Network Working Group
Request for Comments: 3959
Category: Standards Track
December 2004

G. Camarillo
Ericsson

The Early Session Disposition Type for the Session Initiation Protocol (SIP)

Status of This Memo

Copyright Notice

Abstract

This document defines a new disposition type (early-session) for the Content-Disposition header field in the Session Initiation Protocol (SIP). The treatment of "early-session" bodies is similar to the treatment of "session" bodies. That is, they follow the offer/answer model. Their only difference is that session descriptions whose disposition type is "early-session" are used to establish early media sessions within early dialogs, as opposed to regular sessions within regular dialogs.

Table of Contents

1. Introduction

Early media refers to media (e.g., audio and video) that is exchanged before a particular session is accepted by the called user. Within a dialog, early media occurs from the moment the initial INVITE is sent until the User Agent Server (UAS) generates a final response. It may be unidirectional or bidirectional, and can be generated by the caller, the callee, or both. Typical examples of early media generated by the callee are ringing tone and announcements (e.g., queuing status). Early media generated by the caller typically consists of voice commands or dual tone multi-frequency (DTMF) tones to drive interactive voice response (IVR) systems.

The basic SIP specification (RFC 3261 [2]) only supports very simple early media mechanisms. These simple mechanisms have a number of problems related to forking and security, and do not satisfy the requirements of most applications. RFC 3960 [8] goes beyond the mechanisms defined in RFC 3261 [2] and describes two models of early media using SIP: the gateway model and the application server model.

Although both early media models described in RFC 3960 [8] are superior to the one specified in RFC 3261 [2], the gateway model still presents a set of issues. In particular, the gateway model does not work well with forking. Nevertheless, the gateway model is needed because some SIP entities (in particular, some gateways) cannot implement the application server model.

The application server model addresses some of the issues present in the gateway model. This model uses the early-session disposition type specified in this document.

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 [1] and indicate requirement levels for compliant implementations.

3. Issues Related to Early Media Session Establishment

Traditionally, early media sessions have been established in the same way as regular sessions. That is, using an offer/answer exchange where the disposition type of the session descriptions is "session". Application servers perform an offer/answer exchange with the User Agent Client (UAC) to exchange early media exclusively, while UASs use the same offer/answer exchange, first to exchange early media, and once the regular dialog is established, to exchange regular media. This way of establishing early media sessions is known as the gateway model [8], which presents some issues related to forking and security. These issues exist when this model is used by either an application server or by a UAS.

Application servers may not be able to generate an answer for an offer received in the INVITE. The UAC created the offer for the UAS, and so, it may have applied end-to-end encryption or have included information (e.g., related to key management) that the application server is not supposed to use. Therefore, application servers need a means to perform an offer/answer exchange with the UAC that is independent from the offer/answer exchange between both UAs.

UASs using the offer/answer exchange that will carry regular media for sending and receiving early media can cause media clipping, as described in Section 2.1.1 of [8]. Some UACs cannot receive early media from different UASs at the same time. So, when an INVITE forks and several UASs start sending early media, the UAC mutes all the UASs but one (which is usually chosen at random). If the UAS that accepts the INVITE (i.e., sends a 200 OK) was muted, a new offer/answer exchange is needed to unmute it. This usually causes media clipping. Therefore, UASs need a means of performing an offer/answer exchange with the UAC to exchange early media that is independent from the offer/answer exchanged used to exchange regular media.

A potential solution to this need would be to establish a different dialog using a globally routable URI to perform an independent offer/answer exchange. This dialog would be labelled as a dialog for early media and would be somehow related to the original dialog at the UAC. However, performing all the offer/answer exchanges within the original dialog has many advantages:

- It is simpler.
- It does not have synchronization problems, because all the early dialogs are terminated when the session is accepted.
- It does not require globally routable URIs.
- It does not introduce service interaction issues related to services that may be wrongly applied to the new dialog.
- It makes firewall management easier.

This way of performing offer/answer exchanges for early media is referred to as the application server model [8]. This model uses the early-session disposition type defined in the following section.

4. The Early Session Disposition Type

We define a new disposition type for the Content-Disposition header field: early-session. User agents MUST use early-session bodies to establish early media sessions in the same way as they use session bodies to establish regular sessions, as described in RFCs 3261 [2] and 3264 [3]. Particularly, early-session bodies MUST follow the offer/answer model and MAY appear in the same messages as session bodies do with the exceptions of 2xx responses for an INVITE and ACKs. Nevertheless, it is NOT RECOMMENDED that early offers in INVITEs be included because they can fork, and the UAC could receive multiple early answers establishing early media streams at roughly the same time. Also, the use of the same transport address (IP address plus port) in a session body and in an early-session body is NOT RECOMMENDED. Using different transport addresses (e.g., different ports) to receive early and regular media makes it easy to detect the start of the regular media.

If a User Agent (UA) needs to refuse an early-session offer, it MUST do so by refusing all the media streams in it. When SDP [7] is used, this is done by setting the port number of all the media streams to zero.

> This is the same mechanism that UACs use to refuse regular offers that arrive in a response to an empty INVITE.

An early media session established using early-session bodies MUST be terminated when its corresponding early dialog is terminated or it transitions to a regular dialog.

It is RECOMMENDED that UAs generating regular and early session descriptions use, as long as it is possible, the same codecs in both. This way, the remote UA does not need to change codecs when the early session transitions to a regular session.

5. Preconditions

RFC 3312 [4] defines a framework for preconditions for SDP. Early-sessions MAY contain preconditions, which are treated in the same way as preconditions in regular sessions. That is, the UAs do not exchange media, and the called user is not alerted until the preconditions are met.

6. Option Tag

We define an option tag to be used in Require and Supported header fields: early-session. A UA adding the early-session option tag to a message indicates that it understands the early-session disposition type.

7. Example

Figure 1 shows the message flow between two UAs. INVITE (1) has an early-session option tag in its Supported header field and the body shown in Figure 2. The UAS sends back a response with two body parts, as shown in Figure 3: one of disposition type session and the other early-session. The session body part is the answer to the offer in the INVITE. The early-session body part is an offer to establish an early media session. When the UAC receives the 183 (Session Progress) response, it sends the answer to the early-session offer in a PRACK, as shown in Figure 4. This early media session is terminated when the early dialog transitions to a regular dialog. That is, when the UAS sends the (5) 200 (OK) response for the INVITE.

```
      A                            B
      |                            |
      |--------(1) INVITE-------->|
      |            offer           |
      |                            |
      |<--(2) Session Progress----|
      |       early-offer          |
      |       answer               |
      |                            |
      |---------(3) PRACK-------->|
      |             early-answer  |
      |                            |
      |<--------(4) 200 OK--------|
      |                            |
      |  *                     *  |
      | ************************* |
      |*       Early Media       *|
      | ************************* |
      |  *                     *  |
      |                            |
      |<--------(5) 200 OK--------|
      |                            |
      |----------(6) ACK--------->|
      |                            |
```

Figure 1: Message flow

```
Content-Type: application/sdp
Content-Disposition: session

v=0
o=alice 2890844730 2890844731 IN IP4 host.example.com
s=
c=IN IP4 192.0.2.1
t=0 0
m=audio 20000 RTP/AVP 0
```

Figure 2: Offer

```
Content-Type: multipart/mixed; boundary="boundary1"
Content-Length: 401

--boundary1
Content-Type: application/sdp
Content-Disposition: session

v=0
o=Bob 2890844725 2890844725 IN IP4 host.example.org
s=
c=IN IP4 192.0.2.2
t=0 0
m=audio 30000 RTP/AVP 0

--boundary1
Content-Type: application/sdp
Content-Disposition: early-session

v=0
o=Bob 2890844714 2890844714 IN IP4 host.example.org
s=
c=IN IP4 192.0.2.2
t=0 0
m=audio 30002 RTP/AVP 0

--boundary1--
```

Figure 3: Early offer and answer

```
Content-Type: application/sdp
Content-Disposition: early-session

v=0
o=alice 2890844717 2890844717 IN IP4 host.example.com
s=
c=IN IP4 192.0.2.1
t=0 0
m=audio 20002 RTP/AVP 0
```

Figure 4: Early answer

8. Security Considerations

The security implications of using early-session bodies in SIP are the same as when using session bodies; they are part of the offer/answer model.

SIP uses the offer/answer model [3] to establish early sessions in both the gateway and the application server models. User Agents (UAs) generate a session description, which contains the transport address (i.e., IP address plus port) where they want to receive media, and send it to their peer in a SIP message. When media packets arrive at this transport address, the UA assumes that they come from the receiver of the SIP message carrying the session description. Nevertheless, attackers may attempt to gain access to the contents of the SIP message and send packets to the transport address contained in the session description. To prevent this situation, UAs SHOULD encrypt their session descriptions (e.g., using S/MIME).

Still, even if a UA encrypts its session descriptions, an attacker may try to guess the transport address used by the UA and send media packets to that address. Guessing such a transport address is sometimes easier than it may seem because many UAs always pick up the same initial media port. To prevent this situation, UAs SHOULD use media-level authentication mechanisms (e.g., Secure Realtime Transport Protocol (SRTP)[6]). In addition, UAs that wish to keep their communications confidential SHOULD use media-level encryption mechanisms (e.g, SRTP [6]).

Attackers may attempt to make a UA send media to a victim as part of a DoS attack. This can be done by sending a session description with the victim’s transport address to the UA. To prevent this attack, the UA SHOULD engage in a handshake with the owner of the transport address received in a session description (just verifying willingness to receive media) before sending a large amount of data to the transport address. This check can be performed by using a connection oriented transport protocol, by using Simple Traversal of the UDP Protocol through NAT (STUN)[5] in an end-to-end fashion, or by the key exchange in SRTP [6].

In any event, note that the previous security considerations are not early media specific, but apply to the usage of the offer/answer model in SIP to establish sessions in general.

Additionally, an early media-specific risk (roughly speaking, an equivalent to forms of "toll fraud" in the Public Switched Telephone Network (PSTN)) attempts to exploit the different charging policies some operators apply to early and to regular media. When UAs are allowed to exchange early media for free, but are required to pay for regular media sessions, rogue UAs may try to establish a bidirectional early media session and never send a 2xx response for the INVITE.

On the other hand, some application servers (e.g., Interactive Voice Response systems) use bidirectional early media to obtain information from the callers (e.g., the Personal Identification Number (PIN) code of a calling card). So, we do not recommend that operators disallow bidirectional early media. Instead, operators should consider a remedy of charging early media exchanges that last too long, or stopping them at the media level (according to the operator’s policy).

9. IANA Considerations

This document defines a new Content-Disposition header field disposition type (early-session) in Section 4. This value has been registered in the IANA registry for Content-Dispositions with the following description:

early-session	The body describes an early communications session, for example, an RFC 2327 SDP body

This document defines a SIP option tag (early-session) in Section 6. It has been registered in the SIP parameters registry (http://www.iana.org/assignments/sip-parameters) under "Option Tags", with the following description.

early-session	A UA adding the early-session option tag to a message indicates that it understands the early-session content disposition.

10. Acknowledgements

Francois Audet, Christer Holmberg, and Allison Mankin provided useful comments on this document.

11. References

11.1. Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[2] 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.

[3] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

[4] Camarillo, G., Marshall, W., and J. Rosenberg, "Integration of Resource Management and Session Initiation Protocol (SIP)", RFC 3312, October 2002.

[5] Rosenberg, J., Weinberger, J., Huitema, C., and R. Mahy, "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)", RFC 3489, March 2003.

[6] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

11.2. Informational References

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

[8] Camarillo, G. and H. Schulzrinne, "Early Media and Ringing Tone Generation in the Session Initiation Protocol (SIP)", RFC 3960, December 2004.

Author's Address

Gonzalo Camarillo
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

EMail: Gonzalo.Camarillo@ericsson.com

Full Copyright Statement

Intellectual Property

Acknowledgement

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