Internet Engineering Task Force (IETF)
Request for Comments: 7403
Category: Standards Track
ISSN: 2070-1721

H. Kaplan
Oracle

November 2014

A Media-Based Traceroute Function for the Session Initiation Protocol (SIP)

Abstract

SIP already provides the ability to perform hop-by-hop traceroute for SIP messages using the Max-Forwards header field to determine the reachability path of requests to a target. A mechanism for media-loopback calls has also been defined separately, which enables test calls to be generated that result in media being looped back to the originator. This document describes a means of performing hop-by-hop traceroute-style test calls using the media-loopback mechanism to test the media path when SIP sessions go through media-relaying back-to-back user agents (B2BUAs).

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 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7403.

Table of Contents

1. Introduction

In many deployments, the media for SIP-created sessions does not flow directly from the originating User Agent Client (UAC) to the answering User Agent Server (UAS). Often, SIP B2BUAs in the SIP signaling path also insert themselves in the media plane path by manipulating Session Description Protocol (SDP), either for injecting media such as rich ringtones or music-on-hold or for relaying media in order to provide functions such as transcoding, IPv4-IPv6 conversion, NAT traversal, Secure Realtime Transport Protocol (SRTP) termination, media steering, etc.

As more SIP domains get deployed and interconnected, the odds of a SIP session crossing such media-plane B2BUAs increases as well as the number of such B2BUAs any given SIP session may go through. In other words, any given SIP session may cross any number of B2BUAs both in the SIP signaling plane as well as the media plane.

When a failure or degradation occurs in the media plane, it is difficult to determine where in the media path it occurred. In order to aid managing and troubleshooting SIP-based sessions and media

traversing such B2BUAs, it would be useful to progressively test the media path as it reaches successive B2BUAs with a test controlled solely by the source User Agent (UA). A mechanism to perform media-loopback test sessions has been defined in [RFC6849], but it cannot be used directly to test B2BUAs because, typically, the B2BUAs do not have an Address of Record (AOR) to be targeted, nor is it known a priori which B2BUAs will be traversed for any given session.

For example, suppose calls from Alice to Bob have media problems. Alice would like to test the media path to each B2BUA in the path to Bob separately, to determine which segment has the issues. Alice cannot target the B2BUAs directly for each test call; she doesn't know which URIs to use to target them, nor would using such URIs guarantee the same media path be used as a call to Bob. A better solution would be to make a test call targeted to Bob, but with a SIP traceroute-type mechanism that makes the call terminate at the B2BUAs, such that she can perform test sessions to test the media path to each downstream B2BUA.

This document defines how such a mechanism can be employed, using the mechanism in [RFC6849] along with the Max-Forwards SIP header field such that a SIP UA can make multiple test calls, each reaching a B2BUA further downstream. Each B2BUA in the path that supports the mechanism in [RFC6849] would answer the media-loopback call; thus, the originating SIP UA can test the media path up to that B2BUA.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

B2BUA: a SIP Back-to-Back User Agent, which is the logical combination of a User Agent Server (UAS) and User Agent Client (UAC).

UAS: a SIP User Agent Server

UAC: a SIP User Agent Client

Traceroute: a mechanism to trace a path of hops from an originator to a destination. For IP, this is typically done using the Time To Live (TTL) field of the IP header, starting at the value 1 and incrementing by 1 as each IP hop responds with an ICMP error. For SIP, this can be done using Max-Forwards header field starting with the value 0, in a similar fashion to the TTL field.

It is assumed the reader is already familiar with media-loopback [RFC6849].

3. The SIP Traceroute Mechanism

The Max-Forwards header field can already be used to generate a simple SIP-request traceroute by generating a SIP request initially using a Max-Forwards value of 0, receiving a 483 Too Many Hops response from the next-hop, and then incrementing the value for subsequent SIP requests; one would thereby reach SIP devices further and further downstream, receiving 483 from each of them.

The mechanism described in this document uses such a traceroute of a Max-Forwards style to perform media-loopback testing. To perform a SIP media-plane traceroute, the originating UAC (Alice) generates a SIP INVITE to a target AOR (Bob), with a Max-Forwards header field value of 0 and with SDP based on [RFC6849]. The SIP next-hop will either reject the request with a 483 Too Many Hops response or, if the next-hop is a B2BUA that supports this mechanism and if the B2BUA allows such testing from the requesting UAC, the B2BUA will answer the INVITE to establish the dialog and create a media-loopback session.

The originating UAC can then end the media-loopback session, generate another INVITE to the same target AOR with a Max-Forwards header field value of 1, which will reach the second SIP next-hop, and so on.

A SIP Reason header field cause value of '483' (as defined in [RFC3326]) will be in the 200 answer from each B2BUA answering the INVITE, until the INVITE reaches the final UAS (Bob), which does not use the Reason cause value (see Section 3.2 for details).

Using this mechanism, a SIP UAC can test the path from itself to each successive B2BUA on the path to a target. Such a mechanism could also be useful for establishing a permanent test call between an Enterprise and a Service Provider across a SIP Trunk, for example, or for automated measurement systems to test the media path between domains, etc.

3.1. Processing a Received Max-Forwards Header Field

As currently defined in [RFC3261], the UAS half of a B2BUA does not technically need to inspect the Max-Forwards header field value for received requests: only Proxies do. This behavior was updated by [RFC7332], such that a compliant B2BUA needs to both inspect the value in order to prevent loops, as well as copy and decrement the value as if it were a Proxy. This document also requires such

behavior in order for the mechanism to succeed; therefore, a B2BUA supporting the traceroute mechanism defined in this document MUST also comply with [RFC7332].

3.2. Answering the INVITE

If a SIP B2BUA receives a dialog-creating INVITE request with a Max-Forwards header value of 0, with SDP for media-loopback based on [RFC6849], and the policies of the B2BUA allow it to answer such a request, then it is answered as if the original target of the request were the local SIP B2BUA. The normal procedures of SIP apply, as well as [RFC6849], as if the request had been targeted at the local B2BUA device as the intended destination all along.

In the 200 response for the INVITE, the B2BUA MUST also add a Reason header, per [RFC3326], with a protocol field value of "SIP", a cause field value of "483", and a reason-text value of "Traceroute Response". The purpose of the Reason header is to indicate to the UAC that the request is being answered due to reaching a Max-Forwards of 0, rather than being answered due to reaching the final UAS. When the ultimate target UAS answers a loopback-based INVITE with a Max-Forwards greater than or equal to 0, the Reason header would not be added to the response and the UAC will know the traceroute is complete.

If a B2BUA receives an INVITE with media-loopback SDP and a Max-Forwards header field value of 0, as defined in this document, and it does not accept the session (e.g., due to local policy), then it SHOULD respond with a 483 Too Many Hops response, per the normal rules of [RFC3261], as it would previously. In other words, in such a case, it behaves no differently than it would have if it did not support this document's new behavior.

4. Security Considerations

There are security implications for the mechanism defined in this document. Answering media-loopback calls in a B2BUA consumes resources on the B2BUA, and network bandwidth in between and, thus, exposes a vector for denial-of-service (DoS) attacks; therefore, B2BUAs should provide configuration options to control who can make such test calls, how many concurrent calls can be established and maintained, and how long calls can continue. Entities that deploy B2BUAs should set these options to values that reduce the DoS risk to an acceptable level. For example, a B2BUA might perform digest-challenge authentication with specific credentials for such calls or it might only allow specific sources to make such calls, at a

specific time. Such policies are typically vendor specific based on local policies and deployment usage scenarios and cannot be explicitly defined in this document.

The security considerations of [RFC6849] also apply to this document. Since B2BUAs are not end-user devices, there is no human user to monitor the loopback session activity on the B2BUA as recommended in [RFC6849]; instead, B2BUAs should log such events or provide some form of administrative notification.

5. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://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, June 2002, <http://www.rfc-editor.org/info/rfc3261>.

[RFC3326] Schulzrinne, H., Oran, D., and G. Camarillo, "The Reason Header Field for the Session Initiation Protocol (SIP)", RFC 3326, December 2002, <http://www.rfc-editor.org/info/rfc3326>.

[RFC6849] Kaplan, H., Ed., Hedayat, K., Venna, N., Jones, P., and N. Stratton, "An Extension to the Session Description Protocol (SDP) and Real-time Transport Protocol (RTP) for Media Loopback", RFC 6849, February 2013, <http://www.rfc-editor.org/info/rfc6849>.

[RFC7332] Kaplan, H. and V. Pascual, "Loop Detection Mechanisms for Session Initiation Protocol (SIP) Back-to-Back User Agents (B2BUAs)", RFC 7332, August 2014, <http://www.rfc-editor.org/info/rfc7332>.

Acknowledgments

The general concept of performing media-loopback on a hop-by-hop basis using a decrementing header traceroute-style approach came out of discussions several years ago, between the author, Kaynam Hedayat, Nagarjuna Venna, and Patrick MeLampy. Other people that have contributed to the topic over the years since then: Brett Tate, Paul Kyzivat, Peter Dawes, Zaid Ally, Dianna Stiller, Jon Boone, and several others whom I have lost the names of since.

Author's Address

Hadriel Kaplan
Oracle
EMail: hadrielk@yahoo.com