Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: April 28, 2017

P. Dawes
Vodafone Group
C. Arunachalam
Cisco Systems

October 25, 2016

Requirements for Marking SIP Messages to be Logged
draft-ietf-insipid-logme-reqs-10

Abstract

SIP networks use signaling monitoring tools to debug customer reported problems and for regression testing if network or client software is upgraded. As networks grow and become interconnected, including connection via transit networks, it becomes impractical to predict the path that SIP signaling will take between clients, and therefore impractical to monitor SIP signaling end-to-end.

This draft describes requirements for adding an indicator to the SIP protocol data unit (PDU, or a SIP message) that marks the PDU as a candidate for logging. Such marking will typically be applied as part of network testing controlled by the network operator and not used in regular client signaling. However, such marking can be carried end-to-end including the SIP terminals, even if a session originates and terminates in different networks.

Status of This Memo

Table of Contents

1. Introduction

Service providers, enterprises, and others who operate networks that use SIP (see [RFC3261]) need the ability to debug end user reported problems and also to run regression tests if SIP client software/

hardware is upgraded. Such debugging and tests might be confined to a single service provider or network, or may occur between the administrative domains of different network operators, including domains in different countries that are interconnected through networks belonging to one or more third parties.

A mechanism is needed to mark particular SIP sessions, i.e. those related to debugging or regression testing, as candidates for logging and this marking must be carried within the candidate SIP messages as they are routed across networks (and geographies) to enable logging at each SIP entity without having to know in advance the list of SIP entities through which the SIP signaling messages will traverse. Such marking must take into account that SIP messages might traverse different network operators, different countries, regions with different privacy requirements, and different trust domains. This draft describes the requirements for such a "log me" marker for SIP signaling.

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 [RFC2119].

3. Terminology

3.1. Network Boundary

A network boundary is the part of a signaling path where messages pass between entities that are under different administrative control. [RFC5853] Figure 2 shows a network boundary between GW-A1 in operator A's network and the SBC in operator B's network. A network boundary is significant in this document because manipulation of signaling at the boundary could prevent end-to-end testing or troubleshooting.

[RFC5853] gives examples of manipulating signaling to prevent the sending network passing on sensitive information, for example topology hiding, or the receiving network protecting itself from signaling that is not under its control, for example protocol repair. Example SIP device types (see [RFC7092]) that might manipulate signaling at a network boundary are a Session Border Controller performing protocol repair or Interconnection Border Control Function (IBCF) performing topology hiding.

3.2. Trust Domain

In this document a trust domain is the set of entities that have been identified, by prior agreement, as participating elements in logging, typically for the purpose of debugging or regression testing. A trust domain contains all SIP entities under configuration control of the network operator that is performing regression testing plus all SIP entities that are under configuration control of peer network operators who have agreed to participate in that regression testing. The purpose of trust domain requirements is to prevent network operators inadvertently triggering logging in networks that are not part of any testing or troubleshooting.

3.3. Intermediary

The term "intermediary" is defined in [RFC7989] section 2 and refers to any entity along the call signaling path.

4. Motivating Scenario

4.1. Introduction

Signaling for SIP session setup can cross several networks, and these networks may not have common ownership and also may be in different countries. If a single operator wishes to perform regression testing or fault debugging end-to-end, the separate ownership of networks that carry the signaling and the explosion in the number of possible signaling paths through SIP entities from the originating to the terminating user make it impractical to pre-configure logging of an end-to-end SIP signaling of a session of interest.

4.2. Example Network Arrangement

The figure below gives an example of a signaling path through multiple networks.

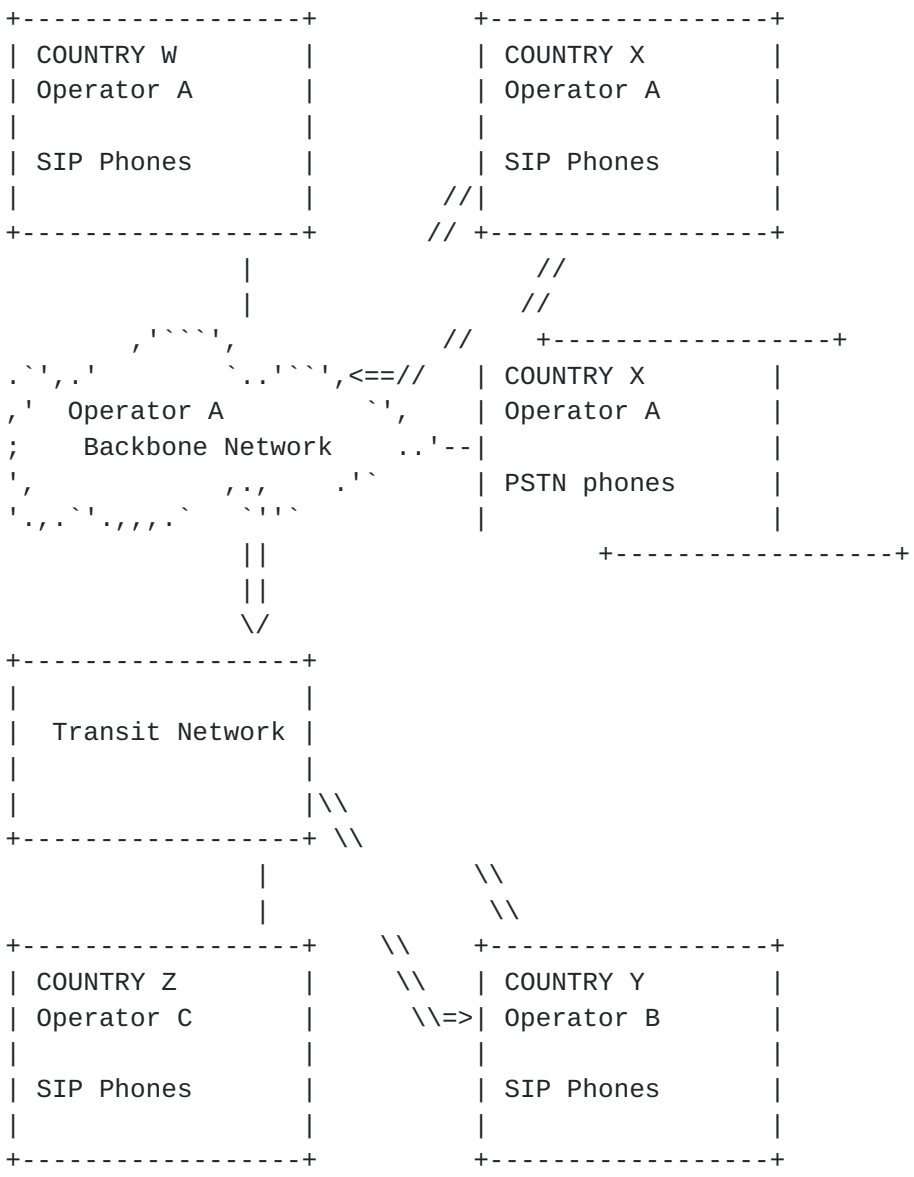

Figure 1: Example signaling path through multiple networks

4.3. Example Debugging Procedure

One possible set of steps is outlined below to illustrate the debugging procedure.

- The user's terminal is placed in debug mode. The terminal logs its own signaling and inserts a "log me" marker into SIP requests for session setup.

- All SIP entities that the signaling traverses, from the first proxy the terminal connects to at the edge of the network to the

destination client terminal, detect that the "log me" marker is present and log SIP requests and responses that contain the marker if configured to do so.

- Subsequent responses and requests in the same dialog are also marked with a "log me" marker.

- Logging stops, either because the dialog has ended or because a 'stop event', typically expiry of a certain amount of time, occurred.

- Logs are retrieved, for example by logging on to the SIP entity or entities that contain the logs.

5. Logme Marking Requirements

5.1. Message Logs

- REQ1: The entire SIP message (SIP headers and message body) MUST be logged using the SIP CLF format defined in [RFC6873], with Vendor-ID = 00000000 and Tag = 02 in the <OptionalFields> portion of the SIP CLF record (see [RFC6873] clause 4.4).

- REQ2: Header fields SHOULD be logged in the form in which they appear in the message, they SHOULD NOT be converted between long and compact forms described in [RFC3261] clause 7.3.3.

When and how signaling logs are retrieved is out of scope of this document. Logs might be retrieved by logging on to the SIP entity that contains the logs, by sending logs to a central server that is co-ordinating debugging, by storing them on removable media for later manual collection, or by some other method.

5.2. "Log Me" Marking

- REQ3: It MUST be possible to mark a SIP request or response as of interest for logging by inserting a "log me" marker. This is known as "log me" marking.

- REQ4: It MUST be possible for a "log me" marker to cross network boundaries.

- REQ5: A "log me" marker MAY include an identifier that indicates the test case that caused it to be inserted, known as a test case identifier. The test case identifier does not have any impact on session setup, it is used by the debugging server to collate all logged SIP requests and responses to the initial SIP request in a dialog or standalone transaction. The local UUID portion of

Session-ID described in [RFC7206] and [RFC7989] could be used as a random test case identifier.

5.3. Processing the "Log Me" Marker

- REQ6: A "log me" marker is most effective if it passes end-to-end. However, source networks should behave responsibly and not leave it to a downstream network to detect and remove a marker that it will not use. A "log me" marker SHOULD be removed at trust domain boundaries.

- REQ7: The presence of a "log me" marker indicates that a request or response is part of debugging or regression testing. SIP entities that support "log me" marking SHOULD log SIP requests or responses that contain a "log me" marker." The SIP entity checks for the presence of a "log me" marker and writes any request or response that contains a "log me" marker to a log file.

- REQ8: If a UA that supports "log me" marking receives a request with a "log me" marker, it MUST echo that "log me" marker in responses to that request. This requirement applies to cases where the UA is the endpoint of communication, where the UA is one side of a gateway such as a SIP/PSTN gateway, and where the UA is one side of a B2BUA.

- REQ9: A SIP intermediary MAY insert a "log me" marker into requests and responses. The typical case for which a intermediary needs to insert a "log me" marker is for compatibility with UAs that have not implemented "log me" marking, i.e. when a UA has not marked a request or when responses received on a dialog of interest for logging do not contain an echoed "log me" marker. Another use case is when the session origination UA that inserted log me marker is no longer participating in the session (e.g., call transfer scenarios) and the intermediary adds "log me" marker in related sessions to enable end-to-end signaling analysis. In these cases, the entity that inserts a "log me" marker is stateful inasmuch as it must remember when a dialog is of interest for logging. An entity that inserts a "log me" marker SHOULD also log the SIP request or response as per REQ4.

- REQ10: SIP intermediaries MAY be stateless in terms of logging of SIP requests that contain a "log me" marker, i.e. they MAY base the decision to log a SIP request or response solely on the presence of the "log me" marker. For example, it is OPTIONAL for a SIP entity to maintain state of which SIP requests contained a "log me" marker in order to log responses to those requests. Echoing a "log me" marker in responses is the responsibility of the UA that receives a request.

- REQ11: "log me" marking of requests and responses MUST be applied on a per-dialog granularity. If applied, "log me" marking MUST begin with the dialog-creating request and SHOULD continue to the dialog end. "log me" marking SHOULD be applied to in-dialog requests and responses in either direction. "log me" marking MUST NOT be stopped and re-started on a given dialog.

The definition of types of events that cause logging to stop and configuring SIP entities to detect such "stop events" is outside the scope of this document.

6. Security Considerations

In order to prevent any security implications of a "log me" marker, the marker itself MUST NOT contain any sensitive information, detecting its presence or absence MUST NOT reveal sensitive information, and maliciously adding a "log me" marker MUST NOT adversely affect a network. This section analyses how to meet these requirements.

6.1. Trust Domain

Since a "log me" marker may cause a SIP entity to log the SIP header and body of a request or response, the "log me" marker SHOULD be removed at a trust domain boundary. If a prior agreement to log sessions exists with the next hop network then the "log me" marker SHOULD NOT be removed.

6.2. Security Threats

6.2.1. "Log Me" Marking

The "log me" marker MUST NOT convey any sensitive information, although the "log me" marker will sometimes be inserted because a particular device is experiencing problems.

The presence of a "log me" marker might cause some SIP entities to log signaling. Therefore, this marker MUST be removed at the earliest opportunity if it has been incorrectly inserted.

Activating a debug mode affects the operation of a terminal, therefore debugging configuration MUST be supplied by an authorized party to an authorized terminal, debugging configuration MUST NOT be altered in transit, and MUST NOT be readable by an unauthorized third party.

Logged signaling is privacy-sensitive data, therefore signaling logs MUST NOT be readable by an unauthorized third party.

6.2.2. Logged Information

A SIP entity that has logged information should prevent unauthorized access to that logged information.

7. IANA Considerations

There are no IANA considerations associated with this document.

8. Acknowledgments

The authors wish to thank Jorgen Axell, Keith Drage, Vijay Gurbani, Christer Holmberg, Hadriel Kaplan, Paul Kyzivat, James Polk, Gonzalo Salgueiro, Alberto Llamas, Brett Tate and Paul Giralt for their constructive comments and guidance while developing this document.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC6873] Salgueiro, G., Gurbani, V., and A. Roach, "Format for the Session Initiation Protocol (SIP) Common Log Format (CLF)", RFC 6873, DOI 10.17487/RFC6873, February 2013, <http://www.rfc-editor.org/info/rfc6873>.

9.2. Informative References

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <http://www.rfc-editor.org/info/rfc3261>.

[RFC5853] 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, DOI 10.17487/RFC5853, April 2010, <http://www.rfc-editor.org/info/rfc5853>.

[RFC7092] Kaplan, H. and V. Pascual, "A Taxonomy of Session Initiation Protocol (SIP) Back-to-Back User Agents", RFC 7092, DOI 10.17487/RFC7092, December 2013, <http://www.rfc-editor.org/info/rfc7092>.

[RFC7206] Jones, P., Salgueiro, G., Polk, J., Liess, L., and H. Kaplan, "Requirements for an End-to-End Session Identification in IP-Based Multimedia Communication Networks", RFC 7206, DOI 10.17487/RFC7206, May 2014, <http://www.rfc-editor.org/info/rfc7206>.

[RFC7989] Jones, P., Salgueiro, G., Pearce, C., and P. Giralt, "End-to-End Session Identification in IP-Based Multimedia Communication Networks", RFC 7989, DOI 10.17487/RFC7989, October 2016, <http://www.rfc-editor.org/info/rfc7989>.

Authors' Addresses

Peter Dawes
Vodafone Group
The Connection
Newbury, Berkshire RG14 2FN
UK

Email: peter.dawes@vodafone.com

Chidambaram Arunachalam
Cisco Systems
7200-12 Kit Creek Road
Research Triangle Park, NC, NC 27709
US

Email: carunach@cisco.com