ISMS J. Schoenwaelder, Ed.
Internet-Draft Jacobs University Bremen
Intended status: Informational R. Story
Expires: October 6, 2011 SPARTA/Cobham
M. Vrecko
MG-SOFT
April 4, 2011

Interoperability Report for RFC 5343, RFC 5590, RFC 5591, and RFC 5953
draft-schoenw-isms-interoperability-report-01

Abstract

This document provides the interoperability report for RFC 5343, RFC 5590, RFC 5591, and RFC 5953.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction 3
2. RFC 5343 Report (Net-SNMP - SNMP Research) 3
3. RFC 5343 Report (MG-SOFT - Net-SNMP / SNMP Research) 4
4. RFC 5590 Report (Net-SNMP - SNMP Research) 5
5. RFC 5590 Report (MG-SOFT - Net-SNMP / SNMP Research) 9
6. RFC 5591 Report (Net-SNMP - SNMP Research) 11
7. RFC 5591 Report (MG-SOFT - Net-SNMP / SNMP Research) 13
8. RFC 5953 Report (Net-SNMP - SNMP Research) 15
9. RFC 5953 Report (MG-SOFT - Net-SNMP / SNMP Research) 16
10. Security Considerations 19
11. IANA Considerations 19
12. Informative References 19
Authors' Addresses 20

1. Introduction

This document provides the interoperability report for SNMP Context EngineID Discovery [RFC5343], the Transport Subsystem for SNMP [RFC5590], the Transport Security Model for SNMP [RFC5591], and the Transport Layer Security (TLS) Transport Model for SNMP [RFC5953].

2. RFC 5343 Report (Net-SNMP - SNMP Research)

Summary

Two independent implementations of SNMP Context EngineID Discovery have been developed, tested, and found to be interoperable. The developers of both implementation agree that RFC 5343 is sufficiently clear to allow for interoperable implementations.

The two implementations which have been tested for interoperability are Net-SNMP release 5.6 and SNMP Research DR-Web EMANATE/Lite Agent Version 17.1.1.3.

Methodology

Each implementation provided remote access to running command responders and tested the other implementation using their own command generators. Packet captures were used to verify data sent/received on the wire.

Exceptions

The list of untestable requirements are listed below in this document. Initially one implementation was erroneously performing discovery for all PDUs, including traps. This was quickly fixed when discovered.

Testable Requirements

There were no testable requirements, as all requirements were internal implementation details.

Packet sniffing was use to determine that implementations were sending the correct localEngineID during discovery.

Untestable Requirements

3.1. Local EngineID

An SNMP command responder implementing this specification MUST register their pduTypes using the localEngineID snmpEngineID value (defined below) by invoking the registerContextEngineID() Abstract Service Interface (ASI) defined in RFC 3412 [RFC3412].

Note that the localEngineID value is intended to be used as a special value for the contextEngineID field in the ScopedPDU. It MUST NOT be used as a value to identify an SNMP engine; that is, this value MUST NOT be used in the snmpEngineID.0 scalar [RFC3418] or in the msgAuthoritativeEngineID field in the securityParameters of the User-based Security Model (USM) [RFC3414].

3.2. EngineID Discovery

Discovery of the snmpEngineID is done by sending a Read Class protocol operation (see Section 2.8 of [RFC3411]) to retrieve the snmpEngineID scalar using the localEngineID defined above as a contextEngineID value. Implementations SHOULD only perform this discovery step when it is needed.

3. RFC 5343 Report (MG-SOFT - Net-SNMP / SNMP Research)

Summary

MG-SOFT's SNMP management utility built on top of MG-SOFT's WinSNMP API version 8.0.500 (www.mg-soft.com), acting as a command generator application, has been successfully tested over the Internet against two other command responder applications:

1. Net-SNMP release 5.6 (www.net-snmp.org)
2. SMMP Research agent (www.snmp.com)

With both of these two independent implementations we have successfully utilized the Context EngineID discovery mechanism as defined in RFC 5343 and successfully passed the interoperability tests. Both TLS and DTLS transport domains have been tested. The SNMP Get and Get-Next operations have been tested.

MG-SOFT provided for interoperability testing purposes a publicly accessible SNMP agent that acts as command responder application. So far, MG-SOFT's SNMP agent supporting SNMP over TLS/DTLS has been successfully tested both by Net-SNMP release 5.6 tools and SNMP Research's tools. Both TLS and DTLS domains

have been tested successfully from both independent implementations. The Context EngineID discovery mechanism has been successfully utilized when (D)TLS sessions have been successfully established.

In all tests the authPriv session has been successfully negotiated. MG-SOFT's implementation does not implement the optional mapping between TLS algorithms and SNMP security levels.

In all cases, testing has been performed with 'interoperability test' command-generator.crt and command-receiver.crt X.509 certificates as they were generated and prepared by the Net-SNMP team.

MG-SOFT's WinSNMP API is utilizing the most recent openSSL library (as of these tests, version 1.0.0d) for supporting the underlying TLS and DTLS functionality.

MG-SOFT's developers believe that RFC 5343 is clear and exact enough to allow a successful implementation.

Tested Requirements

- 3.1 Local EngineID

  Usage of Local Engine ID has been successfully tested. Command generator application successfully read snmpEngineID.0 by using the Local Engine ID.

- 3.2 EngineID Discovery

  The EngineID Discovery procedure has successfully been tested.

4. RFC 5590 Report (Net-SNMP - SNMP Research)

Summary

Two independent implementations of the Transport Subsystem have been developed, tested, and found to be interoperable. The developers of both implementation agree that RFC 5590 is sufficiently clear to allow for interoperable implementations.

The two implementations which have been tested for interoperability are Net-SNMP release 5.6 and SNMP Research EMANATE/Lite Agent Version 17.1.1.3.

Methodology

As the Transport Subsystem is a framework on top of which new transports can be defined, interoperability cannot be tested directly. For this report, the Transport Subsystem interoperability was tested during the interoperability testing for the TLS security model defined in RFC 5953, for which a separate interoperability report was submitted.

Exceptions

Most of the requirements in 5590 are requirements for future transport protocols, and as such are not testable. The list of untestable requirements is provided below as well.

Tested Requirements

- 3.3.4. Message Security versus Session Security

  A Transport Model MAY upgrade the security level requested by a transport-aware Security Model, i.e., noAuthNoPriv and authNoPriv might be sent over an authenticated and encrypted session.

  To test this requirement a client established an authPriv session and sent an authNoPriv message.

- 3.1.1. Security Protocol Requirements

  Since multiple Transport Models can exist simultaneously within the Transport Subsystem, Transport Models MUST be able to coexist with each other.

  Net-SNMP has implemented both the DTLS and SSH transports, with no conflicts.

Untestable Requirements

- 3.1 Message Security Requirements

  Transport security protocols SHOULD provide protection against the following message-oriented threats:

  1. modification of information
  2. masquerade
  3. message stream modification
  4. disclosure

- 3.1.1. Security Protocol Requirements

  A Transport Model SHOULD NOT require modifications to the underlying protocol. Modifying the protocol might change its security characteristics in ways that could impact other existing usages. If a change is necessary, the change SHOULD be an extension that has no impact on the existing usages.

  Since multiple Transport Models can exist simultaneously within the Transport Subsystem, Transport Models MUST be able to coexist with each other.

- 3.2.2.1. securityName and securityLevel Mapping

  Documents defining a new transport domain MUST define a prefix that MAY be prepended to all securityNames passed by the Security Model. The prefix MUST include one to four US-ASCII alpha-numeric characters, not including a ":" (US-ASCII 0x3a) character.

- 3.3.3. Session Maintenance Requirements

  If a Transport Model defines MIB module objects to maintain session state information, then the Transport Model MUST define what happens to the objects when a related session is torn down, since this will impact the interoperability of the MIB module.

- 3.3.4. Message Security versus Session Security

  Cryptographic keys associated with the transport session SHOULD be used to provide authentication, integrity checking, and encryption services, as needed, for data that is communicated during the session. The cryptographic protocols used to establish keys for a Transport Model session SHOULD ensure that fresh new session keys are generated for each session.

- 3.3.4. Message Security versus Session Security

  A Transport Model MUST NOT downgrade the security level requested by a transport-aware Security Model, and SHOULD discard any message where this would occur.

- 5.2. tmStateReference

  For architectural modularity between Transport Models and transport-aware Security Models, a fully-defined tmState MUST conceptually include at least the following fields:

tmTransportDomain
tmTransportAddress
tmSecurityName
tmRequestedSecurityLevel
tmTransportSecurityLevel
tmSameSecurity
tmSessionID

- 5.2.4. Session Information

  For security reasons, if a secure transport session is closed between the time a request message is received and the corresponding response message is sent, then the response message SHOULD be discarded, even if a new session has been established.

  o tmSameSecurity: this flag is used by a transport-aware Security Model to indicate whether the Transport Model MUST enforce this restriction.

  o tmSessionID: in order to verify whether the session has changed, the Transport Model must be able to compare the session used to receive the original request with the one to be used to send the response

  When processing an outgoing message, if tmSameSecurity is true, then the tmSessionID MUST match the current transport session; otherwise, the message MUST be discarded and the Dispatcher notified that sending the message failed.

- 7. Security Considerations

  Since the cache will contain security-related parameters, implementers SHOULD store this information (in memory or in persistent storage) in a manner to protect it from unauthorized disclosure and/or modification.

- 7.1. Coexistence, Security Parameters, and Access Control

  o For outgoing messages, if a Secure Transport Model is selected in combination with a Security Model that does not populate a tmStateReference, the Secure Transport Model SHOULD detect the lack of a valid tmStateReference and fail.

5. RFC 5590 Report (MG-SOFT - Net-SNMP / SNMP Research)

Summary

MG-SOFT's SNMP management utility built on top of MG-SOFT's WinSNMP API version 8.0.500 (www.mg-soft.com), acting as a command generator application, has been successfully tested over the Internet against two other command responder applications:

1. Net-SNMP release 5.6 (www.net-snmp.org)
2. SNMP Research agent (www.snmp.com)

With both of these two independent implementations we have successfully passed the interoperability tests.

MG-SOFT provided for interoperability testing purposes a publicly accessible SNMP agent that acts as command responder application. So far, the MG-SOFT's SNMP agent supporting SNMP over TLS/DTLS has been successfully tested both by Net-SNMP release 5.6 tools and SNMP Research's tools. Both TLS and DTLS domains have been tested successfully from both independent implementations.

In all cases, testing has been performed with 'interoperability test' command-generator.crt and command-receiver.crt X.509 certificates as they were generated and prepared by the Net-SNMP team.

MG-SOFT's WinSNMP API is utilizing the most recent openSSL library (as of these tests, version 1.0.0d) for supporting the underlying TLS and DTLS functionality.

RFC 5590 defines a transport subsystem that extends Simple Network Management Protocol (SNMP) architecture defined in RFC 3411. As RFC 5590 defines a framework for the coexistence of multiple different transport models and MG-SOFT's WinSNMP API version 8.0.500 implements only the Transport Layer Security (TLS) Transport Model defined in RFC 5953, the requirements defined in RFC 5590 could not be tested directly. The interoperability of the framework defined in RFC 5590 has been confirmed indirectly while testing interoperability of the RFC 5953.

MG-SOFT's developers believe that RFC 5590 is clear and exact enough to allow a successful implementation.

Tested Requirements

- 3.3.4. Message Security versus Session Security

  A Transport Model MAY upgrade the security level requested by a transport-aware Security Model, i.e., noAuthNoPriv and authNoPriv might be sent over an authenticated and encrypted session.

  MG-SOFT command generator application sends noAuthNoPriv message for ContextEngineId discovery over previously established authPriv session.

Untested Requirements

- 3.1 Message security requirements

  Protection against message-oriented threads: modification of information, masquerade, message stream modification and disclosure have not been tested.

- 3.1.1 Security protocol requirements

  As MG-SOFT has implemented only the TLS transport model, the coexistence of multiple transport models could not be tested.

- 3.2.1 Architectural Modularity Requirements

  These requirements could not be tested directly. However, MG-SOFT followed these requirements when extending MG-SOFT WinSNMP API.

- 3.2.2 Access Control Requirements

  Access control requirements have not been tested.

- 3.3.1 No SNMP Session

  Maintenance of multiple transport sessions has not been tested.

- 3.3.2 Session Establishment Requirements

  These requirements have not been tested directly.

- 3.3.3. Session Maintenance Requirements

  Session maintenance requirements have not been tested.

- 3.3.4. Message Security versus Session Security

  These requirements have not been completely tested.

- 5.2 tmStateReference

  These requirements have not been tested directly.

- 5.2.4 Session Information

  These requirements have not been tested.

- 7 Security Considerations

  These requirements have not been tested.

- 7 Coexistence, Security Parameters, and Access Control

  These requirements have not been tested.

6. RFC 5591 Report (Net-SNMP - SNMP Research)

Summary

Two independent implementations of the Transport Security Model (TSM) have been developed, tested, and found to be interoperable. The developers of both implementation agree that RFC 5591 is sufficiently clear to allow for interoperable implementations.

The two implementations which have been tested for interoperability are Net-SNMP release 5.6 and SNMP Research DR-Web EMANATE/Lite Agent Version 17.1.1.3.

Methodology

As the TSM is a framework security model to be used with other secure transports, interoperability cannot be tested directly. For this report, TSM interoperability was tested during the interoperability testing for the TLS security model defined in RFC 5953.

Exceptions

The list of untestable requirements are listed below in this document.

Initially one implementation was erroneously setting the security level for response packets to match the security level asserted by the transport layer. This caused the other implementation to

drop the response when it was received. The ASI in section 4.1.2, Sending a Response to the Network, has a comment associated with the securityLevel passed to returnResponsePdu which indicates that the value should match the value from the incoming packet. This is consistent with how the SNMPv3 standard specifies handling of the securityLevel, thus the implementation was in error.

Testable Requirements

- 1.1 Mandatory MIB objects

```
snmpTsmCompliance MODULE-COMPLIANCE
     MANDATORY-GROUPS { snmpTsmGroup }
snmpTsmGroup OBJECT-GROUP
     snmpTsmInvalidCaches,
     snmpTsmInadequateSecurityLevels,
     snmpTsmUnknownPrefixes,
     snmpTsmInvalidPrefixes,
     snmpTsmConfigurationUsePrefix
```

Client side tests

o verify each object can be queried

o verify that snmpTsmConfigurationUsePrefix is writable

Exceptions

o Both existing implementations of RFC 5953 chose to always negotiate authPriv sessions and did not implement the optional mapping of TLS algorithms to SNMP security levels. This made it impossible to send an authPriv message over a transport with an inadequate security level. Net-SNMP plans on implementing mapping in a future release, and SNMP Research has indicated that it will implement it given sufficient customer demand.

Untestable Requirements

- 3.1.2. tmStateReference

For the Transport Security Model, the security parameters used for a response MUST be the same as those used for the corresponding request.

- 3.1.3. Prefixes and securityNames

If snmpTsmConfigurationUsePrefix is set to true, then all securityNames provided by, or provided to, the Transport Security Model MUST include a valid transport domain prefix.

If snmpTsmConfigurationUsePrefix is set to false, then all securityNames provided by, or provided to, the Transport Security Model MUST NOT include a transport domain prefix.

- 8. Security Considerations

This Security Model SHOULD always be used with Transport Models that provide adequate security, but "adequate security" is a configuration and/or run-time decision of the operator or management application.

7. RFC 5591 Report (MG-SOFT - Net-SNMP / SNMP Research)

Summary

MG-SOFT's SNMP management utility built on top of MG-SOFT's WinSNMP API version 8.0.500 (www.mg-soft.com), acting as a command generator application, has been successfully tested over the Internet against two other command responder applications:

1. Net-SNMP release 5.6 (www.net-snmp.org)
2. SNMP Research agent (www.snmp.com)

With both of these two independent implementations we have successfully passed the interoperability tests. Both the TLS and the DTLS transport domain have been tested. The SNMP Get and Get-Next operations have been tested.

MG-SOFT provided for interoperablility testing purposes a publicly accessible SNMP agent that acts as command responder application. So far, the MG-SOFT's SNMP agent supporting SNMP over TLS/DTLS has been sucesfully tested both by Net-SNMP release 5.6 tools and SNMP Research's tools. Both TLS and DTLS domains have been tested successfully from both independent implementations.

In all cases, testing has been performed with 'interoperability test' command-generator.crt and command-receiver.crt X.509 certificates as they were generated and prepared by the Net-SNMP team.

MG-SOFT's implementation does not implement the optional mapping between TLS algorithms and SNMP security levels.

MG-SOFT's WinSNMP API is utilizing the most recent openSSL library (as of these tests, version 1.0.0d) for supporting the

underlying TLS and DTLS functionality.

MG-SOFT's developers believe that RFC 5591 is clear and exact enough to allow a successful implementation.

Tested Requirements

- 2.3.1 Coexistence with Message Processing Models

  Coexistence with SNMPv1 and SNMPv2c message processing models has been successfully tested in the command generator role. The MG-SOFT SNMP management utility application has been successfully performing SNMP operation against different SNMP agents by using SNMPv1, SNMPv2c and SNMPv3-USM over unencrypted UDP (SNMP agents in MG-SOFT lab) and SNMPv3-TSM over TSLTM (Net-SNMP's and SNMP Research's publicly accessible test SNMP agent).

- 2.3.2 Coexistence with Other Security Models

  Coexistence with the SNMPv3-USM security model has been successfully tested in the command generator role. The MG-SOFT SNMP management utility application has been successfully performing SNMP operation against different SNMP agents by using SNMPv3-USM over unencrypted UDP (SNMP agents in MG-SOFT lab) and SNMPv3-TSM over TSLTM (Net-SNMP's and SNMP Research's publicly accessible test SNMP agent).

- 8. Security Considerations

  Usage of TSM without TLSTM is disabled in MG-SOFT's WinSNMP API, so it can not be used with a transport model without adequate security.

Untested Requirements

- 2.3.3 Coexistence with Transport Models

  Coexistence with transport models has not been tested.

- 3.1.3 Prefixes and securityNames

  Usage of SNMP transport domain prefixes and the configuration of its usage in the SNMP-TSM-MIB have not been tested.

- 6. MIB Module Overview

  The implementation of SNMP-TSM-MIB has not been tested.

8. RFC 5953 Report (Net-SNMP - SNMP Research)

Summary

Two independent implementations of the Transport Layer Security (TLS) Transport Model been developed, tested, and found to be interoperable. The developers of both implementation agree that RFC 5953 is sufficiently clear to allow for interoperable implementations.

The two implementations which have been tested for interoperability are Net-SNMP version 5.6 and SNMP Research EMANATE/Lite Agent Version 17.1.1.3. Although the SNMP code for each is independent, both use the (D)TLS libraries from OpenSSL. However, each used a different approach for using the (D)TLS API.

The Net-SNMP project has deployed a publicly available test server to allow for continued interoperability testing with new or existing implementations.

Methodology

Each implementation provided remote access to running command responders and trap receivers, and tested the other implementation using their own command generators. In addition to basic object comparisons, stimulus/response testing was conducted.

Exceptions

Both existing implementations of RFC 5953 chose to always negotiate authPriv sessions and did not implement the optional mapping of TLS algorithms to SNMP security levels. This made it impossible to test sending an authPriv message over a transport with an inadequate security level. (Net-SNMP plans to add security level mapping in a future release, and SNMP Research indicates that they will implement the feature if there is sufficient customer demand.)

Implementations that do choose to implement mapping of TLS algorithms to SNMP security levels should provide clear documentation to their users about the implications of mapping algorithms to security levels other than authPriv. Consider the following scenario: Client A maps MD5/RC4 to authPriv and negotiates a TLS session with Agent B, who maps md5/rc4 to authNoPriv. Packets from Client A that are marked authPriv will

be silently dropped, even though (D)TLS negotiations succeeded.

Details

The short version, for the impatient, is that "it works." Basic interoperability between the Net-SNMP and SNMP Research implementations has been demonstrated for all the core protocol operations (e.g. Get, Get-Next, Set, Trap, Inform).

Neither implementation claims to be a complete, bug-free production ready implementation, and occasional differences have been found noted between the implementations. To date, however, all the differences have fallen into one of these categories:

- object not implemented yet
- corner cases not handled yet
- code needs to be refactored to meet requirement

In other words, so far all issues are with a particular implementation, not with the specification.

Testing has been performed for various certificate configurations, include self-signed certificate and certificates signed by a trusted certificate authority.

Security name mappings have been made by directly specifying the security name for a certificate, and by mapping the common name or subject alt names (including email addresses, dns addresses and IP addresses).

It may be helpful to add text clarifying that the security level associated with a (D)TLS session is only used for ensuring that a session has sufficient security for a packet. The security level in outgoing/incoming packets continue to function per the SNMPv3 standard. In other words, the security level in outgoing packets is not modified to match the security level of the session, and response packets copy the security level from the original packet.

9. RFC 5953 Report (MG-SOFT - Net-SNMP / SNMP Research)

Summary

MG-SOFT's SNMP management utility built on top of MG-SOFT's WinSNMP API version 8.0.500 (www.mg-soft.com), acting as a command generator application, has been successfully tested over

the Internet against two other command responder applications:

1. Net-SNMP release 5.6 (www.net-snmp.org)
2. SNMP Research agent (www.snmp.com)

With both of these two independent implementations we have successfully communicated using TLSTM and so passed the basic interoperability tests. Both TLS and DTLS transport domain have been been tested. The SNMP Get and Get-Next operations have been tested. In all tests an authPriv session has been negotiated. MG-SOFT's implementation does not implement the optional mapping between TLS algorithms and SNMP security levels.

MG-SOFT provided for interoperability testing purposes a publicly accessible SNMP agent that acts as command responder application. So far, MG-SOFT's SNMP agent supporting SNMP over TLS/DTLS has been successfully tested both by Net-SNMP release 5.6 tools and SNMP Research's tools. Both TLS and DTLS domains have been tested successfully from both independent implementations.

In all cases, testing has been performed with 'interoperability test' command-generator.crt and command-receiver.crt X.509 certificates as they were generated and prepared by the Net-SNMP team.

MG-SOFT's WinSNMP API is utilizing the most recent openSSL library (as of these tests, version 1.0.0d) for supporting the underlying TLS and DTLS functionality.

MG-SOFT's developers believe that RFC 5953 is clear and exact enough to allow a successful implementation.

Tested Requirements

- 3.1.2 Message Protection

  In all tests the authPriv session has been negotiated. MG-SOFT's implementation does not implement the optional mapping of TLS security algorithms to SNMP security levels.

- 3.1.3 (D)TLS Connections

  MG-SOFT implementation opens a (D)TLS connection when an SNMP message needs to be sent. The connection remains opened until the user or application decides to close it. Sending and receiving multiple SNMP messages over a single (D)TLS connection has been successfully tested.

- 4.1 X.509 Certificates

  Both entities have used X.509 certificates for authentication.

- 4.1.1 Provisioning for the Certificate

  Usage of a root certificate for certificate verification has also been tested.

- 4.2 (D)TLS Usage

  Both, client and server side have been authenticated by X.509 certificates. For DTLS (over UDP), each SNMP message is placed in a single UDP datagram. Packet fragmentation/concatenation has been enabled.

- 8.3 contextEngineID Discovery

  ContextEngineID Discovery as defined in RFC 5343 has been successfully tested, for which a separate interoperability report was submitted.

- 9.3 Use with SNMPv1/SNMPv2c Messages

  Usage of SNMPv1, SNMPv2c and SNMPv3 with USM security model over (D)TSL is disabled in MG-SOFT's WinSNMP API implementation.

Untested Requirements

- 3.1.2 Message Protection

  MG-SOFT's WinSNMP API implementation does not implement the optional mapping between TLS security algorithms and SNMP security levels.

- 3.1.3 (D)TLS Connections

  Coexistence and operation of multiple (D)TLS connections has not been tested.

- 3.3 Notification and Proxy

  These requirements have not been tested since only a command generator was available at the time of testing.

- 4.1.1 Provisioning for the Certificate

  Mapping of incoming message to tmSecurityName has not been

tested. Mapping of a certificate's fingerprint value to a tmSecurityName has not been tested.

- 4.4.1.1 tmSecurityName

  Mapping from certificate to tmSecurityName has not been tested.

- 8.1 Sessions

  Lifetime limitation of established sessions has not been tested.

- 8.2 Notification Receiver Credential Selection

  Notifications have not been tested.

- 9.1 Certificates, Authentication and Authorization

  Implementation of the SNMP-TLS-TM-MIB has not been tested.

10. Security Considerations

The interoperability testing did not identify any security issues that are not covered in the security considerations of the relevant specifications.

11. IANA Considerations

This document has no IANA actions.

12. Informative References

[RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", STD 62, RFC 3411, December 2002.

[RFC3412] Case, J., Harrington, D., Presuhn, R., and B. Wijnen, "Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)", STD 62, RFC 3412, December 2002.

[RFC3414] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", STD 62, RFC 3414, December 2002.

[RFC3418] Presuhn, R., "Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)", STD 62, RFC 3418, December 2002.

[RFC5343] Schoenwaelder, J., "Simple Network Management Protocol (SNMP) Context EngineID Discovery", RFC 5343, September 2008.

[RFC5590] Harrington, D. and J. Schoenwaelder, "Transport Subsystem for the Simple Network Management Protocol (SNMP)", RFC 5590, June 2009.

[RFC5591] Harrington, D. and W. Hardaker, "Transport Security Model for the Simple Network Management Protocol (SNMP)", RFC 5591, June 2009.

[RFC5953] Hardaker, W., "Transport Layer Security (TLS) Transport Model for the Simple Network Management Protocol (SNMP)", RFC 5953, August 2010.

Authors' Addresses

Juergen Schoenwaelder (editor)
Jacobs University Bremen

Email: j.schoenwaelder@jacobs-university.de

Robert Story
SPARTA/Cobham

Email: robert.story@cobham.com

Matjaz Vrecko
MG-SOFT

Email: matjaz@mg-soft.si