Network Working Group
Internet-Draft
Intended status: Experimental
Expires: April 13, 2011

G. Golovinsky
Alert Logic, Inc.
S. Johnston
Google
Z. Fox
Alert Logic, Inc.

October 10, 2010

Syslog Extension for Cloud Using Syslog Structured Data
draft-cloud-log-00

Abstract

This document provides an open and extensible log format to be used by any cloud entity or cloud application to log and trace activities that occur in the cloud. It is equally applicable for cloud infrastructure (IaaS), platform (PaaS), and application (SaaS) services. CloudLog is defferent in content, but not in nature from the traditional logging as it takes in account transient nature of identities and resources in the cloud.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

This document describes a standard for syslog structured data elements in messages generated by services that may be running on different physical or virtual machines when those services are processing information generated by a single request. The purpose of which is to provide an audit trail that allows correlation of such messages. In addition, this document defines a number of parameters that MUST or SHOULD be included in these structured data elements so these messages can be used to identify users of such services, when the real and/or effective identities of users is known.

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. Problem Statement

3.1. The Traditional Logging and its Applications

Practically all hardware and software entities deployed on the network log their activities. Network elements such as routers, servers, firewalls and switches log information about their activities using mostly Syslog (except for Windows). Applications running on the network also log activities, but often using proprietary mechanisms. While logging mechanisms are inconsistent between different entities - Syslog, Windows events, proprietary files - they generally carry enough information to identify type of the activity, time of the occurrence, physical entity involved in the event, and often user(s) that participated in the event. Availability of this information is crucial for accomplishing multiple business objectives ranging from assuring security and performing forensics to adhering to compliance regulations (SOX, PCI, etc.). The existence of logs and information in them is necessary, but not sufficient for achieving security, compliance and other business objectives. The process of collecting, processing, searching and even simply interpreting information in logs is exceptionally labor and time consuming process and often cannot even be done on any meaningful scale without appropriate tools in place. Log Management tools used to solve the problem of scale and interpretation heavily depend on the fact that format of logs is largely well defined and understood.

3.2. Challenges with the cloud deployment

In cloud deployments the situation with availability of logs in reliability of information in them is drastically different. By definition, cloud resources are shared. A piece of hardware is now running multiple Virtual Instances of "it". They can be brought up and down within very short period of time and at any given moment the hardware can be shared not just by different users but by different users from different companies. Even if Linux or Windows VMs continue to log their activity the information in these logs is very likely to be irrelevant since you cannot really tie logs to the physical entity. Moreover, even if one managed to map logs to a physical entity, there is absolutely no guarantee that the same VM image will be running on the same hardware in its next reincarnation. And there is really no clear way to determine how many users share the hardware and what are their identities and roles. Tracing environmental changes is practically impossible task unless there is traceability between physical and virtual entities. As a result, achieving such business objectives as adhering to compliance regulations or performing regular security auditing is very difficult if not an impossible task.

4. Cloud Log Structured Data Definitions

1. RUI - real user identity, the identity of the user that has authenticated to the entity.

2. EUI - effective or impersonated user identity, the identity of the user that the real user identity is acting for. For example, an administrator account could have the ability to impersonate another user account.

3. Provider - is the domain, service, application, or other entity providing the user identities.

Structured data elements, defined in RFC 5424 [RFC5424], provides a mechanism for adding data to syslog messages. Since additional data is necessary to trace user identities and their activities in the cloud we use the mechanism of structured data elements to provide this additional information in the syslog messages.

4.1. SD-ELEMENT context

The SD-ELEMENT identified by the SD-ID "context" defines the context of the external request that causes for the activity to take place. The syslog message that is generated as a result of this activity should be identified by this "context".

4.1.1. SD-PARAM aid - Mandatory

The parameter "aid" represents the audit identifier, which uniquely identifies an external request for activity. The value is a UTF-8-STRING representation of the UUID generated by the entity when request is received.

This parameter MUST be present within the SD-ELEMENT "context".

4.1.2. SD-PARAM provider - Optional

The parameter "provider" represents the provider of the identity for the Real User Identity - 'rid' and Effective User Identity - 'eid', User identities are not always exist or available. In cases that they are, either "rid" or "eid" MUST be present in the syslog messages.

The parameter "provider" is not required, but SHOULD be present within the SD-ELEMENT "context" when either the 'rid' or 'eid' identifiers are present.

4.1.3. SD-PARAM rid - Optional

The parameter "rid" represents the real user identity.

This parameter SHOULD be present within the SD-ELEMENT "context" when the real user identity is availbale.

4.1.4. SD-PARAM eid - Optional

The parameter "eid" represents the effective user identity. This parameter SHOULD be present within the SD-ELEMENT "context" when user impersonation has happened and the effective user identity is available.

The 'eid' parameter represents the effective user identity.

This parameter SHOULD be present within the 'context' SD-ELEMENT when the effective user identity is known.

4.2. SD-ELEMENT transit

The SD-ELEMENT identified by the SD-ID "transit" defines logical gateway entities which were traversed while request for activity was routed to the final destination entity that would satisfy the request.

4.2.1. SD-PARAM client - Mandatory

The parameter "client" represents the IP address or Fully Qualified Domain Name (FQDN) of the client entity on behalf of which the request is being made. This is different from SD-ID 'ip' in RFC 5424 that defines IP of the entity producing the log message itself. IPv4 or IPv6 addresses MUST be represented as STRING-UTF-8 .

The parameter "client" represents the IP address or FQDN of the client on behalf of which the request is being made.

4.2.2. SD-PARAM gw - Optional

The parameter "gw" represents a gateway entity through which the request for activity passes before arriving to the final destination entity actually responsible processing of the request. The value of the parameter is comprised of the STRING-UTF-8 representation of UUID of the entity , identifying the gateway, a colon character (i.e. ':'), and finally the STRING-UTF-8 representation of IP address or FQDN of the gateway through which the request has been routed.

This parameter MAY appear more than once within the SD-ELEMENT "transit" as request may pass through multiple gateway entities. Each occurrence represents a different gateway through which the request passed.

5. Log Format Samples

5.1. Log Sample of Simple Non-Authenticated Request

Here is an example of a log produced as a result of simple non-authenticated request to a web service. Only the mandatory parameters "aid" and "client" are represented.

```
Jul 7 09:01:40 [context aid="9BE817EB-8ACC-1004-D9DF-
00000A00065E"][transit client="56.2.222.83"] Initializing request to
/example_api/index
```

```
Jul 7 09:01:40 [context aid="9BE817EB-8ACC-1004-D9DF-
00000A00065E"][transit client="56.2.222.83"] "64.39.0.40" - "1023"
""GET /example_api/index HTTP/1.1"" 200 2543 -- performed in 600 ms
```

5.2. Successful Authenticated User Request

Here is an example of a simple request including user authentication. Note that the 'provider' and 'rid' SD-PARAMs are added to the message after the user has authenticated to the service, and that those

parameters are included in each subsequent message.

```
Aug 16 13:34:18 [context aid="149683FC-8DF5-1004-E1A8-
00000A000152"][transit client="172.16.1.82"] Initializing request to
/api/example:instance/1

Aug 16 13:34:18 [context aid="149683FC-8DF5-1004-E1A8-00000A000152"
provider="example.com" rid="1:123"][transit client="172.16.1.82"]
User authentication successful for 1:123

Aug 16 13:34:18 [context aid="149683FC-8DF5-1004-E1A8-00000A000152"
provider="example.com" rid="1:123"][transit client="172.16.1.82"]
"172.16.1.82" - "-" ""GET /api/example:instance/1 HTTP/1.1"" 200 119
-- performed in 2 ms
```

5.3. Log Sample of Successful Request on Behalf of Another Identity

Here is a request made by an authenticated user on behalf of another identity. Note that the parameter "eid" is added after the user authentication takes place and the effective user identity is validated. This parameter is included in each subsequent message.

```
Aug 16 13:34:18 [context aid="149683FC-8DF5-1004-E1A8-
00000A000152"][transit client="172.16.1.82"] Initializing request to
/api/example:instance/1

Aug 16 13:34:18 [context aid="149683FC-8DF5-1004-E1A8-00000A000152"
provider="example.com" rid="1:123"][transit client="172.16.1.82"]
User authentication successful for 1:123

Aug 16 13:34:18 [context aid="149683FC-8DF5-1004-E1A8-00000A000152"
eid="2:456" provider="example.com" rid="1:123"][transit
client="172.16.1.82"] User impersonation successful for 1:123 to
2:456

Aug 16 13:34:18 [context aid="149683FC-8DF5-1004-E1A8-00000A000152"
eid="2:456" provider="example.com" rid="1:123"][transit
client="172.16.1.82"] "172.16.1.82" - "-" ""GET /api/
example:instance/1 HTTP/1.1"" 200 119 -- performed in 2 ms
```

6. Security Considerations

In addition to general syslog security considerations discussed in RFC 5424 [RFC5424], he information contained in these messages may provide information about how services interact, user identities, and other information about network or service inventory.

Users should not have access to these messages if they would not have access to this information through other authenticated means.

7. IANA Considerations

7.1. SD-IDs

ANA is requested to register the syslog structured data element SD-IDs and PARAM-NAMEs shown below:

SD-ID	PARAM-NAME	
context		OPTIONAL
	aid	MANDATORY
	eid	OPTIONAL
	provider	OPTIONAL
	rid	OPTIONAL
transit		OPTIONAL
	client	MANDATORY
	gw	OPTIONAL

Table 1

8. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119.

[RFC5424] Gerhards, R., "The Syslog Protocol", RFC 5424.

Authors' Addresses

Gene Golovinsky
Alert Logic, Inc.
1776 Yorktown
Suite 700
Houston, TX 77056
US

Phone: (713) 484-8383
Email: gene@alertlogic.com
URI: www.alertlogic.com

Sam Johnston
Google
Brandschenkestrasse, 110
Zurich,   8002
Switzerland

Phone: +41.446681679
Email: sj@google.com

Zachary Fox
Alert Logic, Inc.
1776 Yorktown
Suite 700
Houston, TX 77056
US

Phone: (713) 484-8383
Email: zfox@alertlogic.com
URI: www.alertlogic.com