SYSLOG Working Group
Internet-Draft
Expires: September 16, 2006

F. Miao
M. Yuzhi
Huawei Technologies

March 15, 2006

TLS Transport Mapping for SYSLOG
draft-miao-syslog-transport-tls-00.txt

Status of this Memo

Copyright Notice

Abstract

This document describes the security threats to Syslog and counter measures of using Transport Layer Security(TLS) protocol for such threats. Different phases are defined for using TLS to secure Syslog, such as initiation, sending data and closure phase.

Table of Contents

1. Terminology

The following definitions are used in this document:

- A sender is an application that can generate and send or forward a Syslog [2] message from an application to another application. Note: the definition of sender is different from syslog-protocol.
- A receiver is an application that can receive a Syslog message.
- A originator is an application that can generate a Syslog message.
- A relay is an application that can receive syslog messages and forward them to another receiver. A relay will be both a sender and receiver.
- A collector is an application that receives messages and does not relay them to any other receiver.
- A TLS client is an application that initiate a TLS connection by sending a Client Hello to a peer.
- A TLS server is an application that receives a Client Hello from a peer and replies with a Server Hello.

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]

2. Security Requirement of 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 misunderstand the message or cause 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 of Syslog message may be trivial for a potential attacker, but some data may be critical to launch an attack, such as the password of an authorized administrator or user.

The secondary threat is:

- Message stream modification. An attacker may delete a Syslog message from a series of messages, replay message or alter the delivery sequence. Syslog protocol itself is not based on flow, but it is possible that an event in a Syslog message semantically relates to other events in other messages.

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

- Denial of Service

- Traffic Analysis

3. Introduction of TLS

3.1. How TLS works

TLS [3] establishes a private end-to-end connection, optionally including strong mutual authentication, using a variety of cryptosystems. Initially, a handshake phase uses three subprotocols to set up a record layer, authenticate endpoints, set parameters, as well as report errors. Then, there is an ongoing layered record protocol that handles encryption, compression, and reassembly for the remainder of the connection. An application data protocol, such as Syslog, is layered on the record protocol.

3.2. Security Properties

TLS record protocol is used to encapsulate various higher level protocols. It provides connection security with confidentiality, integrity, authentication, and replay prevention.

Confidentiality is provided using symmetric cryptography for data encryption. TLS supports both stream cipher and block cipher. The key for encryption is derived from a secret established by the

handshake protocol. The secret is kept private even if there is an eavesdropper in the middle.

Integrity is provided by using HMAC [5] (computed with secure hash function) to check the integrity of a message. Modification without the appropriate key is detectable.

Authentication is provided by a handshake protocol. The peer's identity is authenticated using certificate and signature, based on asymmetric cryptography.

Replay prevention is provided by using a Sequence Number in each TLS record which is used to detect potential delete and replay of a record or alteration of the delivery sequence.

4. TLS to secure Syslog

UDP transport [6] is popular for Syslog, but it does not address security. TLS can be used to counter all the major and secondary threats to Syslog described in section 2:

- Confidentiality to counter disclosure to message
- Integrity check to counter modification to message
- Peer identity authentication to counter masquerade
- Sequence number along with integrity check to counter message stream modification

The security service is also applicable to BSD Syslog defined in RFC3164 [9]. But, it is not ensured that the protocol specification defined in this document applicable to BSD Syslog.

5. Protocol Elements

5.1. protocol Port

A Syslog sender is always a TLS client and a Syslog receiver is always a TLS server. Similiar to RFC2818 [8], a special listening port is allocated for Syslog over TLS. A Syslog receiver with TLS transport listens on TCP port NNN, which will be IANA-assigned.

[Issue 0]: Do we need a Syslog TCP port for TLS transport? The security community had debates about whether using special ports is desirable.

5.2. Initiation

The sender should initiate a connection to the receiver and then send the TLS Client Hello to begin the TLS handshake. When the TLS handshake has finished the Sender may then send the first Syslog message.

TLS uses certificate [4] to authenticate the peers. When sender authenticates a receiver it MUST check the common name(CN) of the certificate against the host name of the receiver. If the common name does not match the host name, the sender MUST send an "access_denied" error alert with TLS alert protocol to terminate handshake, and then close the connection.

When a receiver authenticates a sender, the common name of the certificate SHOULD be checked. If the certificate is not a generic certificate and the common name does not match the host name, the receiver MAY send an "access_denied" error alert with TLS alert protocol to terminate handshake, and then close the connection. If the certificate is a generic certificate, the check MUST be executed when processing a Syslog message. If the APP-NAME of a Syslog message does not match the name of the common name of the sender's certificate, the receiver MAY send an "access_denied" error alert with TLS alert protocol and close the on-going connection.

[Issue 1]: Is it possible to use "generic certificate for different host? The generic certificate is for specific application type.

[Issue 2]: What to bind to a certificate? Hostname, Syslog APP-NAME(generic certificate)? APP-NAME binding makes authentication/ access control happens both in TLS handshake and Syslog message processing, is efficiency a problem?

An administrator should decide what security level (e.g. cryptographic algorithms and length of keys) is required. It is local policy and up to administrator's decision. Syslog applications should be implemented in a manner that permits administrators to select the cryptographic level they desire.

An earlier TLS session or another active session MAY be resumed to save the effort of TLS handshake. The security parameters of a resumed session are reused for the current session. The certificate MUST be checked when resuming a session. If the resumed session and current session use different certificates, resumption MUST not happen.

5.3. Sending data

All Syslog messages MUST be sent as TLS "application data". There MAY be multiple Syslog message in same TLS record. At the end of each Syslog message, there MUST be CR LF control characters to indicate the termination of a Syslog message. The last Syslog message in a TLS record MUST NOT end with CR LF termination.

[Issue 3] The problem of CR LF is it can not process binary data well. How to process Syslog signature/certificate message?

5.4. Closure

A sender MUST close a connection if it is not using the connection. It MUST send a TLS closure_notify alert before closing the connection. A sender MAY choose not to wait for the receiver's closure_notify alert and simply close the connection, thus generating an incomplete close on the receiver side. Once the receiver gets closure_notify from the sender, it MUST reply with a closure_notify unless it becomes aware of the connection is already closed by sender (e.g. indicated by TCP).

When there are no data received from a connection for a long time (it is up to the application to decide what "long" means), a receiver MAY close a connection. The receiver MUST attempt to initiate an exchange of closure_notify alerts with the sender before closing the connection. Receivers that are unprepared to receive any more data MAY close the connection after sending the closure_notify alert, thus generating an incomplete close on the sender side. When the sender has received the closure_notify alert from the receiver and still has pending data to send, sender SHOULD send the pending data before sending closure_notify alert.

6. Security Consideration

6.1. TLS and Syslog Signature

TLS transport and Syslog signature[7] address quite different security requirements. Basically Syslog signature is between an originator and a collector. Contrastively TLS transport is between sender and receiver. The Peer identity authentication of TLS checks whether the data is received from a legitimate Syslog peer (message originator or relay), but Syslog signature checks whether the data generated by a specific originator. It is possible that administrator to enable both TLS and signature to meet specific requirement.

6.2. Authentication

TLS authentication and secret establishing is based on certificates and asymmetric cryptography, and it makes TLS transport is much more costly than UDP transport. An attacker may initialize and keep a lot of TLS connection to the receiver to launch a denial of service attack. A receiver SHOULD authenticate the identity of a sender to mitigate such attack.

A sender MAY authenticate the identity of a receiver. When confidentiality is a concern and data encryption is chosen, the receiver MUST be authenticated by the Sender to make sure it is talking to the right peer.

[Issue 4]: Shall we mandate the sender MUST be authenticated? Most of the Syslogd accepts messages only from configured address.

6.3. TLS Session Resumption

Different applications in same host may have different security level (e.g. kernel may have higher security level than a document editor). The application can decrypt the Syslog messages of a resuming or resumed session with same cipher parameters. When a session is being resumed from an application in a different security level care must be taken to avoid sensitive data is disclosed to unauthorized application. A sensitive session must not be resumable.

7. Acknowledgments

Authors appreciate Anton Okmianski for his ideas on certificate, and Balazs Scheidler, Tom Petch and other persons because of their input on security threats of Syslog. The author would like to acknowledge David Harrington for his detailed reviews of the content and grammar of the document.

8. References

8.1. Normative References

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

[2] Gerhards, R., "The syslog Protocol", draft-ietf-syslog-protocol-16 (work in progress), January 2006.

[3] Dierks, T. and C. Allen, "The TLS Protocol Version 1.0",

RFC 2246, January 1999.

[4] Housley, R., Polk, W., Ford, W., and D. Solo, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3280, April 2002.

[5] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997.

8.2. Informative References

[6] Okmianski, A., "Transmission of syslog messages over UDP", draft-ietf-syslog-transport-udp-06 (work in progress), November 2005.

[7] Kelsey, J., "Signed syslog Messages", draft-ietf-syslog-sign-17 (work in progress), November 2005.

[8] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000.

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

Authors' Addresses

Fuyou Miao
Huawei Technologies
No. 3, Xinxi Rd
Shangdi Information Industry Base
Haidian District, Beijing 100085
P. R. China

Phone: +86 10 8283 6032
Email: miaofy@huawei.com
URI: www.huawei.com

Yuzhi Ma
Huawei Technologies
No. 3, Xinxi Rd
Shangdi Information Industry Base
Haidian District, Beijing 100085
P. R. China

Phone: +86 10 8283 6033
Email: myz@huawei.com
URI: www.huawei.com

Intellectual Property Statement

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.

Disclaimer of Validity

Copyright Statement

Acknowledgment

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