains any other value, it **MUST** be ignored.

The following is an example of an "m=" line for a BFCP connection:

```
m=application 50000 TCP/TLS/BFCP *
```

5. SDP Attributes

5.1. SDP 'floorctrl' Attribute

This section defines the SDP 'floorctrl' media-level attribute. The attribute is used to determine the floor control roles (client and server) for the endpoints associated with the BFCP stream.

> Attribute Name: floorctrl
>
> Attribute Value: floor-control
>
> Usage Level: media
>
> Charset Dependent: No
>
> Mux Category: TBD

The Augmented BNF syntax [RFC5234] for the attribute is:

```
floor-control = role *(SP role)
role = "c-only" / "s-only" / "c-s"
```

An endpoint includes the attribute to indicate the role(s) it would be willing to perform for the BFCP-controlled media streams:

c-only: The endpoint is willing to act as a floor control client.

s-only: The endpoint is willing to act as a floor control server only.

When inserted in an offer, the offerer **MAY** indicate multiple attribute values ("c-only" and "s-only"). When inserted in an answer, the answerer **MUST** indicate only one attribute value: "c-only" or "s-only". The answerer indicates the role taken by the answerer. The offerer will then take the opposite role.

In [RFC4583], there was a third attribute specified, "c-s", which meant that an endpoint was willing to act as both a floor control client and a floor control server at the same time for the BFCP stream, taking different roles for different BFCP-controlled media streams. The feature was underspecified and implemented in different ways; in particular, many implementations interpreted "c-s" to mean that the endpoint is willing to act as either a client or a server (equivalent to "c-only s-only"). An implementation compliant with this specification **MUST NOT** include the "c-s" 'floorctrl' attribute value in an offer or in an answer but **MUST** accept the attribute value in an offer and process it as equivalent to "c-only s-only" (or "s-only c-only"). Also, as an implementation compliant with this specification is only allowed to include one role -- either "c-only" or "s-only" -- in an answer, each endpoint will only take one role, and as a result the endpoint will take the same role for each BFCP-controlled media stream associated with the BFCP stream.

Table 1 shows the roles that the answerer is allowed to take, based on what roles the offerer has indicated that it is willing to take.

Offerer	Answerer
c-only	s-only
s-only	c-only
c-s	c-only
c-s	s-only

Table 1: Roles

Endpoints compliant with [RFC4583] might not include the 'floorctrl' attribute in offers and answers. If the 'floorctrl' attribute is not present, in order to be interoperable with such endpoints, the offerer will act as a floor control client and the answerer will act as a floor control server.

The SDP Offer/Answer procedures for the 'floorctrl' attribute are defined in Section 10.

The following is an example of a 'floorctrl' attribute in an offer:

```
a=floorctrl:c-only s-only
```

5.2. SDP 'confid' Attribute

This section defines the SDP 'confid' media-level attribute. The attribute is used by a floor control server to convey the conference ID value to the floor control client, using decimal integer representation.

Attribute Name: confid

Attribute Value: conference-id

Usage Level: media

Charset Dependent: No

Mux Category: TBD

The Augmented BNF syntax [RFC5234] for the attribute is:

```
conference-id = 1*DIGIT

DIGIT = <DIGIT as defined in [RFC5234]>
```

The maximum value of the attribute is determined by the COMMON-HEADER format [RFC8855].

The SDP Offer/Answer procedures for the 'confid' attribute are defined in Section 10.

5.3. SDP 'userid' Attribute

This section defines the SDP 'userid' media-level attribute. The attribute is used by a floor control server to convey the user ID value to the floor control client, using decimal integer representation.

Attribute Name: userid

Attribute Value: user-id

Usage Level: media

Charset Dependent: No

Mux Category: TBD

The Augmented BNF syntax [RFC5234] for the attribute is:

```
user-id = 1*DIGIT

DIGIT = <DIGIT as defined in [RFC5234]>
```

The maximum value of the attribute is determined by the COMMON-HEADER format [RFC8855].

The SDP Offer/Answer procedures for the 'userid' attribute are defined in Section 10.

5.4. SDP 'floorid' Attribute

This section defines the SDP 'floorid' media-level attribute. The attribute is used to convey a floor identifier, using decimal integer representation, and, optionally, pointers to one or more BFCP-controlled media streams.

Attribute Name: floorid

Attribute Value: floor-id

Usage Level: media

Charset Dependent: No

Mux Category: TBD

The Augmented BNF syntax [RFC5234] for the attribute is:

```
floor-id = 1*DIGIT SP "mstrm:" token *(SP token)

DIGIT = <DIGIT as defined in [RFC5234]>
token = <token as defined in [RFC4566]>
```

The maximum value of the attribute is determined by the FLOOR-ID format [RFC8855].

The floor identifier value is the integer representation of the Floor ID field value [RFC8855] to be used in BFCP. Each media stream pointer value is associated with an SDP 'label' attribute [RFC4574] of a media stream.

The SDP Offer/Answer procedures for the 'floorid' attribute are defined in Section 10.

> Note: In the first SDP example in Section 9 of [RFC4583], 'm-stream' was erroneously listed. Although the example was non-normative, it is implemented by some vendors and occurs in cases where the endpoint is willing to act as a server. Therefore, it is **RECOMMENDED** to support parsing and interpreting 'm-stream' the same way as 'mstrm' when receiving.

5.5. SDP 'bfcpver' Attribute

This section defines the SDP 'bfcpver' media-level attribute. The attribute is used to negotiate the BFCP version, using decimal integer representation.

Attribute Name: bfcpver

Attribute Value: bfcp-version

Usage Level: media

Charset Dependent: No

Mux Category: TBD

The Augmented BNF syntax [RFC5234] for the attribute is:

```
bfcp-version = version *(SP version)
version = 1*DIGIT

DIGIT = <DIGIT as defined in [RFC5234]>
```

The maximum value of the attribute is determined by the COMMON-HEADER format [RFC8855].

An endpoint uses the 'bfcpver' attribute to convey the version(s) of BFCP supported by the endpoint, using integer values. For a given version, the attribute value representing the version **MUST** match the version ("Ver") field that would be presented in the BFCP COMMON-HEADER [RFC8855]. The BFCP version that will eventually be used will be conveyed with a BFCP-level Hello/HelloAck.

Endpoints compliant with [RFC4583] might not always include the 'bfcpver' attribute in offers and answers. The attribute value, if present, **MUST** be in accordance with the definition of the version ("Ver") field in [RFC8855]. If the attribute is not present, endpoints **MUST** assume a default value in accordance with [RFC8855]: when used over a reliable transport, the default attribute value is "1", and when used over an unreliable transport, the default attribute value is "2". The value is inferred from the transport specified in the "m=" line (Section 4) of the "m=" section associated with the stream.

The SDP Offer/Answer procedures for the 'bfcpver' attribute are defined in Section 10.

6. Multiplexing Considerations

[RFC8843] defines how multiplexing of multiple media streams can be negotiated. This specification does not define how BFCP streams can be multiplexed with other media streams. Therefore, a BFCP stream **MUST NOT** be associated with a BUNDLE group [RFC8843]. Note that BFCP-controlled media streams might be multiplexed with other media streams.

[RFC8859] defines the mux categories for the SDP attributes defined in this specification, except for the 'bfcpver' attribute. Table 2 defines the mux category for the 'bfcpver' attribute:

Name	Notes	Level	Mux Category
bfcpver	Needs further analysis in a separate specification	M	TBD

Table 2: Multiplexing Attribute Analysis

7. BFCP Connection Management

BFCP streams can use TCP or UDP as the underlying transport. Endpoints exchanging BFCP messages over UDP send the BFCP messages towards the peer using the connection address and port provided in the SDP "c=" and "m=" lines. TCP connection management is more complicated and is described in the following section.

> Note: When using Interactive Connectivity Establishment (ICE) [RFC8445], TCP/DTLS/ BFCP, or UDP/TLS/BFCP, the straightforward procedures for connection management via UDP/BFCP, as described above, apply. TCP/TLS/BFCP follows the same procedures as TCP/BFCP and is described below.

7.1. TCP Connection Management

The management of the TCP connection used to transport BFCP messages is performed using the SDP 'setup' and 'connection' attributes [RFC4145]. The 'setup' attribute indicates which of the endpoints initiates the TCP connection. The 'connection' attribute handles TCP connection re-establishment.

The BFCP specification [RFC8855] describes a number of situations when the TCP connection between a floor control client and the floor control server needs to be re-established. However, [RFC8855] does not describe the re-establishment process, because this process depends on how the connection was established in the first place. Endpoints using the offer/answer mechanism follow the following rules.

When the existing TCP connection is closed and re-established following the rules in [RFC8855], the floor control client **MUST** send an offer towards the floor control server in order to re-establish the connection. If a TCP connection cannot deliver a BFCP message and times out, the endpoint that attempted to send the message (i.e., the one that detected the TCP timeout) **MUST** send an offer in order to re-establish the TCP connection.

Endpoints that use the offer/answer mechanism to negotiate TCP connections **MUST** support the 'setup' and 'connection' attributes.

8. TLS/DTLS Considerations

When DTLS is used with UDP, the generic procedures defined in Section 5 of [RFC8842] **MUST** be followed.

When TLS is used with TCP, once the underlying connection is established, the answerer always acts as the TLS server. If the TCP connection is lost, the active endpoint [RFC4583] is responsible for re-establishing the TCP connection. Unless a new TLS connection is negotiated, subsequent SDP offers and answers will not impact the previously negotiated TLS roles.

> Note: For TLS, it was decided to keep the original procedures in [RFC4583] to determine which endpoint acts as the TLS server, in order to retain backward compatibility.

9. ICE Considerations

Generic SDP offer/answer procedures for ICE are defined in [RFC8839].

When BFCP is used with UDP-based ICE candidates [RFC8445], the procedures for UDP/TLS/BFCP are used.

When BFCP is used with TCP-based ICE candidates [RFC6544], the procedures for TCP/DTLS/BFCP are used.

Based on the procedures defined in [RFC8842], endpoints treat all ICE candidate pairs associated with a BFCP stream on top of a DTLS association as part of the same DTLS association. Thus, there will only be one BFCP handshake and one DTLS handshake even if there are multiple valid candidate pairs, and even if BFCP media is shifted between candidate pairs (including switching between UDP candidate pairs and TCP candidate pairs) prior to nomination. If new candidates are added, they will also be part of the same DTLS association.

In order to maximize the likelihood of interoperability between the endpoints, all ICE-enabled BFCP-over-DTLS endpoints **SHOULD** implement support for UDP/TLS/BFCP.

When an SDP offer or answer conveys multiple ICE candidates for a BFCP stream, UDP-based candidates **SHOULD** be included and the default candidate **SHOULD** be chosen from one of those UDP candidates. If UDP transport is used for the default candidate, then the "m=" line proto value **MUST** be 'UDP/TLS/BFCP'. If TCP transport is used for the default candidate, the "m=" line proto value **MUST** be 'TCP/DTLS/BFCP'.

> Note: Usage of ICE with protocols other than UDP/TLS/BFCP and TCP/DTLS/BFCP is out of scope for this specification.

10. SDP Offer/Answer Procedures

This section defines the SDP offer/answer [RFC3264] procedures for negotiating and establishing a BFCP stream. Generic procedures for DTLS are defined in [RFC8842]. Generic procedures for TLS are defined in [RFC8122].

This section only defines the BFCP-specific procedures. Unless explicitly stated otherwise, the procedures apply to an "m=" section describing a BFCP stream. If an offer or answer contains multiple "m=" sections describing BFCP streams, the procedures are applied independently to each stream.

Within this document, 'initial offer' refers to the first offer within an SDP session (e.g., a Session Initiation Protocol (SIP) dialog when SIP [RFC3261] is used to carry SDP) in which the offerer indicates that it wants to negotiate the establishment of a BFCP stream.

If the "m=" line 'proto' value is 'TCP/TLS/BFCP', 'TCP/DTLS/BFCP', or 'UDP/TLS/BFCP', the offerer and answerer follow the generic procedures defined in [RFC8122].

If the "m=" line proto value is 'TCP/BFCP', 'TCP/TLS/BFCP', 'TCP/DTLS/TCP', or 'UDP/TLS/BFCP', the offerer and answerer use the SDP 'setup' attribute according to the procedures in [RFC4145].

If the "m=" line proto value is 'TCP/BFCP', 'TCP/TLS/BFCP', or 'TCP/DTLS/BFCP', the offerer and answerer use the SDP 'connection' attribute according to the procedures in [RFC4145].

> Note: The use of source-specific SDP parameters [RFC5576] is not defined for BFCP streams.

10.1. Generating the Initial SDP Offer

When the offerer creates an initial offer, the offerer **MUST** include an SDP 'floorctrl' attribute (Section 5.1) and an SDP 'bfcpver' attribute (Section 5.5) in the "m=" section.

In addition, if the offerer includes an SDP 'floorctrl' attribute with "s-only" or "c-s" attribute values in the offer, the offerer

- **MUST** include an SDP 'confid' attribute (Section 5.2) in the "m=" section,
- **MUST** include an SDP 'userid' attribute (Section 5.3) in the "m=" section,
- **MUST** include an SDP 'floorid' attribute (Section 5.4) in the "m=" section, and
- **MUST** include an SDP 'label' attribute [RFC4574] with the "m=" section of each BFCP-controlled media stream.

> Note: If the offerer includes an SDP 'floorctrl' attribute with a "c-s" attribute value, or both a "c-only" and an "s-only" attribute value in the offer, the attribute values above will only be used if it is determined (Section 5.1) that the offerer will act as a floor control server.

10.2. Generating the SDP Answer

When the answerer receives an offer that contains an "m=" section describing a BFCP stream, the answerer **MUST** check whether it supports one or more of the BFCP versions supported by the offerer (Section 5.5). If the answerer does not support any of the BFCP versions, it **MUST NOT** accept the "m=" section. Otherwise, if the answerer accepts the "m=" section, the answerer

- **MUST** insert a corresponding "m=" section in the answer, with an identical "m=" line proto value [RFC4566],
- **MUST** include a 'bfcpver' attribute in the "m=" section; the versions indicated by the answer **MUST** be the same or a subset of the versions indicated by the offerer in the corresponding offer, and
- **MUST**, if the offer contained an SDP 'floorctrl' attribute, include a 'floorctrl' attribute in the "m=" section.

In addition, if the answerer includes an SDP 'floorctrl' attribute with an "s-only" attribute value in the answer, the answerer

- **MUST** include an SDP 'confid' attribute in the "m=" section,
- **MUST** include an SDP 'userid' attribute in the "m=" section,
- **MUST** include an SDP 'floorid' attribute in the "m=" section, and
- **MUST** include an SDP 'label' attribute in the "m=" section of each BFCP-controlled media stream.

> Note: An offerer compliant with [RFC4583] might not include 'floorctrl' and 'bfcpver' attributes in offers, in which case the default values apply.

Once the answerer has sent the answer, the answerer

- **MUST**, if the answerer is the active endpoint and if a TCP connection associated with the "m=" section is to be established (or re-established), initiate the establishment of the TCP connection, and
- **MUST**, if the answerer is the active endpoint and if a TLS/DTLS connection associated with the "m=" section is to be established (or re-established), initiate the establishment of the TLS/DTLS connection (by sending a ClientHello message).

If the answerer does not accept the "m=" section in the offer, it **MUST** assign a zero port value to the "m=" line of the corresponding "m=" section in the answer. In addition, the answerer **MUST NOT** establish a TCP connection or a TLS/DTLS connection associated with the "m=" section.

10.3. Offerer Processing of the SDP Answer

When the offerer receives an answer that contains an "m=" section describing a BFCP stream and with a non-zero port value in the "m=" line, the offerer

- **MUST**, if the offerer is the active endpoint and if a TCP connection associated with the "m=" section is to be established (or re-established), initiate the establishment of the TCP connection, and
- **MUST**, if the offerer is the active endpoint and if a TLS/DTLS connection associated with the "m=" section is to be established (or re-established), initiate the establishment of the TLS/ DTLS connection (by sending a ClientHello message).

> Note: An answerer compliant with [RFC4583] might not include 'floorctrl' and 'bfcpver' attributes in answers, in which case the default values apply.

If the "m=" line in the answer contains a zero port value or if the offerer for some other reason does not accept the answer (e.g., if the answerer only indicates support of BFCP versions not supported by the offerer), the offerer **MUST NOT** establish a TCP connection or a TLS/DTLS connection associated with the "m=" section.

10.4. Modifying the Session

When an offerer sends an updated offer, in order to modify a previously established BFCP stream, it follows the procedures in Section 10.1, with the following exceptions:

- If the BFCP stream is carried on top of TCP and if the offerer does not want to re-establish an existing TCP connection, the offerer **MUST** include in the "m=" section an SDP 'connection' attribute with a value of "existing", and
- If the offerer wants to disable a previously established BFCP stream, it **MUST** assign a zero port value to the "m=" line associated with the BFCP connection, following the procedures in [RFC3264].

11. Examples

For the purpose of brevity, the main portion of the session description is omitted in the examples, which only show "m=" sections and their "m=" lines and attributes.

The following is an example of an offer sent by a conference server to a client.

```
m=application 50000 TCP/TLS/BFCP *
a=setup:actpass
a=connection:new
a=fingerprint:sha-256 \
     19:E2:1C:3B:4B:9F:81:E6:B8:5C:F4:A5:A8:D8:73:04: \
     BB:05:2F:70:9F:04:A9:0E:05:E9:26:33:E8:70:88:A2
a=floorctrl:c-only s-only
a=confid:4321
a=userid:1234
a=floorid:1 mstrm:10
a=floorid:2 mstrm:11
a=bfcpver:1 2
m=audio 50002 RTP/AVP 0
a=label:10
m=video 50004 RTP/AVP 31
a=label:11
```

Note that due to RFC formatting conventions, this document splits the SDP entries across lines whose content would exceed the maximum line length. A backslash character ("\") marks where this line folding has taken place. This backslash and its trailing CRLF and whitespace would not appear in actual SDP content.

The following is the answer returned by the client.

```
m=application 9 TCP/TLS/BFCP *
a=setup:active
a=connection:new
a=fingerprint:sha-256 \
     6B:8B:F0:65:5F:78:E2:51:3B:AC:6F:F3:3F:46:1B:35: \
     DC:B8:5F:64:1A:24:C2:43:F0:A1:58:D0:A1:2C:19:08
a=floorctrl:c-only
a=bfcpver:1
m=audio 55000 RTP/AVP 0
m=video 55002 RTP/AVP 31
```

A similar example using an unreliable transport and DTLS is shown below, where the offer is sent from a client.

```
m=application 50000 UDP/TLS/BFCP *
a=setup:actpass
a=dtls-id:abc3dl
a=fingerprint:sha-256 \
     19:E2:1C:3B:4B:9F:81:E6:B8:5C:F4:A5:A8:D8:73:04: \
     BB:05:2F:70:9F:04:A9:0E:05:E9:26:33:E8:70:88:A2
a=floorctrl:c-only s-only
a=confid:4321
a=userid:1234
a=floorid:1 mstrm:10
a=floorid:2 mstrm:11
a=bfcpver:1 2
m=audio 50002 RTP/AVP 0
a=label:10
m=video 50004 RTP/AVP 31
a=label:11
```

The following is the answer returned by the server.

```
m=application 55000 UDP/TLS/BFCP *
a=setup:active
a=dtls-id:abc3dl
a=fingerprint:sha-256 \
     6B:8B:F0:65:5F:78:E2:51:3B:AC:6F:F3:3F:46:1B:35: \
     DC:B8:5F:64:1A:24:C2:43:F0:A1:58:D0:A1:2C:19:08
a=floorctrl:s-only
a=confid:4321
a=userid:1234
a=floorid:1 mstrm:10
a=floorid:2 mstrm:11
a=bfcpver:2
m=audio 55002 RTP/AVP 0
m=video 55004 RTP/AVP 31
```

12. Security Considerations

The BFCP specification [RFC8855], SDP specification [RFC4566], and offer/answer specification [RFC3264] discuss security issues related to BFCP, SDP, and offer/answer, respectively. In addition, [RFC4145] and [RFC8122] discuss security issues related to the establishment of TCP and TLS connections using an offer/answer model. Furthermore, when using DTLS over UDP, the generic offer/answer considerations defined in [RFC8842] MUST be followed.

The usage of certain proto values in the SDP offer/answer negotiation will result in a BFCP stream that is not protected by TLS or DTLS. Operators will need to provide integrity protection and confidentiality protection of the BFCP stream using other means.

The generic security considerations associated with SDP attributes are defined in [RFC3264]. While the attributes defined in this specification do not reveal information about the content of individual BFCP-controlled media streams, they do reveal which media streams will be BFCP controlled.

13. IANA Considerations

This document registers three new values in the "proto" subregistry within the "Session Description Protocol (SDP) Parameters" registry: 'TCP/DTLS/BFCP', 'UDP/BFCP', and 'UDP/TLS/BFCP' (see Section 13.1).

This document also registers a new SDP attribute in the "att-field" subregistry within the "Session Description Protocol (SDP) Parameters" registry: 'bfcpver' (see Section 5.5).

The remaining values are unchanged from [RFC4582], except that the references have been updated to refer to this document.

13.1. Registration of SDP 'proto' Values

The IANA has registered three new values in the SDP 'proto' field under the "Session Description Protocol (SDP) Parameters" registry.

Value	Reference
TCP/BFCP	RFC 8856
TCP/DTLS/BFCP	RFC 8856
TCP/TLS/BFCP	RFC 8856
UDP/BFCP	RFC 8856
UDP/TLS/BFCP	RFC 8856

Table 3: Values for the SDP 'proto' Field

13.2. Registration of the SDP 'floorctrl' Attribute

This document defines the SDP 'floorctrl' attribute. Details regarding this attribute are provided in Section 5.1.

13.3. Registration of the SDP 'confid' Attribute

This document defines the SDP 'confid' attribute. Details regarding this attribute are provided in Section 5.2.

13.4. Registration of the SDP 'userid' Attribute

This document defines the SDP 'userid' attribute. Details regarding this attribute are provided in Section 5.3.

13.5. Registration of the SDP 'floorid' Attribute

This document defines the SDP 'floorid' attribute. Details regarding this attribute are provided in Section 5.4.

13.6. Registration of the SDP 'bfcpver' Attribute

This document defines the SDP 'bfcpver' attribute. Details regarding this attribute are provided in Section 5.5.

14. Changes from RFC 4583

The technical changes and other fixes from [RFC4583] are listed below.

The main purpose of this work was to add signaling support necessary to support BFCP over an unreliable transport, as described in [RFC8855], resulting in the following changes:

- Fields in the "m=" Line (Section 4):

  This section has been rewritten to remove reference to the exclusivity of TCP as a transport for BFCP streams. The proto field values 'TCP/DTLS/BFCP', 'UDP/BFCP', and 'UDP/TLS/BFCP' have been added.

- Security Considerations (Section 12):

  For the DTLS-over-UDP case, we direct the reader to existing considerations and requirements for the offer/answer exchange as provided in [RFC8842].

- Registration of SDP 'proto' Values (Section 13.1):

  This document registers the three new values 'TCP/DTLS/BFCP', 'UDP/BFCP', and 'UDP/TLS/BFCP' in the "Session Description Protocol (SDP) Parameters" registry.

- SDP 'bfcpver' Attribute (Section 5.5):

  A new 'bfcpver' SDP media-level attribute has been added, in order to signal the supported version number.

In addition to the changes associated with support of BFCP over an unreliable transport, the possibility that an endpoint can act as both a floor control client and a floor control server at the same time has been removed. An endpoint will now take the same role for all BFCP-controlled streams associated with the BFCP stream.

Clarifications and bug fixes:

- Erratum ID 712 (Sections 3 and 10 of [RFC4583]; see [Err712] for details):

  Do not use language such as 'used in an "m=" line' when discussing an SDP attribute; instead, make clear that the attribute is a media-level attribute.

- Spelling corrected in the first SDP example in Section 9 of [RFC4583]:

  Do not use 'm-stream' as listed in the first SDP example in [RFC4583]; instead, use the correct 'mstrm' as specified in Section 11 of this document. However, we recommend continuing to interpret 'm-stream', if received, because it is still present in some implementations.

- Assorted clarifications (throughout the document):

  Language clarifications were made as a result of reviews. Also, normative language was "tightened" where appropriate, i.e., changed from "**SHOULD**" strength to "**MUST**" in a number of places.

15. References

15.1. Normative References

[RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3261]   Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <https://www.rfc-editor.org/info/rfc3261>.

[RFC3264]   Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, DOI 10.17487/RFC3264, June 2002, <https://www.rfc-editor.org/info/rfc3264>.

[RFC4145]   Yon, D. and G. Camarillo, "TCP-Based Media Transport in the Session Description Protocol (SDP)", RFC 4145, DOI 10.17487/RFC4145, September 2005, <https://www.rfc-editor.org/info/rfc4145>.

[RFC4566]   Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, DOI 10.17487/RFC4566, July 2006, <https://www.rfc-editor.org/info/rfc4566>.

[RFC4571]   Lazzaro, J., "Framing Real-time Transport Protocol (RTP) and RTP Control Protocol (RTCP) Packets over Connection-Oriented Transport", RFC 4571, DOI 10.17487/RFC4571, July 2006, <https://www.rfc-editor.org/info/rfc4571>.

[RFC4574] Levin, O. and G. Camarillo, "The Session Description Protocol (SDP) Label
           Attribute", RFC 4574, DOI 10.17487/RFC4574, August 2006, <https://www.rfc-
           editor.org/info/rfc4574>.

[RFC4582] Camarillo, G., Ott, J., and K. Drage, "The Binary Floor Control Protocol (BFCP)",
           RFC 4582, DOI 10.17487/RFC4582, November 2006, <https://www.rfc-editor.org/
           info/rfc4582>.

[RFC4583] Camarillo, G., "Session Description Protocol (SDP) Format for Binary Floor
           Control Protocol (BFCP) Streams", RFC 4583, DOI 10.17487/RFC4583, November
           2006, <https://www.rfc-editor.org/info/rfc4583>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications:
           ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <https://
           www.rfc-editor.org/info/rfc5234>.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2",
           RFC 6347, DOI 10.17487/RFC6347, January 2012, <https://www.rfc-editor.org/info/
           rfc6347>.

[RFC6544] Rosenberg, J., Keranen, A., Lowekamp, B. B., and A. B. Roach, "TCP Candidates
           with Interactive Connectivity Establishment (ICE)", RFC 6544, DOI 10.17487/
           RFC6544, March 2012, <https://www.rfc-editor.org/info/rfc6544>.

[RFC8122] Lennox, J. and C. Holmberg, "Connection-Oriented Media Transport over the
           Transport Layer Security (TLS) Protocol in the Session Description Protocol
           (SDP)", RFC 8122, DOI 10.17487/RFC8122, March 2017, <https://www.rfc-
           editor.org/info/rfc8122>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP
           14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/
           rfc8174>.

[RFC8445] Keranen, A., Holmberg, C., and J. Rosenberg, "Interactive Connectivity
           Establishment (ICE): A Protocol for Network Address Translator (NAT)
           Traversal", RFC 8445, DOI 10.17487/RFC8445, July 2018, <https://www.rfc-
           editor.org/info/rfc8445>.

[RFC8839] Petit-Huguenin, M., Nandakumar, S., Holmberg, C., Keränen, A., and R. Shpount,
           "Session Description Protocol (SDP) Offer/Answer Procedures for Interactive
           Connectivity Establishment (ICE)", RFC 8839, DOI 10.17487/RFC8839, May 2020,
           <https://www.rfc-editor.org/info/rfc8839>.

[RFC8842] Holmberg, C. and R. Shpount, "Session Description Protocol (SDP) Offer/Answer
           Considerations for Datagram Transport Layer Security (DTLS) and Transport
           Layer Security (TLS)", RFC 8842, DOI 10.17487/RFC8842, May 2020, <https://
           www.rfc-editor.org/info/rfc8842>.

[RFC8855]   Camarillo, G., Drage, K., Kristensen, T., Ott, J., and C. Eckel, "The Binary Floor Control Protocol (BFCP)", RFC 8855, DOI 10.17487/RFC8855, May 2020, <https://www.rfc-editor.org/info/rfc8855>.

[RFC8859]   Nandakumar, S., "A Framework for SDP Attributes When Multiplexing", DOI 10.17487/RFC8859, RFC 8859, May 2020, <https://www.rfc-editor.org/info/rfc8859>.

15.2. Informative References

[Err712]   RFC Errata, Erratum ID 712, RFC 4583, <https://www.rfc-editor.org/errata/eid712>.

[RFC5576]   Lennox, J., Ott, J., and T. Schierl, "Source-Specific Media Attributes in the Session Description Protocol (SDP)", RFC 5576, DOI 10.17487/RFC5576, June 2009, <https://www.rfc-editor.org/info/rfc5576>.

[RFC8843]   Holmberg, C., Alvestrand, H., and C. Jennings, "Negotiating Media Multiplexing Using the Session Description Protocol (SDP)", RFC 8843, DOI 10.17487/RFC8843, May 2020, <https://www.rfc-editor.org/info/rfc8843>.

Acknowledgements

Authors' Addresses

Gonzalo Camarillo
Ericsson
Hirsalantie 11
FI-02420 Jorvas
Finland
Email: Gonzalo.Camarillo@ericsson.com

Tom Kristensen
Jotron AS
Ringdalskogen 8
3270 Larvik
Norway
Email: tom.kristensen@jotron.com, tomkri@ifi.uio.no

Christer Holmberg
Ericsson
Hirsalantie 11
FI-02420 Jorvas
Finland
Email: christer.holmberg@ericsson.com