Network Working Group
Internet-Draft
Obsoletes: 3195 (if approved)
Intended status: Standards Track
Expires: October 21, 2007

D. New

M. Rose
Dover Beach Consulting, Inc.

E. Lear
Cisco Systems

April 19, 2007

Reliable Delivery for syslog
draft-lear-ietf-syslog-rfc3195bis-01

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on October 21, 2007.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

The syslog protocol describes a number of service options related to propagating event messages. This memo describes a mapping of the syslog protocol to TCP connections, useful for reliable delivery of event messages through the use of a BEEP profile. The earlier RAW and COOKED BEEP syslog profiles are deprecated. The use of syslog over BEEP provides robustness and security in message delivery that is unavailable to the usual UDP-based syslog protocol, by providing encryption and authentication over a connection-oriented protocol.

Table of Contents

1. Introduction 3
2. The Model 4
3. The TARTARE Profile 6
3.1. TARTARE Profile Overview 6
3.2. TARTARE Profile Identification and Initialization 8
3.3. TARTARE Profile Message Syntax 9
3.4. TARTARE Profile Message Semantics 9
4. Additional Provisioning 10
4.1. Message Authenticity 10
4.2. Message Replay 10
4.3. Message Integrity 10
4.4. Message Observation 11
4.5. Summary of Recommended Practices 11
5. Registrations 12
5.1. Registration: The TARTARE Profile 12
6. Reply Codes 13
7. IANA Considerations 14
7.1. Registration: BEEP Profile 14
7.2. Registration: The System (Well-Known) TCP port number for syslog-conn 14
8. Security Considerations 15
9. Acknowledgements 16
10. References 17
10.1. Normative References 17
10.2. Informative References 17
Appendix A. Coexistence with old RAW and COOKED modes 18
Appendix B. Changes from RFC 3195 19
Authors' Addresses 20
Intellectual Property and Copyright Statements 21

1. Introduction

The syslog protocol [1] presents a spectrum of service options for provisioning an event-based logging service over a network. Each option has associated benefits and costs. Accordingly, the choice as to what combination of options is provisioned is both an engineering and administrative decision. This memo describes how to realize the syslog protocol when reliable delivery is selected as a required service. It is beyond the scope of this memo to argue for, or against, the use of reliable delivery for the syslog protocol.

This memo is a revision of previous work [9]. Based on implementation and deployment experience, and the expectation of new work in the field, the principle changes to this document are these:

- Both the COOKED and the RAW profiles are deprecated. The COOKED profile is deprecated because there has been no substantial deployment and it is no longer consistent with the work done in [1].

- The RAW profile is reproduced as a new profile, TARTARE, that has no length limitations. Removal of the 1024 octet limitation is necessary for future worked in the area of "signed syslog". Previous length limitations continue to apply to the deprecated RAW and COOKED profiles.

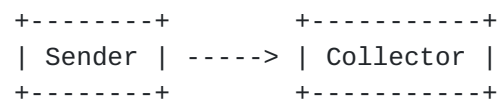
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [2].

2. The Model

The reader is referred to [1] for the authoritative explanation of the syslog model. What follows is a summary of that model, as it is relevant to the BEEP transport mapping for syslog.

The syslog service supports three roles of operation: sender, relay, and collector.

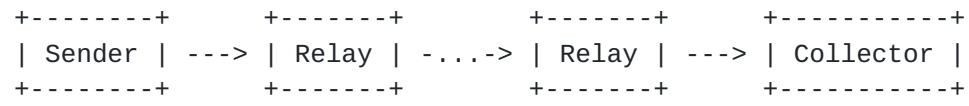
Senders and collectors act as sources and sinks, respectively, of syslog entries. In the simplest case, only a sender and collector are present. E.g.,



The relationship between senders and collectors is potentially many-to-many. I.e., a sender might communicate with many collectors; similarly, a collector might communicate with many senders.

A relay operates in both modes, accepting syslog entries from senders and other relays and forwarding those entries to collectors and other relays.

For example,



As shown, more than one relay may be present between any particular sender and collector.

A relay may be necessary for administrative reasons. For example, a relay might run as an application proxy on a firewall. Also, there might be one relay per company department, which authenticates all the senders in the department, and which in turn authenticates itself to a company-wide collector.

A relay can also serve to filter messages. For example, one relay may collect the syslog information from an entire web server farm, summarizing hit counts for report generation, forwarding "page not found" messages (indicating a possible broken link) to a collector that presents it to the webmaster, and sending more urgent messages (such as hardware failure reports) to a collector that gateways them to a pager. A relay may also be used to convert formats from a sender's output to a collector's input.

It should be noted that a role of sender, relay, or collector is relevant only to a particular BEEP channel (q.v., below). A single server can serve as a sender, a relay, and a collector, all at once, if so configured. It can even serve as a relay and a collector to the same sender or device at the same time using different BEEP channels over the same connection-oriented session; this might be useful to collect status yet relay urgent error messages.

To provide reliable delivery when realizing the syslog protocol, this memo defines a new BEEP profile. BEEP [3] is a generic application protocol framework for connection-oriented, asynchronous interactions. Within BEEP, features such as authentication, privacy, and reliability through retransmission are provided. The new TARTARE profile is designed to provide a high-performance, low-impact footprint, using essentially the same format as the existing UDP-based syslog service.

BEEP defines "transport mappings," specifying how BEEP messages are carried over the underlying transport technologies. At the time of this writing, only one such transport is defined, in [4], which specifies BEEP over TCP. All transport mappings are required to support enough reliability and sequencing to allow all BEEP messages on a given channel to be delivered reliably and in order. Hence, the TARTARE profile provides reliable delivery of messages.

Senders and relays MAY discover relays and collectors via the DNS SRV algorithm [5]. If so configured, the service used is "syslog" and the protocol used is "tcp". This allows for central administration of addressing, fallback for failed relays and collectors, and static load balancing. Security policies and hardware configurations may be such that device configuration is more secure than the DNS server. Hardware devices may be of such limited resources that DNS SRV access is inappropriate. Firewalls and other restrictive routing mechanisms may need to be dealt with before a reliable syslog connection can be established. In these cases, DNS might not be the most appropriate configuration mechanism.

3. The TARTARE Profile

3.1. TARTARE Profile Overview

The TARTARE profile is designed for minimal implementation effort, high efficiency, and backwards compatibility.

It should be noted that even though the TARTARE profile uses the same format for message payloads as the UDP version of syslog uses, delivery is reliable. The TARTARE syslog profile is a profile of BEEP [3], and BEEP guarantees ordered reliable delivery of messages within each individual channel.

When the profile is started, no piggyback data is supplied. All BEEP messages in the TARTARE profile are specified as having a MIME Content-Type [6] of application/octet-stream. Once the channel is open, the listener (not the initiator) sends a MSG message indicating it is ready to act as a syslog sink. (Refer to [3]'s Section 2.1 for a discussion of roles that a BEEP peer may perform, including definitions of the terms "listener", "initiator", "client", and "server".)

The initiator uses ANS replies to supply one or more syslog entries in the current UDP format, as specified in [8]'s Section 3. When the initiator has no more entries to send, it finishes with a NUL reply and closes the channel.

An example might appear as follows:

```
L: <wait for incoming connection>
I: <establish connection>
L: RPY 0 0 . 0 131
L: Content-type: application/beep+xml
L:
L: <greeting>
L:  <profile uri='http://xml.resource.org/profiles/syslog/TARTARE' />
L: </greeting>
L: END
I: RPY 0 0 . 0 52
I: Content-type: application/beep+xml
I:
I: <greeting />
I: END
I: MSG 0 1 . 52 136
I: Content-type: application/beep+xml
I:
I: <start number='1'>
I:  <profile uri='http://xml.resource.org/profiles/syslog/TARTARE' />
```

```
   I: </start>
   I: END
   L: RPY 0 1 . 131 105
   L: Content-type: application/beep+xml
   L:
   L: <profile uri='http://xml.resource.org/profiles/syslog/TARTARE' />
   L: END
   L: MSG 1 0 . 0 50
   L:
   L: Central Services. This has not been a recording.
   L: END
   I: ANS 1 0 . 0 112
   I:
   I: <34>1 2003-10-11T22:14:15.003Z mymachine.example.com su - ID47
      - BOM'su root' failed for lonvick on /dev/pts/8END
   I: ANS 1 0 . 112 101 1
   I:
   I: <165>1 2003-08-24T05:14:15.000003-07:00 192.0.2.1
      myproc 8710 - - %% It's time to make the do-nuts.END
   I: NUL 1 0 . 222 0
   I: END
   L: MSG 0 3 . 236 70
   L: Content-Type: application/beep+xml
   L:
   L: <close number='1' code='200' />
   L: END
   I: RPY 0 3 . 188 46
   I: Content-Type: application/beep+xml
   I:
   I: <ok />
   I: END
   I: MSG 0 4 . 234 72
   I: Content-Type: application/beep+xml
   I:
   I: <close number='0' code='200' />
   I: END
   L: RPY 0 4 . 306 46
   L: Content-type: application/beep+xml
   L:
   L: <ok />
   L: END
   L: <closes connection>
   I: <closes connection>
   L: <awaits next connection>
```

Here we see a BEEP session established, followed by the use of the TARTARE profile. The initiator is a sender, while the listener is a collector. The initiator opens the channel, but the listener sends the first MSG. This allows the initiator to send any number of ANS replies carrying syslog event messages. The initiator sends a NUL reply to indicate it is finished. Upon receiving the NUL, the listener closes the TARTARE channel. The initiator has the choice of closing the entire BEEP session or opening a new syslog channel for more transfers. In this example, the initiator chooses to close the entire BEEP session. (Please note that in the example, as an artifact of the format of this memo, the two lines not beginning with I: or L: have been wrapped.)

The overhead for one ANS frame is about thirty octets, once the initial handshakes have been exchanged. If this overhead is too high, then messages are likely being generated at a high rate. In this case, multiple syslog messages can be aggregated into a single ANS frame, each separated by a CRLF sequence from the preceding. The final message still MUST NOT end with a CRLF.

For example,

```
L: MSG 1 0 . 0 50
L:
L: Central Services. This has not been a recording.
L: END
I: ANS 1 0 . 0 213 0
I:
I: <34>1 2003-10-11T22:14:15.003Z mymachine.example.com su - ID47
   - BOM'su root' failed for lonvick on /dev/pts/8
I: <165>1 2003-08-24T05:14:15.000003-07:00 192.0.2.1
     myproc 8710 - - %% It's time to make the do-nuts.END
I: NUL 1 0 . 213 0
I: END
```

(Again, as an artifact of the format of this memo, the two lines containing syslog entries have been wrapped.)

3.2. TARTARE Profile Identification and Initialization

The TARTARE syslog profile is identified as

http://xml.resource.org/profiles/syslog/TARTARE

in the BEEP "profile" element during channel creation.

No data is piggybacked during channel creation.

3.3. TARTARE Profile Message Syntax

All BEEP messages in this profile have a MIME content-type of application/octet-stream. The listener's first BEEP message is ignored and indeed may be empty except for headers; hence, any syntax is acceptable.

The ANS replies the initiator sends in response MUST be formatted according to Section 6 of [1]. In particular, If the receiver is acting as a relay, then it is expected follow the principles laid out in that specification.

If multiple syslog messages are included in a single ANS reply, each is separated from the preceding with a CRLF. There is no ending delimiter, and there is no length limitation. Note that there MUST NOT be a CRLF between the text of the final syslog event message and the "END" marking the trailer of the BEEP frame.

3.4. TARTARE Profile Message Semantics

The listener's opening BEEP MSG message has no semantics. (It is a good place to put in an identifying greeting.) The initiator's ANS replies MUST specify a facility, severity, and textual message, as described in [1].

4. Additional Provisioning

In more advanced configurations, syslog senders, relays, and collectors can be configured to support various delivery priorities. Multiple channels running the same profile can be opened between two peers, with higher priority syslog messages routed to a channel that is given more bandwidth. Such provisioning is a local matter.

syslog [8] discusses a number of reasons why privacy and authentication of syslog entry messages may be important in a networked computing environment. The nature of BEEP allows for convenient layering of authentication and privacy over any BEEP channel.

4.1. Message Authenticity

Section 8.7 of [1] discusses the dangers of unauthenticated syslog entries. To prevent inauthentic syslog event messages from being accepted, configure syslog peers to require the use of a strong authentication technology for the BEEP session.

If provisioned for message authentication, implementations SHOULD use SASL mechanism DIGEST-MD5 [7] to provision this service.

4.2. Message Replay

Section 8.4 of [1] discusses the dangers of syslog message replay. To prevent syslog event messages from being replayed, configure syslog peers to require the use of a strong authentication technology for the BEEP session.

If provisioned to detect message replay, implementations SHOULD use SASL mechanism DIGEST-MD5 [7] to provision this service.

4.3. Message Integrity

Section 6.5 of [8] discusses the dangers of syslog event messages being maliciously altered by an attacker. To prevent messages from being altered, configure syslog peers to require the use of a strong authentication technology for the BEEP session.

If provisioned to protect message integrity, implementations SHOULD use SASL mechanism DIGEST-MD5 [7] to provision this service.

4.4. Message Observation

Section 6.6 of [8] discusses the dangers (and benefits) of syslog messages being visible at intermediate points along the transmission path between sender and collector. To prevent messages from being viewed by an attacker, configure syslog peers to require the use of a transport security profile for the BEEP session. (However, other traffic characteristics, e.g., volume and timing of transmissions, remain observable.)

If provisioned to secure messages against unauthorized observation, implementations SHOULD use the TLS profile [3] to provision this service. The cipher algorithm used SHOULD be configurable, minimally supporting TLS_RSA_WITH_3DES_EDE_CBC_SHA for backward compatability and TLS_RSA_WITH_AES_256_CBC_SHA for stronger protection. It is expected that new algorithms will need to be added as time passes, in order to prevent compromise. No new revision of this memo should be expected solely for that reason.

4.5. Summary of Recommended Practices

For the indicated protections, implementations SHOULD be configured to use the indicated mechanisms:

Desired Protection	SHOULD tune using
Authentication	http://iana.org/beep/SASL/DIGEST-MD5
+ Replay	http://iana.org/beep/SASL/DIGEST-MD5
+ Integrity	http://iana.org/beep/SASL/DIGEST-MD5
+ Observation	http://iana.org/beep/TLS

BEEP peer identities used for authentication SHOULD correspond to the FQDN of the initiating peer. That is, a relay running on relay.example.com should use a "user ID" of "relay.example.com" within the SASL authentication profiles.

5. Registrations

5.1. Registration: The TARTARE Profile

Profile Identification:
http://xml.resource.org/profiles/syslog/TARTARE

Messages exchanged during Channel Creation: None

Messages starting one-to-one exchanges: Anything

Messages in positive replies: None

Messages in negative replies: None

Messages in one-to-many exchanges: Anything

Message Syntax: See Section 3.3

Message Semantics: See Section 3.4

Contact Information: See the "Authors' Addresses" section of this memo

6. Reply Codes

The following error codes are used in the protocol:

code	meaning
200	success
421	service not available
451	requested action aborted (e.g., local error in processing)
454	temporary authentication failure
500	general syntax error (e.g., poorly-formed XML)
501	syntax error in parameters (e.g., non-valid XML)
504	parameter not implemented
530	authentication required
534	authentication mechanism insufficient (e.g., too weak, sequence exhausted, etc.)
535	authentication failure
537	action not authorized for user
538	authentication mechanism requires encryption
550	requested action not taken (e.g., no requested profiles are acceptable)
553	parameter invalid
554	transaction failed (e.g., policy violation)

7. IANA Considerations

The IANA is requested to note in their registrations that both http://iana.org/beep/SYSLOG/RAW and http://iana.org/beep/SYSLOG/COOKED are deprecated. In addition, the IANA is requested to register the profile listed below.

7.1. Registration: BEEP Profile

The IANA registers the profiles specified in Section 5, and selects IANA-specific URI "http://iana.org/beep/SYSLOG/TARTARE".

7.2. Registration: The System (Well-Known) TCP port number for syslog-conn

A single well-known port (601) is allocated to syslog-conn. In-band negotiation determines which profile to use (either the one defined in this memo or one of the obsolete profiles).

Protocol Number: TCP

Message Formats, Types, Opcodes, and Sequences: See Section 3.3.

Functions: See Section 3.4.

Use of Broadcast/Multicast: none

Proposed Name: Reliable syslog service

Short name: syslog-conn

Contact Information: See the "Authors' Addresses" section of this memo

8. Security Considerations

Consult Section 8 of [1] for a discussion of security issues for the syslog service. In addition, since the TARTARE profile is defined using the BEEP framework, consult [3]'s Section 8 for a discussion of BEEP-specific security issues.

BEEP is used to provide communication security but not object integrity. In other words, the messages "on the wire" can be protected, but a compromised sender may undetectably generate incorrect messages, and relays and collectors can modify, insert, or delete messages undetectably. Other techniques must be used to assure that such compromises are detectable.

9. Acknowledgements

The authors gratefully acknowledge the contributions of Christopher Calabrese, Keith McCloghrie, Balazs Scheidler, and David Waitzman.

10. References

10.1. Normative References

[1] Gerhards, R., "The syslog Protocol", draft-ietf-syslog-protocol-19 (work in progress), November 2006.

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

[3] Rose, M., "The Blocks Extensible Exchange Protocol Core", RFC 3080, March 2001.

[4] Rose, M., "Mapping the BEEP Core onto TCP", RFC 3081, March 2001.

[5] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, February 2000.

[6] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", RFC 2046, November 1996.

[7] Leach, P. and C. Newman, "Using Digest Authentication as a SASL Mechanism", RFC 2831, May 2000.

10.2. Informative References

[8] Lonvick, C., "The BSD Syslog Protocol", RFC 3164, August 2001.

[9] New, D. and M. Rose, "Reliable Delivery for syslog", RFC 3195, November 2001.

Appendix A. Coexistence with old RAW and COOKED modes

This memo specifies a new profile for syslog messages that adhere to the specification in [1]. It is recommended that messages using the older format specified in [8] continue to make use of the deprecated RAW or COOKED profiles specified in [9]. This allows for easy separation of old and new syslog formats.

Appendix B. Changes from RFC 3195

- Deprecated RAW and COOKED profiles.
- Shamelessly copied the RAW profile to a new name and updated it so that there is no length limitation.
- Split references into normative and informative
- In the new model, authors have chosen "sender" instead of "device". We have adopted that language in this draft.
- Updated the language for TLS encryption algorithms to reflect current thinking.
- Use examples from [1]

Authors' Addresses

Darren New
5390 Caminito Exquisito
San Diego, CA 92130
US

Phone: +1 858 350 9733
Email: dnew@san.rr.com

Marshall T. Rose
Dover Beach Consulting, Inc.
POB 255268
Sacramento, CA 95865-5268
US

Phone: +1 916 483 8878
Email: mrose@dbc.mtview.ca.us

Eliot Lear
Cisco Systems
Glatt-com
Glattzentrum, Zurich 8301
CH

Email: lear@cisco.com

Full Copyright Statement

Copyright (C) The IETF Trust (2007).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).