

The Session Description Protocol (SDP) Content Attribute

Status of This Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

This document defines a new Session Description Protocol (SDP) media-level attribute, 'content'. The 'content' attribute defines the content of the media stream to a more detailed level than the media description line. The sender of an SDP session description can attach the 'content' attribute to one or more media streams. The receiving application can then treat each media stream differently (e.g., show it on a big or small screen) based on its content.

Table of Contents

1.	Introduction	2
2.	Terminology	2
3.	Related Techniques	2
4.	Motivation for the New Content Attribute	3
5.	The Content Attribute	4
6.	The Content Attribute in the Offer/Answer Model	5
7.	Examples	6
8.	Operation with SMIL	7
9.	Security Considerations	7
10.	IANA Considerations	8
11.	Acknowledgements	8
12.	References	9
12.1.	Normative References	9
12.2.	Informational References	9

1. Introduction

The Session Description Protocol (SDP) [1] is a protocol that is intended to describe multimedia sessions for the purposes of session announcement, session invitation, and other forms of multimedia session initiation. One of the most typical use cases of SDP is where it is used with the Session Initiation Protocol (SIP) [5].

There are situations where one application receives several similar media streams, which are described in an SDP session description. The media streams can be similar in the sense that their content cannot be distinguished just by examining their media description lines (e.g., two video streams). The 'content' attribute is needed so that the receiving application can treat each media stream appropriately based on its content.

This specification defines the SDP 'content' media-level attribute, which provides more information about the media stream than the 'm' line in an SDP session description.

The main purpose of this specification is to allow applications to take automated actions based on the 'content' attributes. However, this specification does not define those actions. Consequently, two implementations can behave completely differently when receiving the same 'content' attribute.

2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in BCP 14, RFC 2119 [3], and indicate requirement levels for compliant implementations.

3. Related Techniques

The 'label' attribute [10] enables a sender to attach a pointer to a particular media stream. The namespace of the 'label' attribute itself is unrestricted; so, in principle, it could also be used to convey information about the content of a media stream. However, in practice, this is not possible because of the need for backward compatibility. Existing implementations of the 'label' attribute already use values from that unrestricted namespace in an application-specific way. So, it is not possible to reserve portions of the 'label' attribute's namespace without possible conflict with already used application-specific labels.

It is possible to assign semantics to a media stream with an external document that uses the 'label' attribute as a pointer. The downside of this approach is that it requires an external document. Therefore, this kind of mechanism is only applicable to special use cases where such external documents are used (e.g., centralized conferencing).

Yet another way to attach semantics to a media stream is to use the 'i' SDP attribute, defined in [1]. However, values of the 'i' attribute are intended for human users and not for automata.

4. Motivation for the New Content Attribute

Currently, SDP does not provide any means for describing the content of a media stream (e.g., speaker's image, slides, sign language) in a form that the application can understand. Of course, the end user can see the content of the media stream and read its title, but the application cannot understand what the media stream contains.

The application that is receiving multiple similar (e.g., same type and format) media streams needs, in some cases, to know what the contents of those streams are. This kind of situation occurs, for example, in cases where presentation slides, the speaker's image, and sign language are transported as separate media streams. It would be desirable that the receiving application could distinguish them in a way that it could handle them automatically in an appropriate manner.

```

+-----+
|+-----++-----+|
||          ||          ||
|| speaker's ||          ||
||  image    ||          ||
||          ||          ||
||          || presentation ||
||          ||    slides   ||
||          ||          ||
||  sign     ||          ||
|| language  ||          ||
||          ||          ||
|+-----++-----+|
+-----+

```

Figure 1: Application's Screen

Figure 1 shows a screen of a typical communication application. The 'content' attribute makes it possible for the application to decide where to show each media stream. From an end user's perspective, it

is desirable that the user does not need to arrange each media stream every time a new media session starts.

The 'content' attribute could also be used in more complex situations. An example of such a situation is an application controlling equipment in an auditorium. An auditorium can have many different output channels for video (e.g., main screen and two smaller screens) and audio (e.g., main speakers and headsets for the participants). In this kind of environment, a lot of interaction from the end user who operates the application would be required in absence of cues from a controlling application. The 'content' attribute would make it possible, for example, for an end user to specify, only once, which output each media stream of a given session should use. The application could automatically apply the same media layout for subsequent sessions. So, the 'content' attribute can help reduce the amount of required end-user interaction considerably.

5. The Content Attribute

This specification defines a new media-level value attribute, 'content'. Its formatting in SDP is described by the following ABNF (Augmented Backus-Naur Form) [2]:

```
content-attribute = "a=content:" mediact-tag
mediact-tag      = mediact *(", " mediact)
mediact          = "slides" / "speaker" / "sl" / "main"
                  / "alt" / mediact-ext
mediact-ext      = token
```

The 'content' attribute contains one or more tokens, which MAY be attached to a media stream by a sending application. An application MAY attach a 'content' attribute to any media stream it describes.

This document provides a set of pre-defined values for the 'content' attribute. Other values can be defined in the future. The pre-defined values are:

slides: the media stream includes presentation slides. The media type can be, for example, a video stream or a number of instant messages with pictures. Typical use cases for this are online seminars and courses. This is similar to the 'presentation' role in H.239 [12].

speaker: the media stream contains the image of the speaker. The media can be, for example, a video stream or a still image. Typical use cases for this are online seminars and courses.

sl: the media stream contains sign language. A typical use case for this is an audio stream that is translated into sign language, which is sent over a video stream.

main: the media stream is taken from the main source. A typical use case for this is a concert where the camera is shooting the performer.

alt: the media stream is taken from the alternative source. A typical use case for this is an event where the ambient sound is separated from the main sound. The alternative audio stream could be, for example, the sound of a jungle. Another example is the video of a conference room, while the main stream carries the video of the speaker. This is similar to the 'live' role in H.239.

All these values can be used with any media type. We chose not to restrict each value to a particular set of media types in order not to prevent applications from using innovative combinations of a given value with different media types.

The application can make decisions on how to handle a single media stream based on both the media type and the value of the 'content' attribute. If the application does not implement any special logic for the handling of a given media type and 'content' value combination, it applies the application's default handling for the media type.

Note that the same 'content' attribute value can occur more than once in a single session description.

6. The Content Attribute in the Offer/Answer Model

This specification does not define a means to discover whether the peer endpoint understands the 'content' attribute because 'content' values are just informative at the offer/answer model [8] level. The fact that the peer endpoint does not understand the 'content' attribute does not keep the media session from being established. The only consequence is that end user interaction on the receiving side may be required to direct the individual media streams appropriately.

The 'content' attribute describes the data that the application generating the SDP session description intends to send over a particular media stream. The 'content' values for both directions of a media stream do not need to be the same. Therefore, an SDP answer MAY contain 'content' attributes even if none were present in the offer. Similarly, the answer MAY contain no 'content' attributes even if they were present in the offer. Furthermore, the values of 'content' attributes do not need to match in an offer and an answer.

The 'content' attribute can also be used in scenarios where SDP is used in a declarative style. For example, 'content' attributes can be used in SDP session descriptors that are distributed with Session Announcement Protocol (SAP) [9].

7. Examples

There are two examples in this section. The first example, shown below, uses a single 'content' attribute value per media stream:

```
v=0
o=Alice 292742730 29277831 IN IP4 131.163.72.4
s=Second lecture from information technology
c=IN IP4 131.164.74.2
t=0 0
m=video 52886 RTP/AVP 31
a=rtpmap:31 H261/9000
a=content:slides
m=video 53334 RTP/AVP 31
a=rtpmap:31 H261/9000
a=content:speaker
m=video 54132 RTP/AVP 31
a=rtpmap:31 H261/9000
a=content:sl
```

The second example, below, is a case where there is more than one 'content' attribute value per media stream. The difference with the previous example is that now the conferencing system might automatically mix the video streams from the presenter and slides:


```
v=0
o=Alice 292742730 29277831 IN IP4 131.163.72.4
s=Second lecture from information technology
c=IN IP4 131.164.74.2
t=0 0
m=video 52886 RTP/AVP 31
a=rtpmap:31 H261/9000
a=content:slides,speaker
m=video 54132 RTP/AVP 31
a=rtpmap:31 H261/9000
a=content:sl
```

8. Operation with SMIL

The values of 'content' attribute, defined in [Section 5](#), can also be used with Synchronized Multimedia Integration Language (SMIL) [[11](#)]. SMIL contains a 'param' element, which is used for describing the content of a media flow. However, this 'param' element, like the 'content' attribute, provides an application-specific description of the media content.

Details on how to use the values of the 'content' attribute with SMIL's 'param' element are outside the scope of this specification.

9. Security Considerations

An attacker may attempt to add, modify, or remove 'content' attributes from a session description. Depending on how an implementation chooses to react to the presence or absence of a given 'content' attribute, this could result in an application behaving in an undesirable way; therefore, it is strongly RECOMMENDED that integrity protection be applied to the SDP session descriptions.

Integrity protection can be provided for a session description carried in an SIP [[5](#)], e.g., by using S/MIME [[6](#)] or Transport Layer Security (TLS) [[7](#)].

It is assumed that values of 'content' attribute do not contain data that would be truly harmful if it is exposed to a possible attacker. It must be noted that the initial set of values does not contain any data that would require confidentiality protection. However, S/MIME and TLS can be used to protect confidentiality, if needed.

10. IANA Considerations

This document defines a new 'content' attribute for SDP. It also defines an initial set of values for it. Some general information regarding the 'content' attribute is presented in the following:

Contact name: Jani Hautakorpi <Jani.Hautakorpi@ericsson.com>.

Attribute name: 'content'.

Type of attribute Media level.

Subject to charset: No.

Purpose of attribute: The 'content' attribute gives information from the content of the media stream to the receiving application.

Allowed attribute values: "slides", "speaker", "sl", "main", "alt", and any other registered values.

The IANA created a subregistry for 'content' attribute values under the Session Description Protocol (SDP) Parameters registry. The initial values for the subregistry are as follows:

Value of 'content' attribute	Reference	Description
slides	RFC 4796	Presentation slides
speaker	RFC 4796	Image from the speaker
sl	RFC 4796	Sign language
main	RFC 4796	Main media stream
alt	RFC 4796	Alternative media stream

As per the terminology in [RFC 2434](#) [4], the registration policy for new values for the 'content' parameter shall be 'Specification Required'.

If new values for 'content' attributes are specified in the future, they should consist of a meta description of the contents of a media stream. New values for 'content' attributes should not describe things like what to do in order to handle a stream.

11. Acknowledgements

The authors would like to thank Arnoud van Wijk and Roni Even, who provided valuable ideas for this document. We wish to also thank Tom Taylor for his thorough review.

12. References

12.1. Normative References

- [1] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", [RFC 4566](#), July 2006.
- [2] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", [RFC 4234](#), October 2005.
- [3] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [4] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", [BCP 26](#), [RFC 2434](#), October 1998.

12.2. Informational References

- [5] 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.
- [6] Ramsdell, B., "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.1 Message Specification", [RFC 3851](#), July 2004.
- [7] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1", [RFC 4346](#), April 2006.
- [8] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", [RFC 3264](#), June 2002.
- [9] Handley, M., Perkins, C., and E. Whelan, "Session Announcement Protocol", [RFC 2974](#), October 2000.
- [10] Levin, O. and G. Camarillo, "The Session Description Protocol (SDP) Label Attribute", [RFC 4574](#), August 2006.
- [11] Michel, T. and J. Ayars, "Synchronized Multimedia Integration Language (SMIL 2.0) - [Second Edition]", World Wide Web Consortium Recommendation REC-SMIL2-20050107, January 2005, <<http://www.w3.org/TR/2005/REC-SMIL2-20050107>>.
- [12] ITU-T, "Infrastructure of audiovisual services - Systems aspects; Role management and additional media channels for H.300-series terminals", Series H H.239, July 2003.

Authors' Addresses

Jani Hautakorpi
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

E-Mail: Jani.Hautakorpi@ericsson.com

Gonzalo Camarillo
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

E-Mail: Gonzalo.Camarillo@ericsson.com

Full Copyright Statement

Copyright (C) The IETF Trust (2007).

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, THE IETF TRUST 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.

Acknowledgement

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

