Internet Engineering Task Force (IETF)
Request for Comments: 7206
Category: Informational
ISSN: 2070-1721

P. Jones
G. Salgueiro
J. Polk
Cisco Systems
L. Liess
Deutsche Telekom
H. Kaplan
Oracle

May 2014

Requirements for an End-to-End Session Identifier in IP-Based Multimedia Communication Networks

Abstract

This document specifies the requirements for an end-to-end session identifier in IP-based multimedia communication networks. This identifier would enable endpoints, intermediate devices, and management and monitoring systems to identify a session end-to-end across multiple SIP devices, hops, and administrative domains.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

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). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see 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/rfc7206.

Table of Contents

1. Introduction

IP-based multimedia communication systems like SIP [1] and H.323 [2] have the concept of a "call identifier" that is globally unique. The identifier is intended to represent an end-to-end communication session from the originating device to the terminating device. Such an identifier is useful for troubleshooting, session tracking, and so forth.

Unfortunately, there are a number of factors that mean that the current call identifiers defined in SIP and H.323 are not suitable for end-to-end session identification. Perhaps most significant is the fact that the syntax for the call identifier in SIP and H.323 is different between the two protocols. This important fact makes it impossible for call identifiers to be exchanged end-to-end when a network uses both of these session protocols.

Another reason why the current call identifiers are not suitable to identify the session end-to-end is that in real-world deployments, devices like Back-to-Back User Agents (B2BUAs) often change the values as the session signaling passes through. This is true even when a single session protocol is employed and is not a byproduct of protocol interworking.

Lastly, identifiers that might have been used to identify a session end-to-end fail to meet that need when sessions are manipulated through supplementary service interactions. For example, when a session is transferred or if a private branch exchange (PBX) joins or merges two communication sessions together locally, the end-to-end properties of currently defined identifiers are lost.

This document specifies the requirements for an end-to-end session identifier in IP-based multimedia communication networks. This identifier would enable endpoints, intermediate devices, and management and monitoring systems to identify a session end-to-end across multiple SIP devices, hops, and administrative domains.

2. Conventions Used in This Document

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 RFC 2119 [3] when they appear in ALL CAPS. These words may also appear in this document in lower case as plain English words, absent their normative meanings.

3. Terminology

3.1. What Does the Session Identifier Identify?

The identifier on which this document places requirements, the session identifier, identifies a set of signaling messages associated with exactly two endpoints that, from each endpoint's perspective, are related to a single invocation of a communication application.

How the endpoints determine which signaling messages share a given identifier (that is, what constitutes a single invocation of a communication application) is intentionally left loosely defined.

The term "call" is often used as an example of such an invocation for voice and video communication, but different protocols and deployments define the scope of a "call" in different ways. For instance, some systems would associate all of the activity between all three parties involved in a transfer as a single "call".

Similarly, the term "session" is often used as an example of such an invocation, but this term is overloaded to describe both signaling and media-level interaction. A single invocation of the communication application, as described above, may involve multiple RTP "sessions" as described by RFC 3550 [4], and possibly even multiple concurrent sessions.

In this document, unless otherwise qualified, the term "communication session", or simply "session", will refer only to the set of signaling messages identified by the common session identifier. That is, a "session" is a set of signaling messages associated with exactly two endpoints that, from each endpoint's perspective, are related to a single invocation of a communication application.

The requirements in this document put some constraints on what an endpoint will consider the same, or a different, invocation of a communication session. They also ensure that related sessions (as this document is using the term) can be correlated using only the session identifiers for each session. Again, what constitutes a "related" session is intentionally left loosely defined.

The definition considers messages associated with exactly two endpoints instead of messages sent between two endpoints to allow for intermediaries that create messages on an endpoint's behalf. It is possible that an endpoint may not see all of the messages in a session (as this document is using the term) associated with it.

This definition, along with the constraints imposed by the requirements in this document, facilitates specifying an identifier that allows the two endpoints to use two entirely different protocols (and hence to potentially have different ideas of what a single invocation means) or use two applications that have a different idea of what a single invocation means.

3.2. Communication Session

A communication session may exist between two SIP User Agents (UAs) and may pass through one or more intermediary devices, including B2BUAs or SIP proxies. For example:

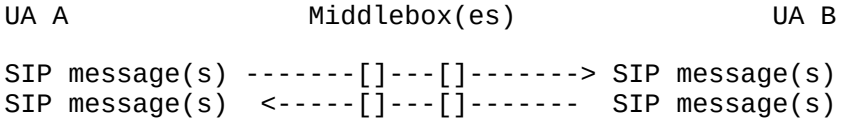

Figure 1: Communication Session through Middlebox(es)

The following are examples of acceptable communication sessions as described in Section 3.1 and are not exhaustive:

- A call directly between two user agents
- A call between two user agents with one or more SIP middleboxes in the signaling path
- A call between two user agents that was initiated using third-party call control (3PCC) [5]
- A call between two user agents (e.g., between Alice and Carol) that results from a different communication session (e.g., Alice and Bob) wherein one of those user agents (Alice) is transferred to another user agent (Carol) using a REFER request or a re-INVITE request

The following are not considered communication sessions:

- A call between any two user agents wherein two or more user agents are engaged in a conference call via a conference focus:
  - each call between the user agent and the conference focus would be a communication session, and
  - each of these is a distinct communication session.

- o A call between three user agents (e.g., Alice, Bob, and Carol) wherein the first user agent (Alice) ad hoc conferences the other two user agents (Bob and Carol):

  - The call between Alice and Bob would be one communication session.

  - The call between Alice and Carol would be a different communication session.

3.3. End-to-End

The term "end-to-end" in this document means the communication session from the point of origin, passing through any number of intermediaries, to the ultimate point of termination. It is recognized that legacy devices may not support the end-to-end session identifier. Since such an endpoint will not create a session identifier, an intermediary device that supports this identifier can inject an identifier into the session signaling.

4. Session Identifier Use Cases

4.1. End-to-End Identification of a Communication Session

For SIP messaging that either does not involve SIP servers or only involves SIP proxies, the Call-ID header field value sufficiently identifies each SIP message within a transaction (see Section 17 of [1]) or dialog (see Section 12 of [1]). This is not the case when either B2BUAs or Session Border Controllers (SBCs) [6] are in the signaling path between User Agents (UAs). Therefore, we need the ability to identify each communication session through a single SIP header field, regardless of which types of SIP servers are in the signaling path between UAs. For messages that create a dialog, each message within the same dialog MUST use the same session identifier.

Derived Requirements: All Requirements in Section 5.

4.2. Protocol Interworking

A communication session might originate on an H.323 [2] endpoint and pass through an SBC before ultimately reaching a terminating SIP user agent. Likewise, a call might originate on a SIP user agent and terminate on an H.323 endpoint. It MUST be possible to identify such sessions end-to-end across the plurality of devices, networks, or administrative domains.

It is anticipated that the ITU-T will define protocol elements for H.323 to make the end-to-end signaling possible.

Derived Requirements: REQ5, REQ7 (Section 5).

4.3. Traffic Monitoring

UA A and UA B communicate using SIP messaging with a SIP B2BUA acting as a middlebox that belongs to a SIP service provider. For privacy reasons, the B2BUA changes the SIP header fields that reveal information related to the SIP users, devices, or domain identities. The service provider uses an external device to monitor and log all SIP traffic coming to and from the B2BUA. In the case of failures reported by the customer or when security issues arise (e.g., theft of service), the service provider has to analyze the logs from the past several days or weeks and then correlates those messages that were messages for a single end-to-end SIP session.

For this scenario, we must consider three particular use cases:

a) UAs A and B support the end-to-end session identifier.

   Derived Requirements: REQ1, REQ3, REQ4, REQ6.

b) Only UA A supports the end-to-end session identifier; UA B does not.

   Derived Requirements: REQ1, REQ3, REQ4, REQ5, REQ6.

c) UAs A and B do not support the end-to-end session identifier.

   Derived Requirements: REQ1, REQ3, REQ4, REQ5, REQ6.

4.4. Tracking Transferred Sessions

It is difficult to track which SIP messages were involved in the same call across transactions, especially when invoking supplementary services such as call transfer or call join. There exists a need for the ability to track communication sessions as they are transferred, one side at a time, until completion of the session (i.e., until a BYE is sent).

Derived Requirements: REQ1, REQ2, REQ9.

4.5. Session Signal Logging

An after-the-fact search of SIP messages to determine which messages were part of the same transaction or call is difficult when B2BUAs and SBCs are involved in the signaling between UAs. Mapping more than one Call-ID together can be challenging because all of the values in SIP header fields on one side of the B2BUA or SBC will likely be different than those on the other side. If multiple B2BUAs and/or SBCs are in the signaling path, more than two sets of header field values will exist, creating more of a challenge. Creating a common header field value through all SIP entities will greatly reduce any challenge for the purposes of debugging, communication tracking (such as for security purposes in case of theft of service), etc.

Derived Requirements: REQ1, REQ3, REQ5, REQ6.

4.6. Identifier Syntax

A syntax that is too lax (e.g., one that allows special characters or a very long identifier) would make it difficult to encode the identifier in other protocols. Therefore, the syntax of the identifier should be reasonably constrained.

Derived Requirement: REQ8.

4.7. 3PCC Use Case

Third-party call control refers to the ability of an entity to create a call in which communication is actually between two or more parties other than the one setting up the call. For example, a B2BUA acting as a third-party controller could establish a call between two SIP UAs using 3PCC procedures as described in Section 4.1 of RFC 3725 [5], the flow for which is reproduced below.

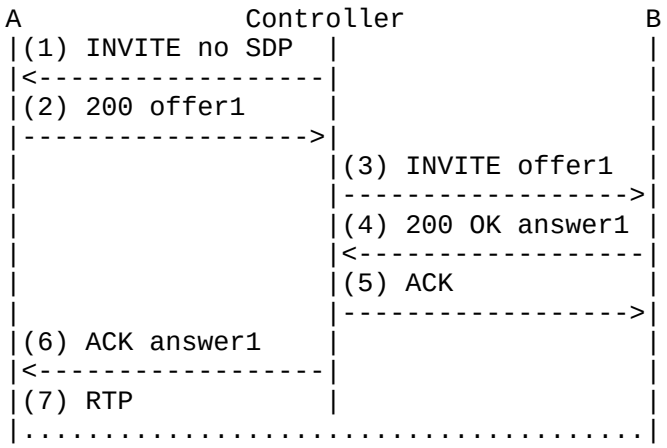

Figure 2: Session Identifier 3PCC Scenario

Such a flow must result in a single session identifier being used for the communication session between UA A and UA B. This use case does not extend to three SIP UAs.

Derived Requirement: REQ9.

5. Requirements for the End-to-End Session Identifier

The following requirements are derived from the use cases and additional constraints regarding the construction of the identifier.

REQ1: It MUST be possible for an administrator or an external device that monitors the SIP traffic to use the identifier to identify those dialogs, transactions, and messages that were at some point in time components of a single end-to-end SIP session (e.g., parts of the same call).

REQ2: It MUST be possible to correlate two end-to-end sessions when a session is transferred or if two different sessions are joined together via an intermediary (e.g., a PBX).

REQ3: The solution MUST require that the identifier, if present, pass unchanged through SIP B2BUAs or other intermediaries.

REQ4: The identifier MUST NOT reveal any information related to any SIP user, device, or domain identity. Additionally, it MUST NOT be possible to correlate a set of session identifiers produced over a period of time with one another, or with a particular user or device. This includes any IP address, port, hostname, domain name, username, Address-of-Record, Media Access Control (MAC) address, IP address family, transport type, subscriber ID, Call-ID, tags, or other SIP header field or body parts.

REQ5: It MUST be possible to identify SIP traffic with an end-to-end session identifier from and to end devices that do not support this new identifier, such as by allowing an intermediary to inject an identifier into the session signaling.

REQ6: The identifier SHOULD be unique in time and space, similar to the Call-ID.

REQ7: The identifier SHOULD be constructed in such a way as to make it suitable for transmission in SIP [1] and H.323 [2].

REQ8: The identifier SHOULD use a restricted syntax and length so as to allow the identifier to be used in other protocols.

REQ9: It MUST be possible to correlate two end-to-end sessions when the sessions are created by a third-party controller using 3PCC procedures as shown in Figure 1 of RFC 3725 [5].

6. Related Work in Other Standards Organizations

6.1. Coordination with the ITU-T

IP multimedia networks are often comprised of a mix of session protocols like SIP [1] and H.323 [2]. A benefit of the session identifier is that it uniquely identifies a communication session end-to-end across session protocol boundaries. Therefore, the need for coordinated standardization activities across Standards Development Organizations (SDOs) is imperative.

To facilitate this, a parallel effort is underway in the ITU-T to introduce the session identifier for H.323 in such a way as to be interoperable with the procedures defined by the IETF.

6.2. Requirements within 3GPP

The Third Generation Partnership Project (3GPP) identified in their Release 9 the need for a session identifier for operation and maintenance purposes to correlate flows in an end-to-end communication session. 3GPP TS24.229 [7] points to the fact that the session identifier can be used to correlate SIP messages belonging to the same session. In the case where signaling passes through SIP entities like B2BUAs, the end-to-end session identifier indicates that these dialogs belong to the same end-to-end SIP communication session.

7. Security Considerations

The security vulnerabilities, attacks, and threat models affecting other similar SIP identifiers are well documented in RFC 3261 [1] and are equally applicable to the end-to-end session identifier and subject to the same mitigating security best practices. Further, storage of the session identifier in a log file is also subject to the security considerations specified in RFC 6872 [8].

An end-to-end identifier, if not properly constructed, could provide confidential information that would allow one to identify the individual, device, or domain initiating or terminating a communication session. In adhering to REQ4, the solution produced in accordance with these requirements MUST take appropriate measures to properly secure and obfuscate sensitive or private information that might allow one to identify a person, device, or domain. This means that the end-to-end session identifier MUST NOT reveal information elements such as the MAC address or IP address. It is outside the scope of this document to specify the implementation details of such security and privacy measures. Those details may vary with the specific construction mechanism selected for the end-to-end session identifier and therefore will be discussed in the document specifying the actual end-to-end identifier.

A key security consideration is to ensure that an attacker cannot surreptitiously spoof the identifier and effectively render it useless to diagnostic equipment that cannot properly correlate signaling messages due to the duplicate session identifiers that exist in the same space and time. In accordance with REQ6, this end-to-end identifier MUST be sufficiently long and random to prevent it from being guessable as well as avoid collision with another identifier. The secure transport of the identifier, need for authentication, encryption, etc. should be appropriately evaluated based on the network infrastructure, transport domain, and usage scenarios for the end-to-end session identifier.

8. Acknowledgments

The authors would like to acknowledge Paul Kyzivat, Christer Holmberg, Charles Eckel, Andy Hutton, Salvatore Loreto, Keith Drage, and Chris Pearce for their contribution and collaboration in developing this document.

9. Contributors

Roland Jesske and Parthasarathi Ravindran provided substantial contributions to this document during its initial creation.

10. References

10.1. Normative References

[1] 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.

[2] Recommendation ITU-T H.323, "Packet-based multimedia communications systems", December 2009.

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

10.2. Informative References

[4] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

[5] Rosenberg, J., Peterson, J., Schulzrinne, H., and G. Camarillo, "Best Current Practices for Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP)", BCP 85, RFC 3725, April 2004.

[6] Hautakorpi, J., Ed., Camarillo, G., Penfield, R., Hawrylyshen, A., and M. Bhatia, "Requirements from Session Initiation Protocol (SIP) Session Border Control (SBC) Deployments", RFC 5853, April 2010.

[7] 3GPP TS 24.229, "IP multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3".

[8] Gurbani, V., Ed., Burger, E., Ed., Anjali, T., Abdelnur, H., and O. Festor, "The Common Log Format (CLF) for the Session Initiation Protocol (SIP): Framework and Information Model", RFC 6872, February 2013.

Authors' Addresses

Paul E. Jones
Cisco Systems, Inc.
7025 Kit Creek Rd.
Research Triangle Park, NC 27709
USA

Phone: +1 919 476 2048
EMail: paulej@packetizer.com
IM: xmpp:paulej@packetizer.com

Gonzalo Salgueiro
Cisco Systems, Inc.
7025 Kit Creek Rd.
Research Triangle Park, NC 27709
USA

Phone: +1 919 392 3266
EMail: gsalguei@cisco.com
IM: xmpp:gsalguei@cisco.com

James Polk
Cisco Systems, Inc.
3913 Treemont Circle
Colleyville, TX
USA

Phone: +1 817 271 3552
EMail: jmpolk@cisco.com
IM: xmpp:jmpolk@cisco.com

Laura Liess
Deutsche Telekom NP
64295 Darmstadt
Heinrich-Hertz-Str. 3-7
Germany

Phone: +49 6151 268 2761
EMail: laura.liess.dt@gmail.com

Hadriel Kaplan
Oracle
71 Third Ave.
Burlington, MA 01803
USA

EMail: hadriel.kaplan@oracle.com