I2NSF
Internet-Draft
Intended status: Standards Track
Expires: January 19, 2018

S. Hares
Hickory Hill Consulting
R. Moskowitz
HTT Consulting

July 18, 2017

Secure Session Layer Services
draft-hares-i2nsf-ssls-02.txt

Abstract

Each I2NSF agent and I2NSF client needs to provide application level support for management traffic during periods of DDoS and network security attacks to deal with congestion (burst and/or continuous), high error rates and packet loss due to the attacks, and the inability to utilize a transport protocol (E.g. TCP) due to a specific protocol attack. This application level support needs to be able to select the key management system and provide "chunking" of data (in order to fit in reduced effective MTUs), compression of data (in order to fit into reduced bandwidth), small security envelope)in order to maximize room for management payload), and fragmentation and reassembly at the application layer for those protocols which do not support fragmentation/reassembly (E.g. UDP or SMS).

These Secure Session Layer services may only be deployed on a the few management ports which need to be protected during DDoS attacks or network security attacks, and turned on/off based on need. The application and the network instrumentation need to cooperate to determine if this service needs to be turned on or off. This draft specifies a security session layer services(SSLs) which provide these features in terms of APIs (North-Bound and South-bound), and the component features (interface to key management systems, data compression, chunking of data, secure session envelope (SSE) to send data, and fragmentation and reassembly, and ability to detect existence of attack).

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Table of Contents

1. Introduction 2
2. SSLS Processes 4
2.1. Chunking of Data 4
2.2. Secure Session Envelope 5
2.3. Application Packet Fragmentation and Reassembly 5
2.4. Proprietary Plugins: Detect Conditions + Select Transport 5
3. IANA Considerations 5
4. Security Considerations 6
5. Acknowledgements 6
6. References 6
6.1. Normative References 6
6.2. Informative References 7
Authors' Addresses 7

1. Introduction

Each I2NSF agent and I2NSF client needs to provide application level support for management traffic during periods of DDoS and network security attacks to deal with congestion (burst and/or continuous), high error rates and packet loss due to the attacks, and the inability to utilize a transport protocol (E.g. TCP) due to a specific protocol attack. Some of the services the I2NSF controller must provide during these periods of DDoS or network security attacks are:

- receiving information regarding DDoS Threats from DOTS systems,
- Changing policy on vNSF and NSF devices during these periods,
- exchanging information with user security applications using I2NSF to obtain information from the controller,
- Aid the I2NSF reporting of attacks with the the CERT (MILE) either by providing data or sendign the report
- and manages network connnectivity of devices out of compliance (SACM).

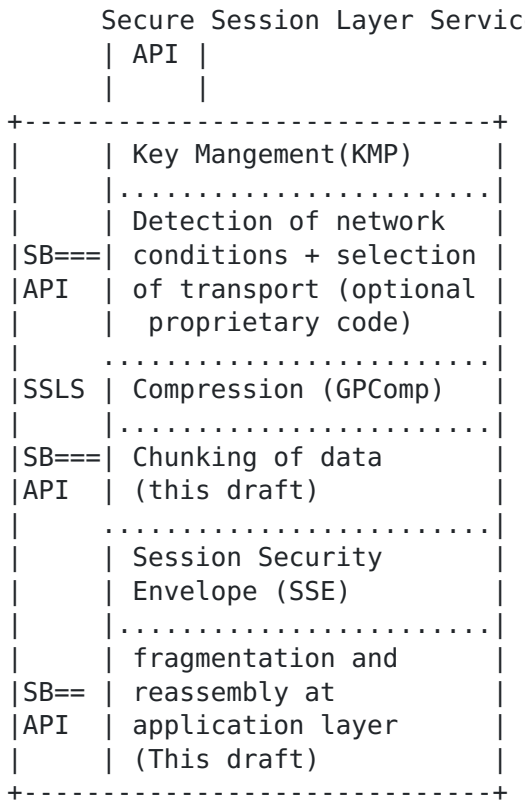
This application level support for I2NSF client-agent communication needs to be able to select the key management system and provide "chunking" of data (in order to fit in reduced effective MTUs), compression of data (in order to fit into reduced bandwidth), small security envelope)in order to maximize room for mangement payload), and fragmentation and reassembly at the application layer for those protocols which do not support fragmentation/reassembly (E.g. UDP or SMS). The application layer needs to be able to turn off this features if the system detects these features are no longer needed.

These requirements can be well met with the Secure Session Layer Service [draft-hares-ssls-00]:

- A North-bound API from the application to the session layer
- A South-bound API from the session layer to the network layer
- interface to key management system,
- data compression
- chunking of data
- secure envelope,
- fragmentation and reassembly,
- detection of network conditions that require this service.

A diagram of the SSLS with these process is in figure 1.

The API for this SSLS allows the application to select the types of key management, and the different types of services (data compression, chunking of data, secure e)



2. SSLS Processes

2.1. Chunking of Data

The process that "chunks" data breaks down the application stream after the compression process. If the compression process has compressed the data, the chunking process will chunk compressed data. If the user has requested no compression, this chunking process will chunk uncompressed data.

The secure session envelope must be bigger than the chunk.

If the SSE is using TCP or STCP, that assembles the application flow into a byte stream, then the SSE packages will contain a chunk within the secure session envelope.

If Transports that do not fragment and re-assembly are being specified, the SSL will support application layer fragmentation and reassembly. (see the fragmentation section below).

2.2. Secure Session Envelope

The Secure Session Envelope (SSE) [I-D.moskowitz-sse] creates a secure envelope using the SPI created by the key management and running over the transport selected by the user.

2.3. Application Packet Fragmentation and Reassembly

SSE's secure envelope may be passed over UDP to avoid transport-level security attacks. Alternatively SSE's secure transport may go over the extremely limited SMS fabric so that some security management information gets through. In both cases, the user (or the "detection log") can select the transport and fragmentation.

If fragmentation is turned on, the individual SSE envelopes will track the IP messages the SSE envelope is broken into. The SSE process receiving the traffic will send back an acknowledge SSE packet. It is anticipate that the fragmentation process will attempt to bundle some acks.

2.4. Proprietary Plugins: Detect Conditions + Select Transport

The SSL process allows two proprietary plugins:

1. Plugin to detect error conditions which require SSLS services which include:

    * High levels of end-to-end congestion,

    * High levels of error and loss,

    * Input from IDS/IPS that detects problems

    * Signals from other I2NSF applications

2. Proprietary actions may select transport based on input from other standardize security services (DOTS, CERT, MILE) or proprietary services.

Prototype code will provide instances to show plugin values, and the South-Bound API to these plugins.

3. IANA Considerations

TBD

4. Security Considerations

The SSLS shares the following security considerations with the SSE Technology:

- As SSE uses an AEAD block cipher, it is vulnerable to attack if a sequence number is reused for a given key. Thus implementations of SSE MUST provide for rekeying prior to Sequence Number rollover. An implementation should never assume that for a given context, the sequence number space will never be exhausted. Key Management Protocols like IKEv2 [RFC7296] or HIP [RFC7401] could be used to provide for rekeying management. The KMP SHOULD not create a network layer fate-sharing limitation.

- As any security protocol can be used for a resource exhaustion attack, implementations should consider methods to mitigate flooding attacks of messages with valid SPIs but invalid content. Even with the ICV check, resources are still consumed to validate the ICV.

- SSE makes no attempt to recommend the ICV length. For constrained network implementations, other sources should guide the implementation as to ICV length selection. The ICV length selection SHOULD be the the responsibility of the KMP.

- As with any layered security protocol, SSE makes no claims of protecting lower or higher processes in the communication stack. Each layer's risks and liabilities need be addressed at that level.

5. Acknowledgements

The authos would like to thank Frank (Liang) Xia for his comments and suggestions on this draft.

6. References

6.1. 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>.

6.2. Informative References

[I-D.moskowitz-sse]
Moskowitz, R., Faynberg, I., Lu, H., Hares, S., and P. Giacomin, "Session Security Envelope", draft-moskowitz-sse-05 (work in progress), June 2017.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <http://www.rfc-editor.org/info/rfc7296>.

[RFC7401] Moskowitz, R., Ed., Heer, T., Jokela, P., and T. Henderson, "Host Identity Protocol Version 2 (HIPv2)", RFC 7401, DOI 10.17487/RFC7401, April 2015, <http://www.rfc-editor.org/info/rfc7401>.

Authors' Addresses

Susan Hares
Hickory Hill Consulting
Saline
US

Email: shares@ndzh.com

Robert Moskowitz
HTT Consulting
Oak Park, MI 48237
USA

Phone: +1-248-968-9809
Email: rgm@htt-consult.com