SYSLOG Working Group
Internet-Draft
Expires: January 20, 2007

R. Gerhards
Adiscon GmbH
July 19, 2006

SSH transport mapping for SYSLOG
draft-gerhards-syslog-transport-ssh-00.txt

Status of this Memo

This Internet-Draft will expire on January 20, 2007.

Copyright Notice

Abstract

This document describes a method for invoking and running the SYSLOG protocol within a Secure Shell (SSH) session as an SSH subsystem.

Table of Contents

1. Introduction

The SYSLOG protocol [9] is a text-based protocol used to convey event information. SYSLOG is defined to be session-layer and transport independent, allowing mappings to be defined for multiple session-layer or transport protocols. This document defines how SYSLOG can be used within a Secure Shell (SSH) session, using the SSH connection protocol RFC4254 [8] over the SSH transport protocol RFC4253 [7]. This mapping will allow SYSLOG to be executed from a secure shell session by a user or application. Throughout this document, the terms "client" and "server" are used to refer to the two ends of the SSH transport connection. The client actively opens the SSH connection, and the server passively listens for the incoming SSH connection. The terms "sender" and "receiver" are used to refer to the two ends of the SYSLOG protocol session and are consistent with the definitions in SYSLOG-protocol. When SYSLOG is run over SSH using the mapping defined in this document, the client is always the sender, and the server is always the receiver. This document describes a layered architecture for SYSLOG. The goal of this architecture is to separate message content from message transport while enabling easy extensibility for each layer.

2. Security Requirements for SYSLOG

SYSLOG messages may pass several hops to arrive at the intended receiver. Some intermediary networks may not be trusted by the sender or the receiver or both because the network is in a different security domain or at a different security level from the receiver or sender. Another security concern is that the sender or receiver itself is in an insecure network.

There are several threats to be addressed for SYSLOG security. The primary threats are:

- Masquerade. An unauthorized sender may send messages to a legitimate receiver, or an unauthorized receiver tries to deceive a legitimate sender into sending SYSLOG messages to it.
- Modification. An attacker between the sender and receiver may modify an in-transit SYSLOG message from the sender and then forward the message to receiver. Such modification may make the receiver misunderstands the message or causes the receiver to behave in undesirable ways.
- Disclosure. An unauthorized entity may examine the content of the SYSLOG messages, gaining unauthorized access to the information. Some data in SYSLOG messages is sensitive and may be useful to an attacker, such as the password of an authorized administrator or user.

The secondary threat is:

- o Message stream modification. An attacker may delete a SYSLOG message from a series of messages, replay a message or alter the delivery sequence. SYSLOG protocol itself is not based on message order, but an event in a SYSLOG message may relate semantically to events in other messages, so message ordering may be important to understanding a sequence of events.

The following threats are deemed to be of lesser importance for SYSLOG, and are not addressed in this document:

- o Denial of Service
- o Traffic Analysis

3. 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 [1].

4. Starting SYSLOG over SSH

To run SYSLOG over SSH, the client will first establish an SSH transport connection using the SSH transport protocol, and the client and server will exchange keys for message integrity and encryption. The client will then invoke the "ssh-userauth" service to authenticate the user, as described in the SSH authentication protocol RFC4252 [6]. Once the user has been successfully authenticated, the client will invoke the "ssh-connection" service, also known as the SSH connection protocol.

After the ssh-connection service is established, the client will open a channel of type "session", which will result in an SSH session.

Once the SSH session has been established, the user (or application) will invoke SYSLOG as an SSH subsystem called "syslog". Subsystem support is a feature of SSH version 2 (SSHv2) and is not included in SSHv1. Running SYSLOG as a SSH subsystem avoids the need for the script to recognize shell prompts or skip over extraneous information, such as a system message that is sent at shell start-up. However, if a subsystem cannot be used, it should be possible for a client to skip over any system messages that are sent at shell start-up by searching for a SYSLOG <hello> element. Note that this may not avoid problems if system messages are recieved later in the session.

In order to allow SYSLOG traffic to be easily identified and filtered

by firewalls and other network devices, SYSLOG servers MUST default to providing access to the "syslog" SSH subsystem only when the SSH session is established using the IANA-assigned TCP port <TBD>. Servers SHOULD be configurable to allow access to the syslog SSH subsystem over other ports.

A user (or application), could use the following command line to invoke SYSLOG as an SSH subsystem on the IANA-assigned port:

```
[user@client]$ ssh -s server.example.org -p <TBD> syslog
```

Note that the -s option causes the command ("syslog") to be invoked as an SSH subsystem.

5. Using SYSLOG over SSH

A SYSLOG over SSH session consists of the client sending a continous stream of syslog frames to the receiver. The receiver does not acknowledge frames.

5.1. framing

The SYSLOG frame has the following ABNF [2] definition:

```
  SYSLOG-FRAME = HEADER SP SYSLOG-MSG TRAILER
  HEADER = ENTITY SP FRAME-LEN
  ENTITY = "MSG"
  FRAME-LEN = NONZERO-DIGIT 0*DIGIT
  SP = %d32
  DIGIT = %d48 / NONZERO-DIGIT
  NONZERO-DIGIT = %d49-57
  TRAILER = CRLF
```

Figure 1

SYSLOG-MSG is defined in RFCXXXX [9].

[This text needs to be edited once the specific framing has been selected. This eventually happens in a separate document.]

6. Exiting the SYSLOG Subsystem

Exiting SYSLOG is accomplished using the "CLOSE" operation verb on the frame stream. If the server receivers the "CLOSE" operation, it will return an "ACK" and terminate the connetion.

The server MAY decide to terminate a session at its discretion. In this case, the underlying SSH connection is terminated. No notification other than the SSH error occurs to the server.

[This text needs to be edited once the specific framing has been selected. This eventually happens in a separate document.]

7. Security Considerations

SYSLOG is used to convey potentially sensitive information, so the ability to access this protocol should be limited to users and systems that are authorized to view this information.

The identity of the server MUST be verified and authenticated by the client according to local policy before password-based authentication data or any configuration or state data is sent to or received from the server. The identity of the client MUST also be verified and authenticated by the server according to local policy to ensure that the incoming client request is legitimate before any configuration or state data is sent to or received from the client. Neither side should establish a syslog over SSH connection with an unknown, unexpected or incorrect identity on the opposite side.

SYSLOG messages may include sensitive information, such as usernames or security keys. So, SYSLOG should only be used over communications channels that provide strong encryption for data privacy. This document defines a SYSLOG over SSH mapping which provides for support of strong encryption and authentication.

This document requires that servers default to allowing access to the "syslog" SSH subsystem only when using a specific TCP port assigned by IANA for this purpose. This will allow SYSLOG over SSH traffic to be easily identified and filtered by firewalls and other network nodes. However, it will also allow SYSLOG over SSH traffic to be more easily identified by attackers.

This document also recommends that servers be configurable to allow access to the "syslog" SSH subsystem over other ports. Use of that configuration option without corresponding changes to firewall or network device configuration may unintentionally result in the ability for nodes outside of the firewall or other administrative boundary to gain access to "syslog" SSH subsystem.

8. Authors

The author of this draft is:

Rainer Gerhards
Email: rgerhards@adiscon.com

Phone: +49-9349-92880
Fax: +49-9349-928820

Adiscon GmbH
Mozartstrasse 21
97950 Grossrinderfeld
Germany

9. IANA Considerations

IANA is requested to assign a TCP port number which will be the default port for SYSLOG over SSH sessions as defined in this document.

IANA has assigned port <TBD> for this purpose.

IANA is also requested to assign "syslog" as an SSH Service Name as defined in RFC 4250 [5] as follows:

Service Name	Reference
syslog	[This Document]

10. Acknowledgments

This document was written using the xml2rfc tool described in RFC2629 [4].

The authors wish to thank Chris Lonvick, Anton Okmianski, David Harrington, Tom Petch, and all other people who commented on various versions of this proposal.

11. Notes to the RFC Editor

These are notes to the RFC editor. Please delete this section after the notes have been followed.

Please replace the instances of <TBD> the port number assigned by IANA.

This ID is submitted along with draft-ietf-syslog-protocol. When a RFC number is determined for draft-ietf-syslog-protocol, replace XXXX

in RFCXXXX with the proper RFC number.

12. Normative

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

[2] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, November 1997.

[3] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

[4] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

[5] Lehtinen, S. and C. Lonvick, "The Secure Shell (SSH) Protocol Assigned Numbers", RFC 4250, January 2006.

[6] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Authentication Protocol", RFC 4252, January 2006.

[7] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Transport Layer Protocol", RFC 4253, January 2006.

[8] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Connection Protocol", RFC 4254, January 2006.

[9] Gerhards, R., "The syslog Protocol", draft-ietf-syslog-protocol-17 (work in progress), June 2006.

Author's Address

Rainer Gerhards
Adiscon GmbH
Mozartstrasse 21
Grossrinderfeld, BW 97950
Germany

Email: rgerhards@adiscon.com

Intellectual Property Statement

Disclaimer of Validity

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 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.

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.