Using SNMPv3 with NMIS for Secure Network Management

NMIS supports using SNMPv3 for securing the collection of sensitive network information. This is especially important from core switches and routers which if compromised could have a considerable business impact. This configuration note does not include details about the SNMPv3 protocol, and assumes that people are wanting to use the authPriv (Authentication and Privilege) mode which is the most secure.

IMPORTANT NOTE ON 256 BIT ENCRYPTION PROTOCOL SUPPORT

Regarding SNMPv3 protocol support, different vendors and even different products support different combinations of authentication and privilege protocols. The above example is for an older Cisco router, newer devices support SHA256 and AES256, the combinations will depend on your device.

Configuring Cisco IOS for SNMPv3

The first step is to enable SNMPv3 on your router or switch. If using Cisco IOS, the commands are below, if using other Cisco operating systems or other vendors, the concepts are the same and the commands will likely be similar. The most important thing is that the device will support SNMPv3, it will require encryption features if you want to use full auth/priv mode.

Required Cisco IOS Configuration for SNMPv3 communication to NMIS

The following three lines of Cisco IOS commands are required to enable SNMPv3 on the Cisco IOS device. When running a show run, the configured user will not show up in the running configuration, the configured users can be viewed by running the command "show snmp user".

```
snmp-server view NMIS8RO iso included
snmp-server group NMIS8 v3 priv match exact read NMIS8RO
snmp-server user nmis8 NMIS8 v3 auth md5 nmis4242 priv des nmis4242
```

The commands above will create a user called **nmis8,** with an **authorisation password** of **nmis4242** and a **privilege password** of **nmis4242**

View the configured SNMP users

```
asgard# show snmp user
User name: nmis8
Engine ID: 800000090300001E13B18D00
storage-type: nonvolatile active
Authentication Protocol: MD5
Privacy Protocol: DES
Group-name: NMIS8
```

Details about Cisco IOS SNMPv3

More details about Cisco IOS SNMPv3 can be found at http://www.cisco.com/en/US/docs/ios/12_0t/12_0t3/feature/guide/Snmp3.html

Configuring Linux SNMP Daemon (Net-SNMP SNMPD) for SNMPv3

The first step is to enable SNMPv3 on in the /etc/snmp/snmpd.conf file, then restart the daemon.

Required Linux SNMPD Configuration for SNMPv3 communication to NMIS9

Add the following configuration to the top, edit the /etc/snmp/snmpd.conf file as the root user, e.g.

```
sudo vi /etc/snmp/snmpd.conf
```

Add the following configuration replacing the username and passwords is you require.

```
createUser nmis SHA banana4242 AES monkey4242
rouser nmis priv 1.3.6.1
```

The commands above will create a user called **nmis,** with an **authorisation password** of **banana4242** and a **privilege password** of **monkey4242**

The view of 1.3.6.1, will permit access to the Standard MIB and the Enterprise MIB, essentially providing full access.

Restart the SNMP Daemon

```
sudo service snmpd restart
```

Testing your SNMPv3 Configuration with NET-SNMP

To verify that SNMPv3 is working as configured run the following command. Change the username and passwords if you have used different ones.

```
snmpwalk -v 3 -l authPriv -u nmis -a sha -A banana4242 -x aes -X monkey4242 <HOSTNAME> .1.3.6.1.2.1.1
```

Configuring NMIS9 for SNMPv3

Prerequisites

To use SNMP version 3 NMIS requires two perl modules that are not normally or automatically installed, `Crypt::DES` and `Digest::HMAC`.
From version 8.5.14 onwards the installer will try to install these modules; until then you'll have to resolve this dependency by hand:

- on Debian or Ubuntu use: `sudo apt-get install libcrypt-des-perl libdigest-hmac-perl`
- on RedHat/CentOS use: `sudo yum install perl-Digest-HMAC perl-Crypt-DES`
- or, if neither option appeals you may also use CPAN: `sudo  cpan Crypt::DES Digest::HMAC`.

Change Nodes.nmis Configuration

You can edit a Node using the NMIS GUI to include support for SNMPv3, as described in Adding and Editing a Device in NMIS8 (same works in NMIS9).

For **NMIS 9** it can be done using the GUI or the node admin tool.

You will need to modify the node configuration to use SNMPv3, the user name, protocols and passwords need to match the above IOS or Linux configuration.

```
/usr/local/nmis9/admin/node_admin.pl act=set node=<YOURNODENAME> \
entry.configuration.authpassword=banana4242 \
entry.configuration.authprotocol=sha \
entry.configuration.privpassword=monkey4242 \
entry.configuration.privprotocol=aes \
entry.configuration.username=nmis \
entry.configuration.version=snmpv3
```

Test NMIS SNMPv3 communication to the device

Ensure NMIS has the necessary encryption modules installed, it may be missing Crypt::DES, you will only need to do this if you see an error message below

```
cpan
install Crypt::DES
```

Run a test NMIS collect to the device using SNMPv3

```
/usr/local/nmis8/bin/nmis.pl type=collect node=asgard debug=true
```

For **nmis9**, it can be done:

```
/usr/local/nmis9/bin/nmis-cli type=schedule job.type=collect job.node=asgard job.verbosity=7
```

In nmis9, the credentials can be also tested with an admin tool:

```
/usr/local/nmis9/admin/tests.pl act=snmp node=NODENAME
```

An example output:

```
*** Testing snmp with snmpget snmpv2c
 Running...     snmpget -v 2c -c **** host.opmantek.net 1.3.6.1.2.1.1.1.0
 Result: iso.3.6.1.2.1.1.1.0 = STRING: "Cisco IOS Software, 1841 Software (C1841-ADVENTERPRISEK9-M), Version
12.4(25f), RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2011 by Cisco Systems, Inc.
Compiled Tue 16-Aug-11 06:21 by prod_rel_team"



*** Testing snmp with internal NMIS API
 SNMP session open to HOST success

    Result: Cisco IOS Software, 1841 Software (C1841-ADVENTERPRISEK9-M), Version 12.4(25f), RELEASE SOFTWARE
(fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2011 by Cisco Systems, Inc.
Compiled Tue 16-Aug-11 06:21 by prod_rel_team

** Model: CiscoRouter
```

In the command output you are looking to verify that data was collected from the device, so any updates to an RRD will show that data was collected and is being stored.

```
--snip--
11:19:02 updateRRD, DS MemoryUsedPROC:MemoryFreePROC:avgBusy5:avgBusy1:bufferFail:bufferElHit:MemoryFreeIO:
bufferElFree:MemoryUsedIO
11:19:02 updateRRD, value N:19299276:27249732:3:2:0:810903:30345952:1118:4257056
--snip--
```

You should now be using SNMPv3 to communicate with this device from NMIS8.

Updating NMIS9 to support SHA256 and AES256 including Cisco variants

The history of encryption in SNMPv3 is long and winding and possibly interesting to some people, the reality is that the only consistency with SNMPv3 implementations is the inconsistency in the implementations by different vendors and projects. Frequently combinations of protocols are not supported (or do not work), so you need to find the matching combinations.

Once you have updated the SNMP libraries NMIS uses for SNMP, you should be able to use the following protocols for SNMPv3.

Authentication Protocols

NMIS9 name	Name	OID	Notes
md5	usmHMACMD5AuthProtocol	1.3.6.1.6.3.10.1.1.2	RFC3411
sha (or sha1)	usmHMACSHAAuthProtocol	1.3.6.1.6.3.10.1.1.3	RFC3411
sha224	usmHMAC128SHA224AuthProtocol	1.3.6.1.6.3.10.1.1.4	RFC3411
sha256	usmHMAC192SHA256AuthProtocol	1.3.6.1.6.3.10.1.1.5	RFC3411
sha384	usmHMAC256SHA384AuthProtocol	1.3.6.1.6.3.10.1.1.6	RFC3411
sha512	usmHMAC384SHA512AuthProtocol	1.3.6.1.6.3.10.1.1.7	RFC3411

Privilege Protocols

NMIS9 name	Name	OID	Notes
des	usmDESPrivProtocol	1.3.6.1.6.3.10.1.2.1	RFC3411
3des	usm3DESPrivProtocol	1.3.6.1.4.1.14832.1	RFC3411
aes (or aes128)	usmAESCfb128PrivProtocol	1.3.6.1.4.1.14832.2	**Blumenthal implementation** of SNMPv3
aes192	usmAESCfb192PrivProtocol	1.3.6.1.4.1.14832.3	**Blumenthal implementation** of SNMPv3
aes256	usmAESCfb256PrivProtocol	1.3.6.1.4.1.14832.4	**Blumenthal implementation** of SNMPv3
aes192c	cusmAESCfb192PrivProtocol	1.3.6.1.4.1.9.12.6.1.1	**Cisco implementation** of SNMPv3 AES192
aes256c	cusmAESCfb256PrivProtocol	1.3.6.1.4.1.9.12.6.1.2	**Cisco implementation** of SNMPv3 AES256
aes192c2	usmAES192Cisco2PrivProtocol	1.3.6.1.4.1.9.12.6.1.101	A mysterious version 2 of the Cisco implementation, possibly related to pysnmp
aes256c2	usmAES256Cisco2PrivProtocol	1.3.6.1.4.1.9.12.6.1.102	A mysterious version 2 of the Cisco implementation, possibly related to pysnmp

Update or Install Perl Modules

Crypt::Rijndael module needs to be installed for AES support. This is the command to install Crypt::Rijndael if this module is not already installed and will ensure we have the latest version:

```
sudo cpanm Crypt::Rijndael --sudo
```

Net::SNMP module needs to be up to date - currently v6.0.1 - this command will ensure we have the latest version:

```
sudo cpanm Net::SNMP --sudo
```

Patch Net::SNMP::Security::USM to support 256 bit and higher encryption

The NMIS development team have added support to the Net::SNMP library using the work done recently by the team and leveraging the work done by Napsty @ https://raw.githubusercontent.com/Napsty/scripts/master/perl-net-snmp-sha2/USM.pm

If you are using NMIS 9.4.3 or earlier you will need to obtain the contrib folder from GitHub @ https://github.com/Opmantek/nmis9/tree/nmis9_dev/contrib/perl-net-snmp-256

We will use a patched Net::SNMP::Security::USM, for Net::SNMP v6.0.1, which is backwards compatible with all snmp protocol strings used in the original Net::SNMP::Security::USM module.
All protocol strings are case-insensitive.

Copy the shipped USM.pm from the contrib folder and replace the Net::SNMP v6.0.1 version.

On RedHat 8 based systems (including our CentOS Virtual Machine)

```
sudo cp /usr/share/perl5/vendor_perl/Net/SNMP/Security/USM.pm /usr/share/perl5/vendor_perl/Net/SNMP/Security/USM.pm.original
sudo cp /usr/local/nmis9/contrib/perl-net-snmp-256/USM.pm /usr/share/perl5/vendor_perl/Net/SNMP/Security/USM.pm
```

On Debian/Ubuntu based systems

```
sudo cp /usr/share/perl5/Net/SNMP/Security/USM.pm /usr/share/perl5/Net/SNMP/Security/USM.pm.original
sudo cp /usr/local/nmis9/contrib/perl-net-snmp-256/USM.pm /usr/share/perl5/Net/SNMP/Security/USM.pm
```

Find where USM.pm is installed

Older Linux versions will have the Perl module somewhere else, the fastest way to find it is to use find.

```
sudo find /usr -name USM.pm
```

Restart the NMIS9 Daemon

```
sudo systemctl restart nmis9d
```

Update NMIS GUI to show new options

```
# Make a copy of original incase you have customization and forget to add it
# If command says it doesnt exisit you can skip to next command
sudo cp /usr/local/nmis9/conf/Table-Nodes.nmis /usr/local/nmis9/conf/Table-Nodes.nmis.bak
# Adding in new SNMPv3 Options
sudo cp /usr/local/nmis9/contrib/perl-net-snmp-256/Table-Nodes.nmis /usr/local/nmis9/conf
```

Testing SNMPv3 quickly

The contrib folder includes a lightweight SNMP testing tool, which differs from the nmis9/admin/tests.pl tool, in that it does not use net-snmp Linux package at all, it purely exercises the NMIS SNMP libraries.

```
/usr/local/nmis9/contrib/perl-net-snmp-256/test-snmp.pl node=lab-fortigate

SNMP test results for lab-fortigate:
  Open SNMP session to lab-fortigate
    Auth Protocol: sha, Priv Protocol: aes
  Testing SNMP session
  Performing SNMP get of 1.3.6.1.2.1.1.1.0 and 1.3.6.1.2.1.1.2.0
    sysDescr: lab-fortigate-int
    sysObjectID: 1.3.6.1.4.1.12356.101.1.65

SNMP PASSED
```

To quickly change NMIS configuration to use a different combination, update the device and commit/apply changes.

Update NMIS node details:

```
/usr/local/nmis9/admin/node_admin.pl act=set node=lab-fortigate entry.configuration.authprotocol=sha256 entry.
configuration.privprotocol=aes256c
```

Repeat your SNMP test

```
SNMP test results for lab-fortigate:
  Open SNMP session to lab-fortigate
    Auth Protocol: sha256, Priv Protocol: aes256c
  Testing SNMP session
  Performing SNMP get of 1.3.6.1.2.1.1.1.0 and 1.3.6.1.2.1.1.2.0
    sysDescr: lab-fortigate-int
    sysObjectID: 1.3.6.1.4.1.12356.101.1.65

SNMP PASSED
```

More on Vendor Support for SHA and AES 256

In testing the NMIS development team found that the implementation of SNMP options was not consistent.

For example on a Fortigate device, the administration GUI allowed setting SHA256 and AES256 but these would not work together. When SHA256 and AES256 Cisco were used, the system was very happy.

Many Cisco devices will support SHA256 but only AES128 (which given the entropy is reasonable "AES-128 would take about 2.61*10^12 years to crack" https://www.ubiqsecurity.com/128bit-or-256bit-encryption-which-to-use/).

At the time of writing (March 2023) net-snmp on Linux does not include support for AES256 by default (including SNAP repositories). net-snmp does support AES256, you just need to compile if yourself.

SNMPv3 Error Messages and How to Decode Them

No response from remote host during synchronization

The test-snmp.pl tool would show this:

```
ERROR: Could not open SNMP session to node lab-fortigate: No response from remote host "lab-fortigate-int.
opmantek.net" during synchronization
```

This means you have the wrong authentication protocol or password, you will need to change them and try again

No response from remote host

The test-snmp.pl tool would show this:

```
ERROR: Could not retrieve SNMP vars from node lab-fortigate: No response from remote host "lab-fortigate-int.
opmantek.net"
```

This means you have the wrong privilege protocol or password, you will need to change them and try again

The authProtocol is unknown during discovery

The test-snmp.pl tool would show this:

```
ERROR: Could not open SNMP session to node lab-fortigate: The authProtocol "1.3.6.1.6.3.10.1.1.5" is unknown
during discovery
```

This means the remote SNMP agent in the end device (node) does not know what this authentication protocol is.

Confirmed working combinations

The below is a list of confirmed working SNMPv3 combinations across a variety of different vendor and operating systems.

This is by no means a comprehensive list of the products we support.

Vendor / Operating System	SHA	AES	NMIS Considerations
Cisco IOS	SHA1	AES256	aes256c needs to be configured as the entry.configuration.privprotocol value
	SHA1	AES192	aes192c needs to be configured as the entry.configuration.privprotocol value
	SHA1	AES128	
Cisco NX-OS	SHA1	AES128	
	SHA256	AES128	
Fortinet	SHA1	AES	
	SHA224	AES256 Cisco	sha224 needs to be configured as the entry.configuration.authprotocol value **AND** aes256c needs to be configured as the entry.configuration.privprotocol value
	SHA256	AES256 Cisco	sha256 needs to be configured as the entry.configuration.authprotocol value **AND** aes256c needs to be configured as the entry.configuration.privprotocol value
	SHA384	AES256 Cisco	sha384 needs to be configured as the entry.configuration.authprotocol value **AND** aes256c needs to be configured as the entry.configuration.privprotocol value
	SHA512	AES256 Cisco	sha512 needs to be configured as the entry.configuration.authprotocol value **AND** aes256c needs to be configured as the entry.configuration.privprotocol value
Palo Alto	SHA1	AES128	
	SHA224	AES128	
	SHA256	AES128	
	SHA384	AES128	
	SHA224	AES192	aes192c needs to be configured as the entry.configuration.privprotocol value
	SHA256	AES192	aes192c needs to be configured as the entry.configuration.privprotocol value
	SHA256	AES256	aes256c needs to be configured as the entry.configuration.privprotocol value
	SHA384	AES192	aes192c needs to be configured as the entry.configuration.privprotocol value
	SHA384	AES256	aes256c needs to be configured as the entry.configuration.privprotocol value
NET-SNMP (Tested on v5.8 with Ubuntu 20.04)	SHA512	AES128	sha512 needs to be configured as the entry.configuration.authprotocol value

You may notice that when configuring SNMPv3 on a (for example) Cisco IOS device that there is not an explicit AES192C/AES256C in the command, rather it is needed to be defined as AES 192 and/or AES 256.

When configuring the device for NMIS, you will need to explicitly tell it to use AES192C/AES256C using node_admin.pl (example covered previously).

Related Topics

- Opmantek Virtual Machine: Implementing SNMPv3 AES256 in the NMIS9 VM for Secure Network Management