

NuDesign SNMPv3 Agent Service Proxy Forwarder Application Configuration Demo Guide

Introduction

This Demo provides for experimentation with a sample set of two SNMPv1/v2c Agents that can be proxied to through the appropriately configured NuDesign SNMPv3 Agent Service.

The two subordinate agents accept SNMP requests on ports 11161 and 12161, they also send SNMP notifications to a receiver listening on port 4162. The only functional difference between these two agents is their configuration.

In order to run the Demo you will first need to download and install the Evals of NuDesign SNMPv3 Service and MIBrowser Pro. They come with a set of default configurations that make SNMP Manager (Browser Pro) - SNMP Agent communications operational quickly.

To install and start the Demo agents, unzip the downloaded Demo file into a temporary directory and run the batch file "startV2Agent1.bat". This will start the Agent1 operating on port 11161 and using the configuration file "SNMPV2cAgent1.xnv". Later on, as an exercise, use the enclosed configuration file "SNMPV2cAgent2.xnv" to add the Agent2, operating on port 12161, by running the batch file "startV2agent2.bat".

To simplify the discussion a bit, this Demo will use only SNMPv3 to communicate with and through the proxy SNMPv3 Service Agent, to the proxied SNMPv1/v2c Agents. You can use v1 or v2 protocol access as well, but for this example, the communications have been limited to secure V3 communications. Of course communications between the SNMPv3 Service and the Demo's v1/v2c Agents will always be limited to SNMPv1/v2c protocol.

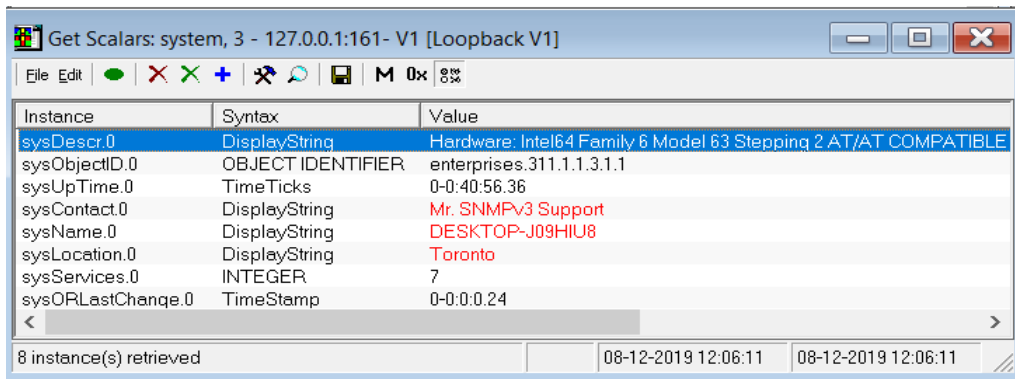
You'll first need to complete configuring MIBrowser Pro and SNMPv3 Service to talk to these SNMPv1/v2c Agents.

Verifying installation & functions of MIBrowser Pro & SNMPv3 Service

First let's verify the installation and operations of NuDesign SNMPv3 Service and MIBrowser Pro. Their default configurations provide for SNMPv1 communications on the loopback address 127.0.0.1, port 161 (the loopback address 127.0.0.1 is used throughout this demo).

Select the 127.0.0.1:161 [Loopback V1] session from the "SNMP Agent" list box on the Browser's "Browser" tab. Now navigate to the "System" node on the Browser Tree below and select "Get Scalars" from the mouse right click menu.

You should see a window appear with the contents similar to a screen capture below.



Now verify the reception of SNMP notifications. From the Browser's top menu line open the "Notification Receiver" box by clicking on "Trap Rx" button.

Then from Start / NDT SNMPv3 Agent Service Programs Group open the SNMPv3 Configuration Editor (stopping the SNMPv3 Service in response to a query). The Editor will be used extensively later, now simply inspect the Notify, Target Address and Target Parameters Tables.

Note that Service notification of "type" Trap has already been enabled (row n1 in the Notify Table, row n2 will be reviewed shortly too) and it uses "tag1" parameter which identifies where the Trap will be sent to, review the t1 row in the Target Address Table, and note that security parameters are specified via entries of row p2 in the Target Parameters Table.

Before quitting the Editor it might be useful to disable one of the standard MS extension DLLs that will be sending multiple notifications that will distract from our Proxy Trap forwarding testing. Under "Subagents / Dynamic Loading" tick off the "inetmib1" DLL. This will unload this DLL on the next start and at this point only the "coldStart" v2Trap will be generated by SNMPv3 Service.

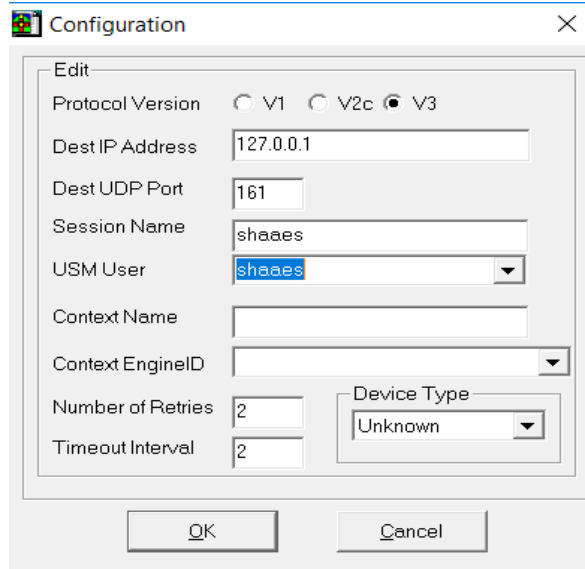
Press Save and Quit the Editor (re-starting the Service in response to the query). Note that the Notification Receiver displays the "coldStart" v2Trap originated by the SNMPv3 Service upon re-start.



BTW, when working with Browser's dialog windows, press green "Clear Query" button to clear the window in preparation for new results.

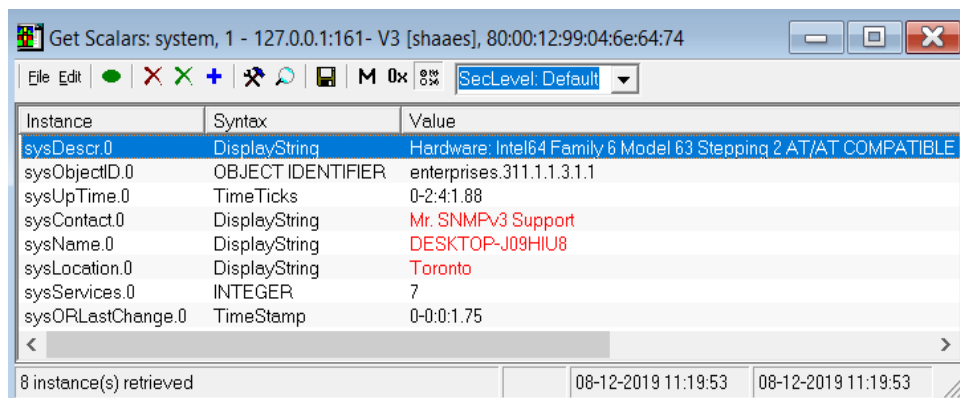
Enabling secure SNMPv3 communications between MIBrowser Pro and SNMPv3 Service

Open the Browser's "Edit | SNMP Agents" menu. You will be shown "SNMP Agents" dialog box. From there select the 'Add' button. You should see a dialog similar to the image below.



You should be able to select "shaaes" entry in the "USM User" dropdown menu due to the default configuration provided by the MIBrowser Pro install. Update the other fields as per provided dialog sample. Press the 'OK' button on the 'Configuration' dialog, and the 'OK' button in the 'SNMP Agents' dialog.

Since the "shaaes" User is also a default SNMPv3 Service User, you can now verify that the setup you've just created works. Select the agent session you've just created from the "SNMP Agent" list box on the Browser's "Browser" tab. Navigate again to the "System" node on the tree and select "Get Scalars" from the mouse right click menu. You should see a window appear as per below. Note the Title bar identifies secure "shaaes" v3 user now.

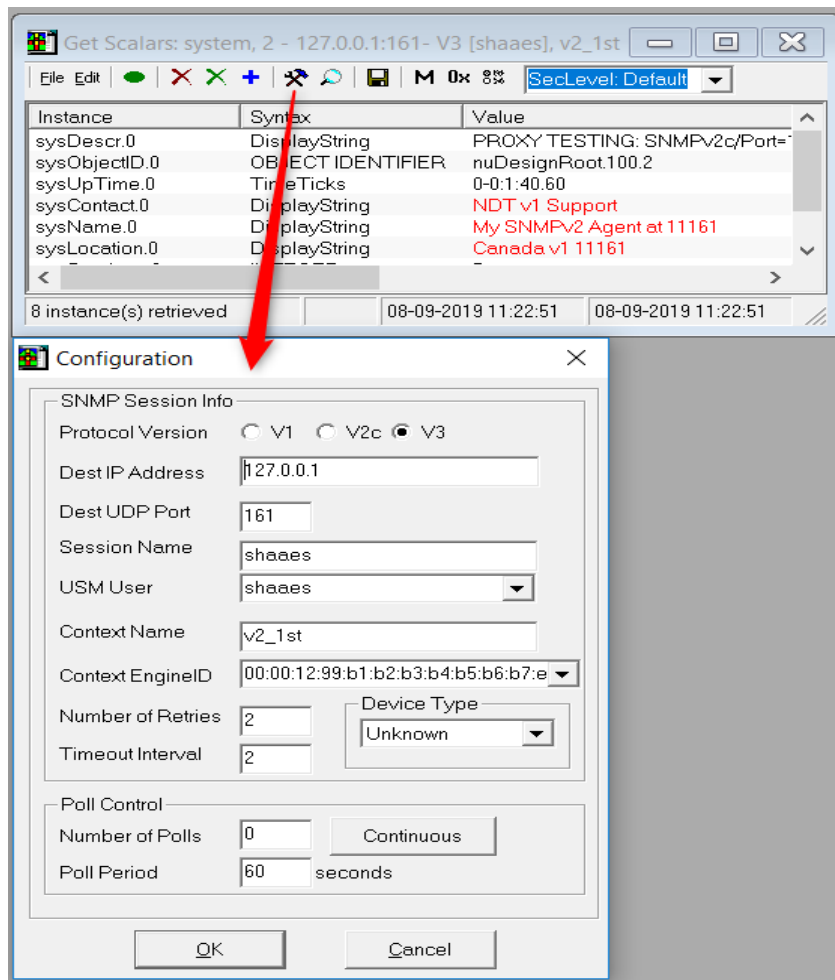


Enabling read / write access to SNMPv2 Test Agent via SNMPv3 Service Proxy Forwarder

Review the SNMPv2cAgent1.xnv file from the "Proxy_Demo.zip" file. It is a configuration file for the TestSNMPAg.exe Agent. It enables v1/v2c access to the Agent on port 11161, and provides for coldStart / warmStart notifications being sent to receiver on port 4162.

In order for Browser to access this v2 Agent via SNMPv3 Service Proxy Forwarder Applications the following configurations of SNMPv3 Service need to be added.

The goal is to execute read / write access to the v2 Test Agent going via secure SNMPv3 Access to the SNMPv3 Service. The Browser's "System" node / "Get Scalars" function (the mouse right click menu) will now need to be updated by providing ContextEngineID / ContextName, pointing to the v2Agent, via the Configuration dialog box opened via the "Configure" button on the "Get Scalars" form. The preselected here Context values are: 00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6 / v2_1st.



To get the results, as displayed above, provide the following updates to the SNMPv3 Service Configuration Tables (highlighted entries) using the SNMPv3 Service Configuration Editor tool:

Proxy Table:

Name	ContextEngineID	ContextN...	Type	Target...	SingleT...	MultipleTa...
ND-v1X-I	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2n_1st	inform(...)	p22		tag3
ND-v1X-R	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st	read(1)	p4	t3	tag2
ND-v1X-T	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2n_1st	trap(3)	p22		tag2
ND-v1X-W	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st	write(2)	p4	t4	

Since the above "Get Scalars" request is sent to the Service which has EngineId = 80:00:12:99:04:6e:64:74 and it is different than the request's Context EngineId = 00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6 the request is passed to the Proxy Table where a "read" row is selected, identifying secure side of the exchange (Browser - Service), "p4" parameters, and v2 side of the exchange (Service - v2Agent), "t3" index value that identifies "p22" parameters, as per the tables below.

Target Parameters Table:

Name	MPModel	SecurityModel	SecurityName	SecurityLevel
p1	mpSNMPv1(0)	secSNMPv1(1)	public	noAuthNoPriv(1)
p2	mpSNMPv2c(1)	secSNMPv2c(2)	public	noAuthNoPriv(1)
p22	mpSNMPv2c(1)	secSNMPv2c(2)	MyV2Agent	noAuthNoPriv(1)
p23	mpSNMPv2c(1)	secSNMPv2c(2)	MyV2Agent_W	noAuthNoPriv(1)
p3	mpSNMPv3(3)	secUSM(3)	private	noAuthNoPriv(1)
p4	mpSNMPv3(3)	secUSM(3)	shaaes	authPriv(3)

Target Address Table:

Name	TDo...	TAddress	Tim...	Retry	TagList	Para...	TMask	MMS
t1	udp	127.0.0.1:162	5	0	tag1	p2		484
t2	udp	127.0.0.1:162	5	0	tag2	p4		484
t3	udp	127.0.0.1:11161	5	0	tagx	p22		484
t4	udp	127.0.0.1:11161	5	0	tagx	p23		484
t5	udp	127.0.0.1:162	5	0	tag3	p6		484

Community Table:

Com...	Name	Security Name	Context EngineID	Context N...
c1	public	public	80:00:12:99:04:6e:64:74	
c2	private	public	80:00:12:99:04:6e:64:74	
c3	public	MyV2Agent	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st
c4	private	MyV2Agent_W	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st
c5	public_tr	MyV2Agent	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2n_1st

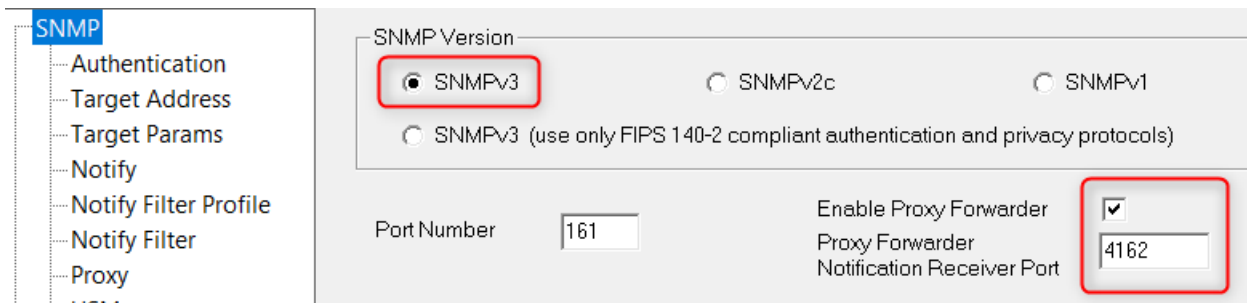
This document is provided for informational purposes only, 'as is' and without warranty of any kind, using it you are consenting to NuDesign SNMPv3 Agent Service for Windows License Agreement. This document may be copied, however without any modification, all pages, full text and notices must be included.

Finally the "Community Table" is consulted and for a given Security Name (MyV2Agent) and v2Agent identified by ContextEngineId / ContextName pair, the "c3" row's Community Name is selected, and "public" community name is selected for v1/v2 request send by the SNMPv3 Service to v2Agent to fulfill the "Read" request.

The reply traces a similar path back to reach the Browser as a v3 reply send over with the secure "sha / aes" user credentials. The SNMP Set (write) request follows a similar path with "write" / p23 / t4 / c4" rows holding appropriate parameters.

Converting SNMPv2 Test Agent's v2_Traps into v3_Traps by Proxy Forwarder

Run the Service Configuration Editor again and on the SNMP tab leave the Proxy Forwarder function enabled, while setting its Notification Receiver port to 4162.



In order to have the v2Agent v2 Traps converted and re-send by Service's Proxy function as v3 Traps, provide the following updates to the SNMP Configuration Tables (highlighted entries), using the Service Configuration Editor tool:

Proxy Table:

The v2 Trap originated by v2 Test Agent is received by SNMPv3 Service Proxy Application on port 4162. Its Community string "public_tr" is mapped in row c5 of Community Table into a "trap" row with ContextEngineId / ContextName that identifies row NS-v1X-T" in the Proxy Table, confirming unsecure Service - v2Agent side of exchange credentials via "p22" parameter (TargetParamsTable) and identifying "tag2" and "p4" target parameters in the Target Address Table for the secure Service - Browser side of the exchange.

Name	ContextEngineID	ContextN...	Type	Targe...	Sin...	Multi...
ND-v1X-I	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2n_1st	inform...	p22		tag3
ND-v1X-R	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st	read(1)	p4	t3	
ND-v1X-T	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2n_1st	trap(3)	p22		tag2
ND-v1X-W	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st	write(2)	p4	t4	

Target Parameters Table:

snmpTargetParamsTable					
Name	MPModel	SecurityModel	SecurityName	SecurityLevel	
p1	mpSNMPv1(0)	secSNMPv1(1)	public	noAuthNoPriv(1)	
p2	mpSNMPv2c(1)	secSNMPv2c(2)	public	noAuthNoPriv(1)	
p22	mpSNMPv2c(1)	secSNMPv2c(2)	MyV2Agent	noAuthNoPriv(1)	
p23	mpSNMPv2c(1)	secSNMPv2c(2)	MyV2Agent_W	noAuthNoPriv(1)	
p3	mpSNMPv3(3)	secUSM(3)	private	noAuthNoPriv(1)	
p4	mpSNMPv3(3)	secUSM(3)	shaaes	authPriv(3)	

Target Address Table:

snmpTargetAddrTable								
Na...	TDom...	TAddress	Tim...	Retry	TagList	Para...	T.	MMS
t1	udp	127.0.0.1:162	5	0	tag1	p2		484
t2	udp	127.0.0.1:162	5	0	tag2	p4		484
t3	udp	127.0.0.1:11161	5	0	tagx	p22		484
t4	udp	127.0.0.1:11161	5	0	tagx	p23		484
t5	udp	127.0.0.1:162	5	0	tag3	p6		484

Community Table:

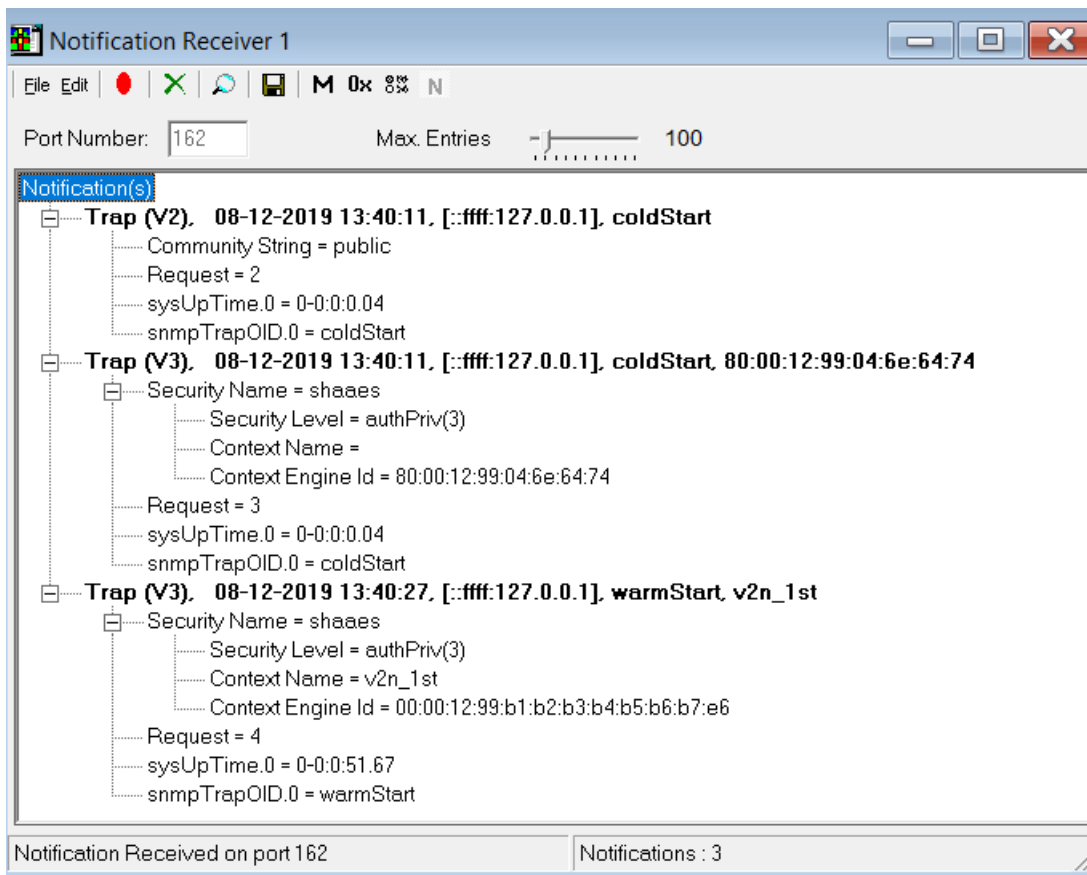
snmpCommunityTable						
Co...	Name	Security Name	Context EngineID	Context N...	Tran	
c1	public	public	80:00:12:99:04:6e:64:74			
c2	private	public	80:00:12:99:04:6e:64:74			
c3	public	MyV2Agent	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st		
c4	private	MyV2Agent_W	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st		
c5	public_tr	MyV2Agent	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2n_1st		

The v2 Test Agent console interface provides for sending "warmStart" notifications and both, Traps and Informs are enabled, see the SNMPV2cAgent1.xnv configuration file. The above SNMPv3 Proxy Forwarder configuration will translate the v2Trap into v3Trap, the Inform from the v2Agent will time out at this point (until the next section's configuration updates are implemented).

At this point one can also add v3 Trap configuration for SNMPv3 Service's own notifications by adding an "n2" trap row in the "Notify Table", pointing to a newly created t2 row in a Target Address Table that uses "tag2" parameter, enabling secure "p4" parameters of Target Parameters Table for the secure "Service - Browser" side of the exchange.

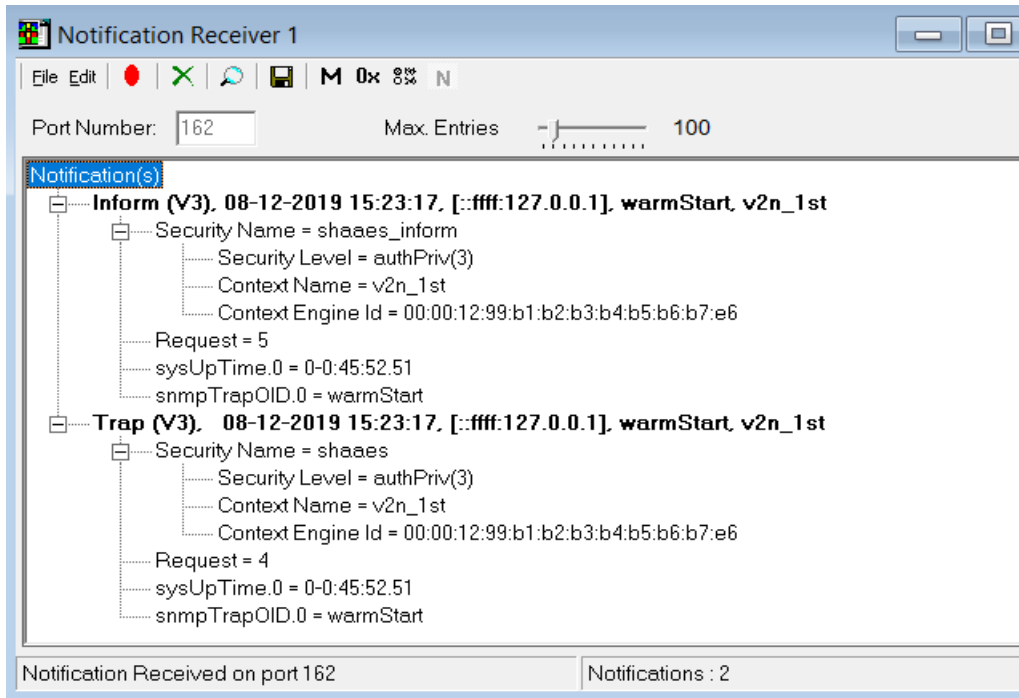
Name	Tag	Type
mstrap_n1	mstraps	trap(1)
n1	tag1	trap(1)
n2	tag2	trap(1)

Restarting the Service now will produce both, the v2 and v3 "coldStart" Traps from the Service itself. Type "warm" in the v2 Agent console and v3 "warmStart" Trap that originated as a v2 Trap in a v2 Agent and was translated into v3Trap by the Proxy Forwarder Application appears too, as per below screen capture of Browser's Notification Receiver.



Converting SNMPv2 Test Agent's v2_Informs into v3_Informs by Proxy Forwarder

To get the Notification Receiver window's results as per below, provide updates to the SNMP Configuration Tables (highlighted entries), using the Service Configuration Editor tool. Then from the v2Agent console issue a "warm" command, note that startV2agent1.xnv configuration file enables both Traps & Informs, that are converted by Proxy App into v3Trap and v3Inform.



Proxy Table:

Name	ContextEngineID	Context...	Type	Targ...	Sin...	Multi...
ND-v1X-I	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2n_1st	inform(4)	p22		tag3
ND-v1X-R	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st	read(1)	p4	t3	
ND-v1X-T	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2n_1st	trap(3)	p22		tag2
ND-v1X-W	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st	write(2)	p4	t4	

Target Parameters Table:

Name	MpModel	SecurityModel	SecurityName	SecurityLevel
p2	mpSNMPv2c(1)	secSNMPv2c(2)	public	noAuthNoPriv(1)
p22	mpSNMPv2c(1)	secSNMPv2c(2)	MyV2Agent	noAuthNoPriv(1)
p23	mpSNMPv2c(1)	secSNMPv2c(2)	MyV2Agent_W	noAuthNoPriv(1)
p3	mpSNMPv3(3)	secUSM(3)	private	noAuthNoPriv(1)
p4	mpSNMPv3(3)	secUSM(3)	shaaes	authPriv(3)
p6	mpSNMPv3(3)	secUSM(3)	shaaes_inform	authPriv(3)

This document is provided for informational purposes only, 'as is' and without warranty of any kind, using it you are consenting to NuDesign SNMPv3 Agent Service for Windows License Agreement. This document may be copied, however without any modification, all pages, full text and notices must be included.

Target Address Table:

snmpTargetAddrTable								
Na...	TD...	TAddress	Tim...	Retry	TagList	Para...	T..	MMS
t1	udp	127.0.0.1:162	5	0	tag1	p2		484
t2	udp	127.0.0.1:162	5	0	tag2	p4		484
t3	udp	127.0.0.1:11161	5	0	tagx	p22		484
t4	udp	127.0.0.1:11161	5	0	taqx	p23		484
t5	udp	127.0.0.1:162	5	0	taq3	p6		484

Community Table:

snmpCommunityTable						
Co...	Name	Security Name	Context EngineID	Context N...	Tran	
c1	public	public	80:00:12:99:04:6e:64:74			
c2	private	public	80:00:12:99:04:6e:64:74			
c3	public	MyV2Agent	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st		
c4	private	MyV2Agent_W	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2_1st		
c5	public_tr	MyV2Agent	00:00:12:99:b1:b2:b3:b4:b5:b6:b7:e6	v2n_1st		

Please note that Informs require an update to USM Tables of both, respectively the Browser and the SNMPv3 Service, with the USM User containing Browser's EngineID (since this is the authoritative EngineID in the Inform packets exchange).

V3 USM Users						
Name	EngineID	Host	Auth.	Priv.	Pr...	
public	00:00:12:99:7f:00:00:01:3c:19:32:b2	MIBrowser Pro	none	no...		
public	00:00:12:99:7f:00:00:01:ab:cd:ef:bb		none	no...		
sha	80:00:12:99:04:6e:64:74	[::ffff:127.0.0.1]...	SHA	no...		
sha	00:00:12:99:7f:00:00:01:ab:cd:ef:bb		SHA	no...		
shaaes	80:00:12:99:04:6e:64:74	[::ffff:127.0.0.1]...	SHA	AE...	2b...	
shaaes	00:00:12:99:7f:00:00:01:ab:cd:ef:bb		SHA	AE...	06...	
shaaes_inform	00:00:12:99:7f:00:00:01:3c:19:32:b2	MIBrowser Pro	SHA	AE...	56...	
shades	80:00:12:99:04:6e:64:74	[::ffff:127.0.0.1]...	SHA	DES	39...	
shades	00:00:12:99:7f:00:00:01:ab:cd:ef:bb		SHA	DES	92...	
shanopriv	80:00:12:99:04:6e:64:74	[::ffff:127.0.0.1]...	SHA	no...		
shanopriv	00:00:12:99:7f:00:00:01:ab:cd:ef:bb		SHA	no...		

usmUserTable				
EngineID	Name	AuthProtocol	PrivProtocol	
80:00:12:99:04:6e:64:74	public	none(1)	none(1)	
80:00:12:99:04:6e:64:74	shaaes	SHA(3)	AES128(4)	
80:00:12:99:04:6e:64:74	shades	SHA(3)	DES(2)	
80:00:12:99:04:6e:64:74	md5nopriv	MD5(2)	none(1)	
80:00:12:99:04:6e:64:74	shanopriv	SHA(3)	none(1)	
00:00:12:99:7f:00:00:01:3c:19:32:b2	shaaes_inform	SHA(3)	AES128(4)	

The Browser's USM Table is under menu Tools / Options then V3 USM Users tab, the Service USM Table is accessible via Service Configuration Editor, USM branch under the SNMP subtree.

While setting up this new USM user select respectively the same AuthProtocol and PrivProtocol passwords in both USM Tables.

Adding additional SNMPv2 Test Agents

As an exercise add similar proxy configurations to the SNMPv3 Service tables for your second test v2 Agent that can be started at port 12161 using enclosed configuration file "SNMPV2cAgent2.xnv" and startup batch file "startV2agent2.bat".

It's ContextEngineId / ContextName pair values could be set to =
00:00:12:99:b1:b2:b3:b4:b5:b6:b7:f6 / v2_2nd.

About NuDesign Technologies

NuDesign Technologies, Inc, based in Toronto, Canada, specializes in the development of SNMP management agents and client applications used in remote configuration, monitoring and control of Windows and Linux Servers & Workstations, embedded devices, networking services and applications.

For all of our software products, supporting standard management protocols such as NETCONF, SNMP, CLI and web protocols such as RESTCONF, please visit our main site at www.ndt-inc.com.

The benefits of deploying NuDesign's management software technologies are reliable, low risk, quick-to-market and well supported solutions. The highly automated code generation tools with associated applications and tutorials enable fast prototyping and development. They also facilitate organization and design process for multiple target environments.

Contact Information



NuDesign Technologies, Inc.

tel: 416 737 0328 / fax: 416 445 9101

toll free: 1 866 886 SNMP (7667)

contact@ndt-inc.com

www.ndt-inc.com / www.snmp4windows.com

This document is provided for informational purposes only, 'as is' and without warranty of any kind, using it you are consenting to NuDesign SNMPv3 Agent Service for Windows License Agreement. This document may be copied, however without any modification, all pages, full text and notices must be included.