INTERNET-DRAFT
Document: draft-ietf-mmusic-sdp-comedia-01.txt
Expires April 2002

D. Yon
Dialout.Net

October 2001

Connection-Oriented Media Transport in SDP
<draft-ietf-mmusic-sdp-comedia-01.txt>

Status of this Memo

Abstract

This document describes how to express media transport over connection-oriented protocols using the Session Description Protocol (SDP). It defines two new protocol identifiers: TCP and TLS. It also defines the syntax and semantics for an SDP "direction" attribute that describes the connection setup procedure.

Introduction

The Session Description Protocol [SDP] provides a general-purpose format for describing multimedia sessions in announcements or invitations. SDP uses an entirely textual data format (the US-ASCII subset of [UTF-8]) to maximize portability among transports. SDP does not define a protocol, but only the syntax to describe a multimedia session with sufficient information to discover and participate in that session. Session descriptions may be sent using any number of existing application protocols for transport (e.g., SAP, SIP, RTSP, email, HTTP, etc.).

Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119 [7] and indicate requirement levels for compliant implementations.

Motivation

[SDP] describes two protocol identifiers: RTP/AVP and UDP, both of which are unreliable, connectionless protocols, an appropriate choice for multimedia streams. There are, however, applications for which the connection-oriented transports such as TCP are more appropriate, but [SDP] provides no way to describe a session that uses protocols other than RTP or UDP.

Connection-oriented protocols introduce a new factor when describing a session: not only must it be possible to express that a protocol will be based on this protocol, but it must also describe the connection setup procedure.

1 Protocol Identifiers

1.1 TCP

The TCP protocol identifier is similar to the UDP protocol identifier in that it only describes the transport protocol without any connotation as to the upper-layer protocol. An m= line that specifies "TCP" MUST further qualify the protocol using a fmt identifier (see [SDP] Appendix B).

1.2 TLS

The TLS protocol identifier specifies that the session will use the Transport Layer Security protocol [TLS] with an implied transport protocol of TCP. To describe a media session that uses TLS over TCP, the protocol identifier "TLS" must be specified in the m= line. An m= line that specifies TLS MUST further qualify the protocol using a fmt identifier.

2 Direction Attribute

An important attribute of connection-oriented protocols is the setup procedure. One endpoint needs to initiate the connection and the other endpoint needs to accept the connection. The direction attribute is used to describe these roles, and the syntax is as follows:

```
a=direction:<role> [<source-address>]
```

The <role> is one of the following:

passive: The endpoint will accept an incoming connection.

active: The endpoint will initiate an outgoing connection.

both: The endpoint will both accept an incoming connection and will initiate an outgoing connection.

reuse: The endpoint will use the connection that has already been established with the opposite endpoint.

The <source-address> is a sequence of values that describe the address and port number from where the connection will originate, and consists of the following values:

```
nettype addrtype unicast-address [port]
```

The <source-address> is an optional value that may be specified with direction:active, direction:both, or direction:reuse. Within the <source-address>, the source port number is RECOMMENDED but may be omitted.

2.1 Semantics of direction:passive

By specifying direction:passive, the endpoint indicates that the port number specified in the m= line is available to accept a connection from the other endpoint. The endpoint MUST NOT specify a <source-address> after direction:passive.

2.2 Semantics of direction:active

By specifying direction:active, the endpoint indicates that it will initiate a connection to the port number on the m= line of the other endpoint. The port number on its own m= line is irrelevant, and the opposite endpoint MUST NOT attempt to initiate a connection to the port number specified there. Nevertheless, since the m= line must contain a valid port number, the endpoint specifying direction:active SHOULD specify a port number of 9 (the discard port) on its m= line. The endpoint MUST NOT specify a port number of zero, as that carries other semantics in [SDP].

The endpoint SHOULD specify the address and port number from which it will initiate the connection in the <source-address> position on the a= line.

2.3 Semantics of direction:both

By specifying direction:both, the endpoint indicates that it will both accept a TCP connection on the port number of its own m= line, and that it will also initiate a connection to the port number on the m= line of the other endpoint.

As with direction:active, the endpoint SHOULD specify the address and port number from which it will initiate the connection in the <source-address> position on the a= line.

Since this attribute describes behavior that is similar to connectionless media descriptions in [SDP], it is the default value for the direction attribute and is therefore optional.

Endpoints may choose to specify direction:both for one or more of the following reasons:

1) The endpoint has no preference as to whether it accepts or initiates the connection, and therefore is offering the remote endpoint a choice of connection setup procedures.

2) The endpoints intend to use a single connection to transport the media, but it is not known whether firewall issues will prevent either endpoint from initiating or accepting the connection. Therefore both endpoints will attempt to initiate a connection in hopes that at least one will succeed.

3) The endpoints intend to use two connections to transport the media, and one must be initiated by the remote endpoint and the other must be initiated by the local endpoint.

If one endpoint specifies either direction:active or direction:passive and the other specifies direction:both, both endpoints MUST behave as if the latter had specified the inverse direction of the former. For example, specifying direction:both when the other endpoint specifies direction:active SHALL cause both endpoints to behave as if the former had specified direction:passive. Conversely, specifying direction:both when the other endpoint specifies direction:passive SHALL cause both endpoints to behave as if the former had specified direction:active.

If both endpoints specify direction:both then each endpoint MUST initiate a connection to the port number specified on the m= line of the opposite endpoint. If a single connection is needed (case #1 or #2 above), there is one exception to this requirement: if an endpoint receives the incoming connection from the opposite endpoint prior to initiating its own outbound connection, then that endpoint MAY use that connection rather than attempt to make an outbound connection to the opposite endpoint.

If only one connection succeeds, then that connection will be used to carry the media. Once it has transmitted data on this connection, the initiating endpoint MUST NOT perform another connection attempt to the accepting endpoint. This allows the accepting endpoint to release or recycle the listening port for another session once it has received data from the initiating endpoint.

If both connections succeed but only one was needed (case #2 above), the following rules SHALL apply:

a) Each endpoint MUST accept data from either connection.

b) Once an endpoint has transmitted data to one of the connections, it MUST use that connection exclusively for transmission.

c) Once an endpoint has transmitted AND received data, if one of the connections is determined to be idle, the endpoint MAY close the idle connection.

2.4 Semantics of direction:reuse

By specifying direction:reuse, the endpoint indicates that it is changing the parameter(s) of an existing session on a previously established connection with the opposite endpoint. Therefore no new connections are to be created. This is intended for cases where media types are added, removed, or changed during a session. For example, an endpoint adding a video stream to an existing audio session may elect to multiplex the new stream over the same connection that is currently transporting the audio stream.

2.5 Bidirectional versus Unidirectional Media

In traditional SDP transport types the flow is unidirectional. If the intent is for media to flow in both directions, both endpoints must specify SDP that describes where to deliver the media and what media type(s) to use. For example, if only Endpoint A presents SDP then media can only flow towards Endpoint A, as Endpoint B has not specified where and how to send media to it.

Because most connection-oriented media is inherently bi-directional, endpoints may encounter a situation where only one side presented SDP yet there is now a network path that can carry media in either direction. In keeping with traditional SDP semantics, an endpoint MUST NOT send data to the other endpoint unless it has specified SDP information describing the type of media it can accept.

It is, however, perfectly acceptable for an endpoint to transmit data on the same connection it is using to receive data, so long as the other endpoint has advertised its willingness to accept data. Likewise, it is perfectly acceptable for an endpoint to receive data on the same connection it is using to transmit data to the corresponding remote endpoint. In other words, for a bi-directional application-level session, a connection may be used to send data in both directions (contingent to rules outlined in Section 2.3) as long as one side of the connection is attached to either of the advertised SDP transport addresses.

3 Source-Address Considerations

In the cases where the endpoint is initiating the connection, it is RECOMMENDED that a source address be specified on the a= line by that endpoint. It is also RECOMMENDED that the source port be included in the source address. In most environments, the source port number can be determined by binding the socket before initiating the connect, as shown in the sample C code below:

```
{
 SOCKET s_id
```

```
    SOCKADDR_IN cli_sin;
    int namelen;

       // Create the socket
       s_id = socket(AF_INET,SOCK_STREAM,IPPROTO_TCP);

       // Bind the socket to any IP address and port
       bzero((char *)&cli_sin,sizeof(cli_sin));
       cli_sin.sin_family      = AF_INET;
       cli_sin.sin_addr.s_addr = htonl(INADDR_ANY);
       cli_sin.sin_port        = 0;
       bind(s_id,(SOCKADDR *)&cli_sin,sizeof(cli_sin));

       // Find the port number that was bound
       namelen = sizeof(cli_sin);
       getsockname(s_id,(SOCKADDR *)&cli_sin,&namelen);

       // Print the port number
       printf("Source Port = %d\n",ntohs(cli_sin.sin_port));
   }
```

If the source address is omitted, the receiver of the SDP packet MUST NOT make any assumptions in regards to the address or port from where the connection will originate. In particular, the receiver MUST NOT assume that the address information listed on the c= line has any implication as to where the media connection originates.

NOTE:

The motivation for specifying the source address is twofold. First, it aids Application-Level Proxies by explicitly announcing the source of the outbound connection. This allows, for example, a dynamic firewall pinhole to be created that will allow the connection to pass.

Second, it allows the passive endpoint to correlate the incoming connection with the session being negotiated. Note that great care must be taken when using the source address as a means to identify incoming connections, as Network Address Translation (NAT) can render the source address unreliable. In addition if the originating endpoint omits the source port, the source address can be ambiguous if multiple, logical endpoints share the same network address. Therefore it is NOT RECOMMENDED that the source address be used for this purpose unless the SDP occurs in the context of a controlled network topology that guarantees that the source address is both correct (i.e., no NAT, or a NAT with an Application-Level Proxy that rewrites the SDP) and unambiguous (i.e., the source port is specified).

3.1 Source Address Timing Considerations

When used in conjunction with a session signaling protocol such as SIP, there may be cases where an endpoint initiates a connection prior to the opposite endpoint receiving the SDP that describe the source address of the initiating endpoint. Therefore, an endpoint that has advertised an address and port number with direction:both or direction:passive MUST be ready to accept a connection on that address and port immediately. If the accepting endpoint requires the source address to identify the initiating endpoint, it MUST keep the connection active and allow sufficient time for the source address to arrive before discarding the connection.

4 Examples

What follows are a number of examples that show the most common usage of the direction attribute combined with TCP-based media descriptions. For the purpose of brevity, the main portion of the session description is omitted in the examples and is assumed to be the following:

```
v=0
o=me 2890844526 2890842807 IN IP4 10.1.1.2
e=Me <me@ietf.org>
s=Call me using TCP
t=0 0
```

4.1 Example: simple passive/active

An endpoint at 10.1.1.2 signals the availability of a T.38 fax session at port 54111:

```
c=IN IP4 10.1.1.2
m=image 54111 TCP t38
a=direction:passive
```

An endpoint at 10.1.1.1 receiving this description responds with the following:

```
c=IN IP4 10.1.1.1
```

```
     m=image 9 TCP t38
     a=direction:active
```

The endpoint at 10.1.1.1 then initiates the TCP connection to port 54111 at 10.1.1.2. Note that the TCP connection may originate from any address or port. The endpoint at 10.1.1.1 could have optionally committed to a source address with a simple modification:

```
     c=IN IP4 10.1.1.1
     m=image 9 TCP t38
     a=direction:active IN IP4 10.1.1.1 1892
```

By adding the source address to the a= line, the endpoint at 10.1.1.1 must now use a source port of 1892 when initiating the TCP connection to port 54111 at 10.1.1.2.

4.2 Example: agnostic both

An endpoint at 10.1.1.2 signals the availability of a T.38 fax session at TCP port 54111, but is also willing to set up the media stream by initiating the TCP connection:

```
     c=IN IP4 10.1.1.2
     m=image 54111 TCP t38
     a=direction:both
```

The endpoint at 10.1.1.1 has three choices:

1) It can respond with either of the two direction:active descriptions listed in the previous example. In this case the endpoint at 10.1.1.1 must initiate a connection to port 54111 at 10.1.1.2.

2) It can respond with a description similar to the following:

```
     c=IN IP4 10.1.1.1
     m=image 54321 TCP t38
     a=direction:passive
```

   In this case the endpoint at 10.1.1.2 must initiate a connection to port 54321 at 10.1.1.1.

3) It can respond with a description that specifies direction:both, which is covered in the next example.

4.3 Example: redundant both

An endpoint at 10.1.1.2 uses the same description as the previous example:

```
        c=IN IP4 10.1.1.2
        m=image 54111 TCP t38
        a=direction:both
```

Unlike the previous example, the endpoint at 10.1.1.1 responds with the following description:

```
        c=IN IP4 10.1.1.1
        m=image 54321 TCP t38
        a=direction:both
```

This will cause the endpoint at 10.1.1.2 to initiate a connection to port 54321 at 10.1.1.1, and the endpoint at 10.1.1.1 to initiate a connection to port 54111 at 10.1.1.2. Whichever TCP connection succeeds will be used. If both succeed, one of the connections may be closed as an optimization, using the rules in section 2.3.

5 Security Considerations

See [SDP] for security and other considerations specific to the Session Description Protocol in general. There are no new security considerations introduced by these protocol identifiers and attributes.

6 IANA Considerations

As recommended by [SDP] Appendix B, the direction attribute described in this document should be registered with IANA, as should the "TCP" and "TLS" protocol identifiers.

Acknowledgements

The author would like to thank Jonathan Rosenberg, Anders Kristensen, Paul Kyzivat, and Robert Fairlie-Cuninghame for their valuable insights.

Appendix A: Direction Attribute Syntax

This appendix provides an Augmented BNF [ABNF] grammar for expressing the direction attribute for connection setup. It is intended as an extension to the grammar for the Session Description Protocol, as defined in [SDP]. Specifically, it describes the syntax for the new "connection-setup" attribute field, which MAY be either a session-level or media-level attribute.

```
connection-setup =    "direction" ":" direction-spec

direction-spec =      "passive" | qualified-direction

qualified-direction = direction-ident | direction-ident source

direction-ident =     "both" | "active" | "reuse"

source =              nettype addrtype unicast-address |
                      nettype addrtype unicast-address port
```

References

[ABNF] D. Crocker, P. Overell, "Augmented BNF for Syntax Specifications: ABNF," RFC 2234, November 1997

[SDP] M. Handley, V. Jacobson, "SDP: Session Description Protocol," RFC 2327, April 1998

[T38] International Telecommunication Union, "Procedures for Real-Time Group 3 Facsimile Communications over IP Networks," Recommendation T.38, June 1998

[TLS] T. Dierks, C. Allen, "The TLS Protocol," RFC 2246, January 1999

[UTF-8] F. Yergeau, "UTF-8, a transformation format of Unicode and ISO 10646," RFC 2044, October 1996

AuthorÆs Address

David Yon
Dialout.Net, Inc.
One Indian Head Plaza

Nashua, NH 03060

Phone: (603) 324-4100
EMail: yon@dialout.net

Full Copyright Statement