Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: January 8, 2017

P. Dawes
Vodafone Group
C. Arunachalam
Cisco Systems

July 7, 2016

Requirements for Marking SIP Messages to be Logged
draft-ietf-insipid-logme-reqs-07

Abstract

SIP networks use signalling 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 signalling will take between clients, and therefore impractical to monitor SIP signalling 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 signalling. However, such marking can be carried end-to-end including the SIP terminals, even if a session originates and terminates in different networks.

Table of Contents

1. Introduction

Service providers who use SIP (see RFC 3261 [RFC3261]) in their networks need the ability to debug customer reported problems and also need 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 service providers, including providers 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 service providers, different countries, regions with different privacy requirements, and different trust domains. This draft describes the requirements for such a "log me" marker for SIP signalling.

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 [RFC2119].

3. Motivating Scenario

Signalling 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 signalling and the explosion in the number of possible signalling paths through SIP entities from the originating to the terminating user make it impractical to pre-configure logging of an end-to-end SIP signalling of a session of interest.

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

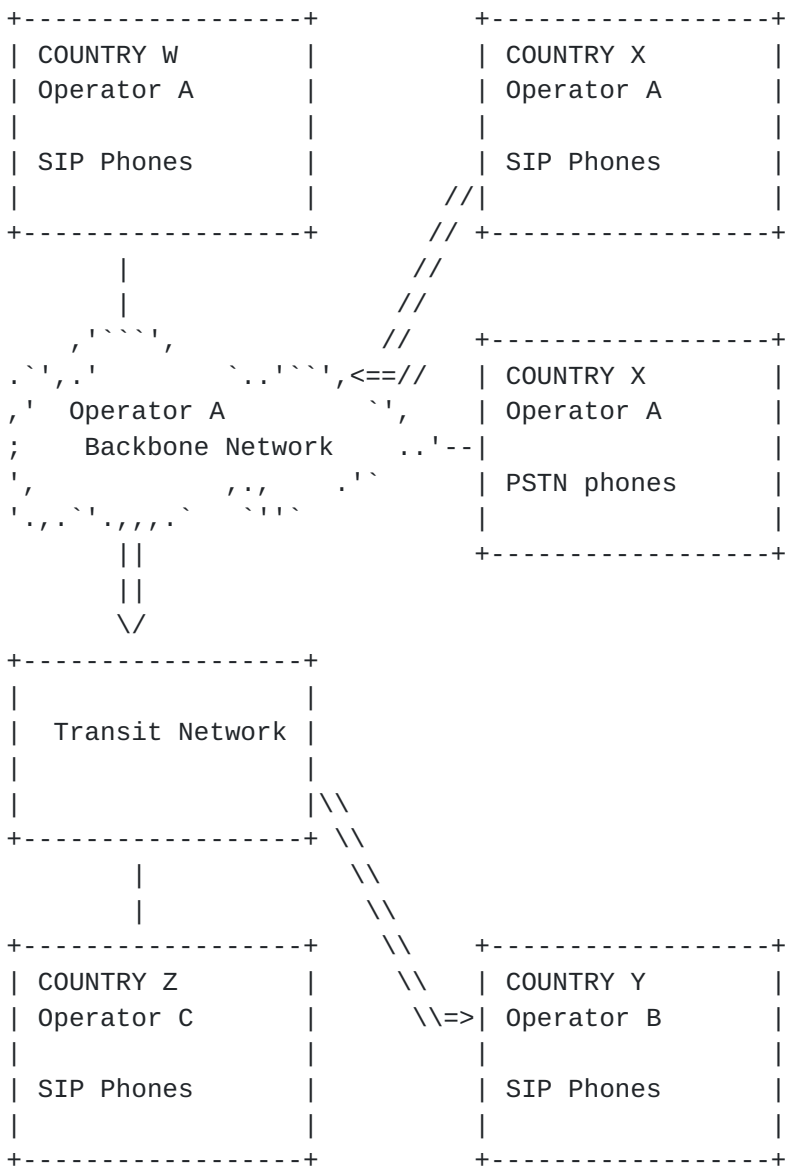

Figure 1: Example signalling path through multiple networks

4. Basic Debugging Procedure

The debugging procedure steps are outlined below. The entire SIP message (SIP headers and message body) MUST be logged using the SIP CLF format defined in RFC 6873 [RFC6873], with Vendor-ID = 00000000 and Tag = 02 in the <OptionalFields> portion of the SIP CLF record (see RFC 6873 [RFC6873] clause 4.4). Header fields MUST be logged in their long form and not the compact form described in RFC 3261 [RFC3261] clause 7.3.3.

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

- All SIP entities that the signalling traverses, from the first proxy the terminal connects to at the edge of the network to the destination client terminal, can detect that the "log me" marker is present and can log SIP requests and responses that contain the marker if configured to do so.

- Subsequent responses and requests in the same dialog are logged.

- Logging stops, either because the dialog has ended or because a 'stop event', typically expiry of a certain amount of time, occurred. The definition of stop event types and the configuration of stop events in the SIP entity is outside the scope of this document.

- When and how signalling 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. Requirements for a "Log Me" Marker

- REQ1: 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.

- REQ2: It MUST be possible for a "log me" marker to cross network boundaries.

- REQ3: 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.

- REQ4: 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.

- REQ5: 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.

- REQ6: A SIP proxy MAY insert a "log me" marker into requests and responses. The typical case for which a proxy needs to insert a "log me" marker is for compatibility with UAs that have not implemented "logme" 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. 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.

- REQ7: SIP proxies 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.

- REQ8: 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 Session-ID described in RFC 7206 [RFC7206] and I-D.ietf-insipid-session-id-12 [I-D.ietf-insipid-session-id] could be used as the test case identifier but it would be useful for the UA to log a human readable name together with this Session-ID when it performs "log me" marking of an initial SIP request.

- REQ9: "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 MUST NOT be stopped and re-started on a given dialog.

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 might 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 will cause some SIP entities to log signalling. 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 server to an authorized terminal, debugging configuration must not be altered in transit, and must not be readable by an unauthorized third party.

Logged signalling is privacy-sensitive data, therefore signalling logs must be passed to an authorized server, must not be altered in transit, and must not be readable by an unauthorized third party.

6.2.2. Sending Logged Information

A SIP entity that has logged information should encrypt it, such that it can be decrypted only by the debug server, before sending it to a debug server in order to protect the content of logs from a third party.

7. Acknowledgments

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

8. References

8.1. 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>.

8.2. Informative References

[I-D.ietf-insipid-session-id] Jones, P., Polk, J., Salgueiro, G., and C. Pearce, "End-to-End Session Identification in IP-Based Multimedia Communication Networks", draft-ietf-insipid-session-id-12 (work in progress), January 2015.

[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>.

[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>.

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