

SNMP exercises, Part I

Contents

1	Introduction	1
1.1	Goals	1
1.2	Notes	2
1.3	0. Installing client (manager) tools	2
1.4	1. Configure SNMP on Your Router	2
1.5	2. Testing SNMP	3
1.6	3. SNMP Walk and OIDs	4
1.7	4. Configuration of snmpd on your PC	4
1.8	5. Check that snmpd is working:	5
1.9	6. Test your neighbors	6
1.10	7. Adding MIBs	6
1.11	8. SNMPwalk - the rest of MIB-II	7
1.12	9. More MIB-OID fun	7

1 Introduction

1.1 Goals

- Install and learn to use the SNMP commands
- Explore and identify standard vs enterprise parts of the MIB tree
- Install vendor specific MIBs and use those with the SNMP commands

1.2 Notes

- Commands preceded with “\$” imply that you should execute the command as a general user - not as root.
- Commands preceded with “#” imply that you should be working as root.
- Commands with more specific command lines (e.g. “rtrX>” or “mysql>”) imply that you are executing commands on remote equipment, or within another program.

1.3 0. Installing client (manager) tools

Start by installing the net-snmp tools:

```
$ sudo apt-get install snmp
$ sudo apt-get install snmp-mibs-downloader
```

The second of the two commands downloads the standard IETF and IANA SNMP MIBs which are not included by default.

Note: for this to work, you must enable the “multiverse” source in your APT configuration, if you are using Ubuntu 12.04. This has already been done for you in this workshop.

Now, edit the file `/etc/snmp/snmp.conf`

Change this line:

```
mibs :
```

... so that it looks like:

```
# mibs :
```

(You are “commenting out” the empty mibs statement, which was telling the snmp* tools **not** to automatically load the mibs in the `/usr/share/mibs/` directory)

1.4 1. Configure SNMP on Your Router

For this exercise you need to work in groups. Assign one person to type on the keyboard.

If you are unsure of what group you are in refer to the Network Diagram on the classroom wiki by going to <http://noc.ws.nsrc.org/> and clicking on the Network Diagram link.

Now connect to your router:

```
$ ssh cisco@rtrN.ws.nsrc.org      (or "ssh cisco@10.10.N.254")
```

```
username: cisco  
password: <CLASS PASSWORD>
```

```
rtrN> enable  
Password: <CLASS PASSWORD>  
rtrN# configure terminal          (conf t)
```

Now we need to add an Access Control List rule for SNMP access, turn on SNMP, assign a read-only SNMP community string and tell the router to maintain SNMP information across reboots. To do this we do:

```
rtrN(config)# access-list 99 permit 10.10.0.0 0.0.255.255  
rtrN(config)# snmp-server community NetManage ro 99  
rtrN(config)# snmp-server ifindex persist
```

Now let's exit and save this new configuration to the routers permanent config.

```
rtrN(config)# exit  
rtrN# write memory                (wr mem)  
rtrN# exit                        (until you return to your pc)
```

Now to see if your changes are working.

1.5 2. Testing SNMP

To check that your SNMP installation works, run the `snmpstatus` command on each of the following devices

```
$ snmpstatus -c 'NetManage' -v 2c <IP_ADDRESS>
```

Where is each of the following:

- * The NOC server: 10.10.0.250
- * Your group's router: 10.10.N.254
- * The backbone switch: 10.10.0.253
- * The access points: 10.10.0.251, 10.10.0.252

What happens if you try using the wrong community string (i.e. change `NetManage` to something else?)

1.6 3. SNMP Walk and OIDs

Now, you are going to use the ‘snmpwalk’ command, part of the SNMP toolkit, to list the tables associated with the OIDs listed below, on each piece of equipment you tried above:

```
.1.3.6.1.2.1.2.2.1.2
.1.3.6.1.2.1.31.1.1.1.18
.1.3.6.1.4.1.9.9.13.1
.1.3.6.1.2.1.25.2.3.1
.1.3.6.1.2.1.25.4.2.1
```

You will try this with two forms of the ‘snmpwalk’ command:

```
$ snmpwalk -c 'NetManage' -v 2c <IP_ADDRESS> <OID>
```

and

```
$ snmpwalk -On -c 'NetManage' -v 2c <IP_ADDRESS> <OID>
```

... where `OID` is one of the OIDs listed above: `.1.3.6...`

... where `IP_ADDRESS` can be your group’s router...

Note: the `-On` option turns on numerical output, i.e.: no translation of the OID <-> MIB object takes place.

For these OIDs:

- a) Do all the devices answer ?
- b) Do you notice anything important about the OID on the output ?

1.7 4. Configuration of snmpd on your PC

For this exercise your group needs to verify that the `snmpd` service is running and responding to queries for all machines in your group. First enable `snmpd` on your machine, then test if your machine is responding, then check each machine of your other group members.

- Install the SNMP agent (daemon)

```
$ sudo apt-get install snmpd
```

- Configuration.

We will make a backup of the distributed config, and then we will create our own:

```
$ cd /etc/snmp
$ sudo mv snmpd.conf snmpd.conf.dist
$ sudo editor snmpd.conf
```

Then, copy/paste the following:

```
# Listen for connections on all interfaces (both IPv4 *and* IPv6)
agentAddress udp:161,udp6:[::1]:161

# Configure Read-Only community and restrict who can connect
rocommunity NetManage 10.10.0.0/16
rocommunity NetManage 127.0.0.1

# Information about this host
sysLocation NSRC Network Management Workshop
sysContact sysadm@pcX.ws.nsrc.org

# Which OSI layers are active in this host
# (Application + End-to-End layers)
sysServices 72

# Include proprietary dskTable MIB (in addition to hrStorageTable)
includeAllDisks 10%
```

Now save and exit from the editor.

- Restart snmpd

```
$ sudo service snmpd restart
```

1.8 5. Check that snmpd is working:

```
$ snmpstatus -c NetManage -v 2c localhost
```

What do you observe ?

1.9 6. Test your neighbors

Check now that you can run `snmpstatus` against your other group members servers:

```
$ snmpstatus -c NetManage -v 2c pcN.ws.nsrc.org
```

For instance, in group 5, you should verify against:

```
* pc17.ws.nsrc.org
* pc18.ws.nsrc.org
* pc19.ws.nsrc.org
* pc20.ws.nsrc.org
```

1.10 7. Adding MIBs

Remember when you ran:

```
$ snmpwalk -c NetManage -v 2c 10.10.X.254 .1.3.6.1.4.1.9.9.13.1
```

If you noticed, the SNMP client (`snmpwalk`) couldn't interpret all the OIDs coming back from the Agent:

```
SNMPv2-SMI::enterprises.9.9.13.1.3.1.2.1 = STRING: "chassis"
SNMPv2-SMI::enterprises.9.9.13.1.3.1.6.1 = INTEGER: 1
```

What is 9.9.13.1.3.1 ?

To be able to interpret this information, we need to download extra MIBs:

We will use the following MIBs (Don't download them yet!):

CISCO MIBS

```
ftp://ftp.cisco.com/pub/mibs/v2/CISCO-SMI.my
ftp://ftp.cisco.com/pub/mibs/v2/CISCO-ENVMON-MIB.my
```

To make it easier, we have a local mirror on <http://noc.ws.nsrc.org/mibs/>

Download them now as follows:

```
$ sudo apt-get install wget
$ cd /usr/share/mibs
$ sudo mkdir cisco
$ cd cisco
```

```
$ sudo wget http://noc.ws.nsrc.org/mibs/CISCO-ENVMON-MIB.my
$ sudo wget http://noc.ws.nsrc.org/mibs/CISCO-SMI.my
```

Now we need to tell the snmp tools that we have the cisco MIBS it should load.
So edit the file `/etc/snmp/snmp.conf`, and add the following two lines:

```
mibdirs +/usr/share/mibs/cisco
mibs +CISCO-ENVMON-MIB:CISCO-SMI
```

Save the file, quit.

Now, try again:

```
$ snmpwalk -c NetManage -v 2c 10.10.X.254 .1.3.6.1.4.1.9.9.13.1
```

What do you notice ?

1.11 8. SNMPwalk - the rest of MIB-II

Try and run `snmpwalk` on any hosts (routers, switches, machines) you have not tried yet, in the 10.10.0.X network

Note the kind of information you can obtain.

```
$ snmpwalk -c NetManage -v 2c 10.10.0.X ifDescr
$ snmpwalk -c NetManage -v 2c 10.10.0.X ifAlias
$ snmpwalk -c NetManage -v 2c 10.10.0.X ifTable | less
$ snmpwalk -c NetManage -v 2c 10.10.0.X ifXTable | less
$ snmpwalk -c NetManage -v 2c 10.10.0.X ifOperStatus
$ snmpwalk -c NetManage -v 2c 10.10.0.X ifAdminStatus
$ snmpwalk -c NetManage -v 2c 10.10.0.X if
```

(Remember that with 'less' you press space for next page, 'b' to go back to previous page, and 'q' to quit)

Can you see what's different between `ifTable` and `ifXTable`?

What do you think might be the difference between `ifOperStatus` and `ifAdminStatus`? Can you imagine a scenario where this could be useful ?

1.12 9. More MIB-OID fun

- Use SNMP to examine:
 - a. the running processes on your neighbor's server (`hrSWRun`)
 - b. the amount of free disk space on your neighbor's server (`hrStorage`)
 - c. the interfaces on your neighbor's server (`ifIndex`, `ifDescr`)

Can you use short names to walk these OID tables ?

- Experiment with the “snmptranslate” command, example:

```
$ snmptranslate .1.3.6.1.4.1.9.9.13.1
```

- Try with various OIDs