

Lab 5: SNMP in Cisco Routers

NET311 - Computer Networks Management

Instructor: Dr. Mostafa Dahshan

Objectives

1. Configuring SNMP agent on Cisco Routers.
2. Understanding SNMP v2 traps.

References

1. Cisco Networking Academy, Lab 8.2.2.4 Configuring SNMP.
2. GNS3 Documentation.

Instructions

1. Read the lab instructions.
2. Provide question answers and screenshots in the supplied answer sheet.
3. After finishing the lab, upload your saved answer sheet to LMS.

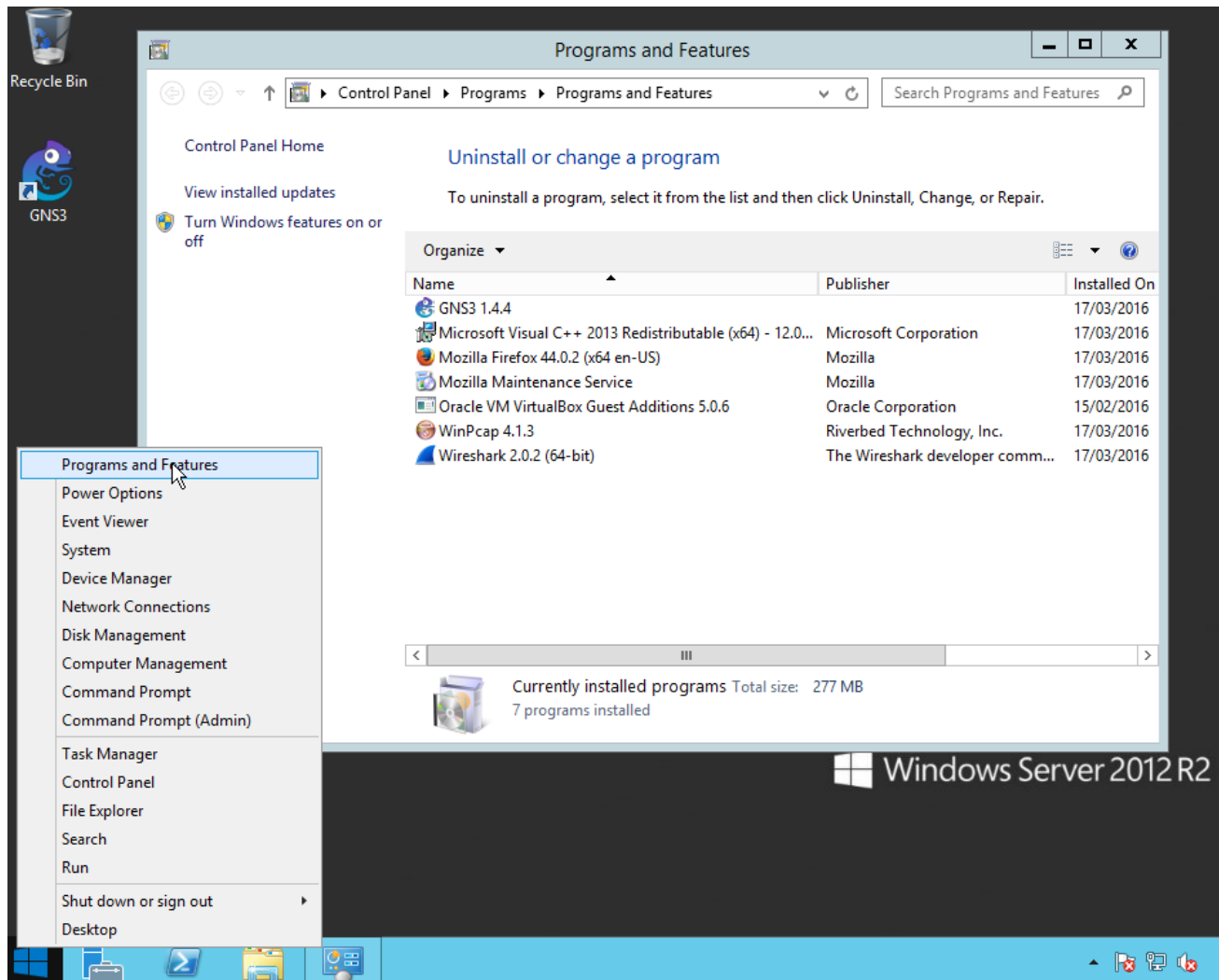
Part 1: Lab Setup

The following programs need to be installed on your PC before starting this lab.

1. Winpcap 4.1.3
2. Wireshark 2.0.2
3. GNS3 1.4.4
4. SnmpB

Refer to the supplemental video for setting up the environment of this lab.

1. Verify that the programs are installed.



Lab sheet 1.1: provide a screenshot of the Programs and Features screen.

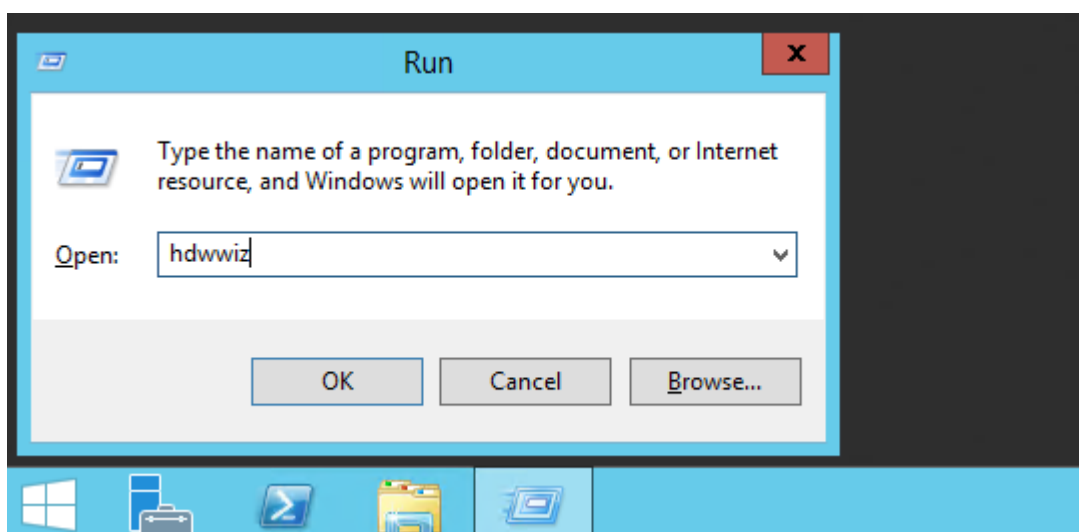
Install the Microsoft KM-TEST Loopback Adapter using the Hardware Wizard

2. Right click on the Windows icon and click **Run**.

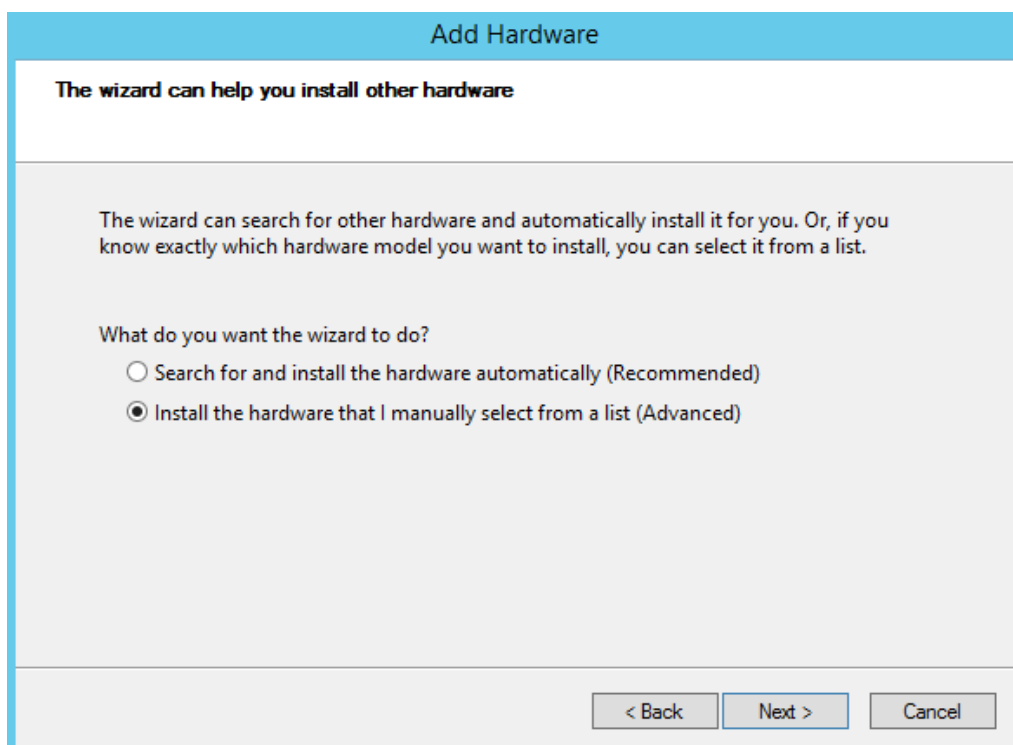


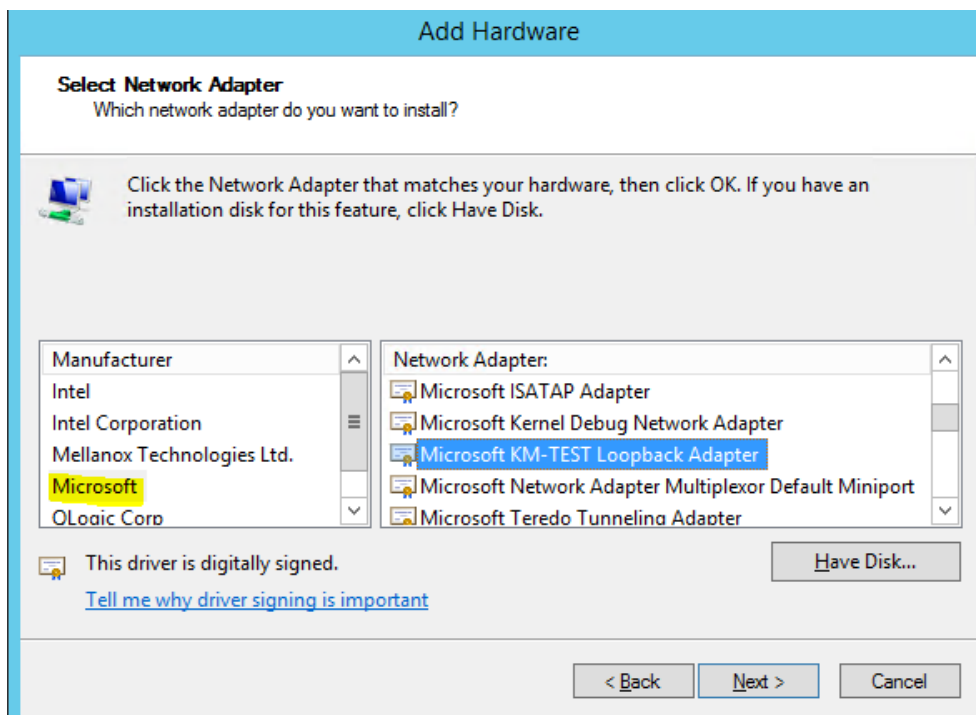
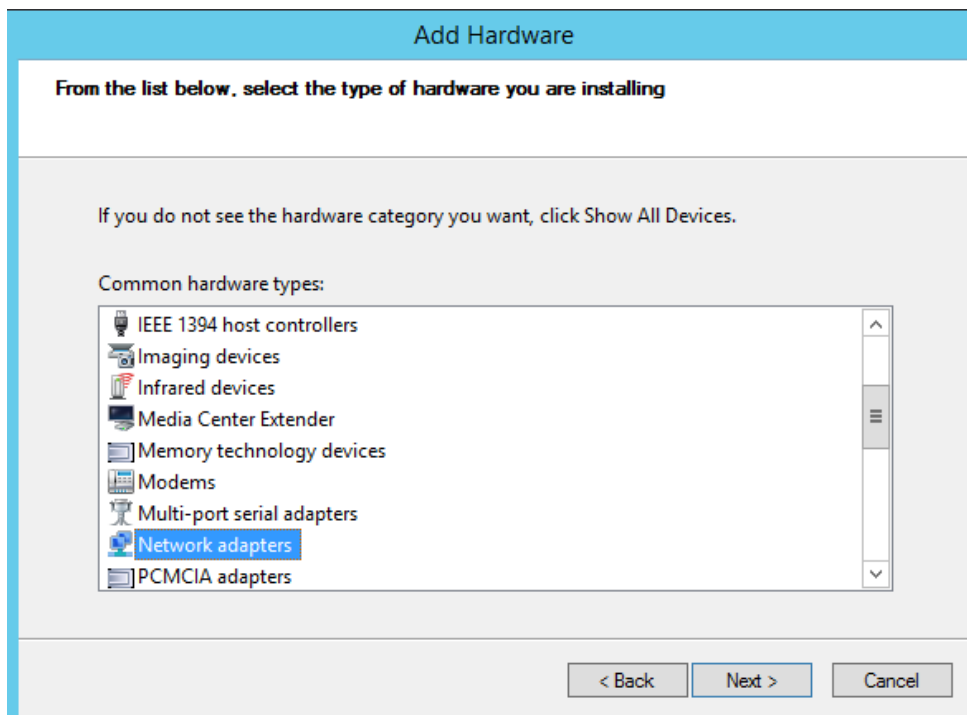
3. Type the command

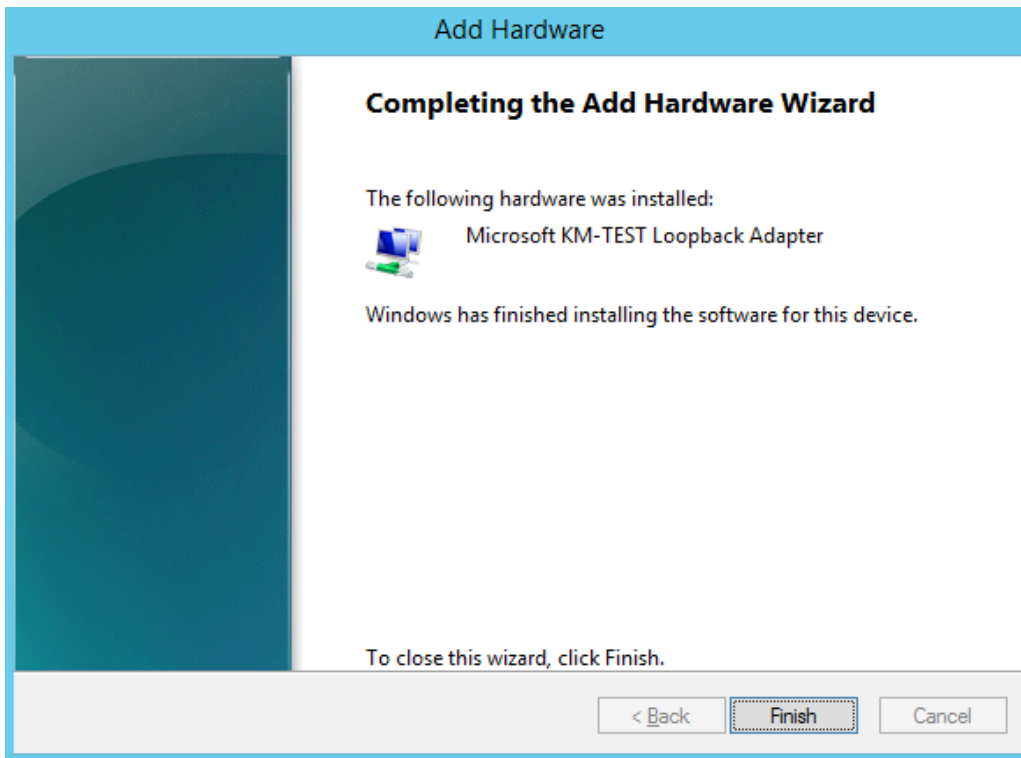
hdwwiz



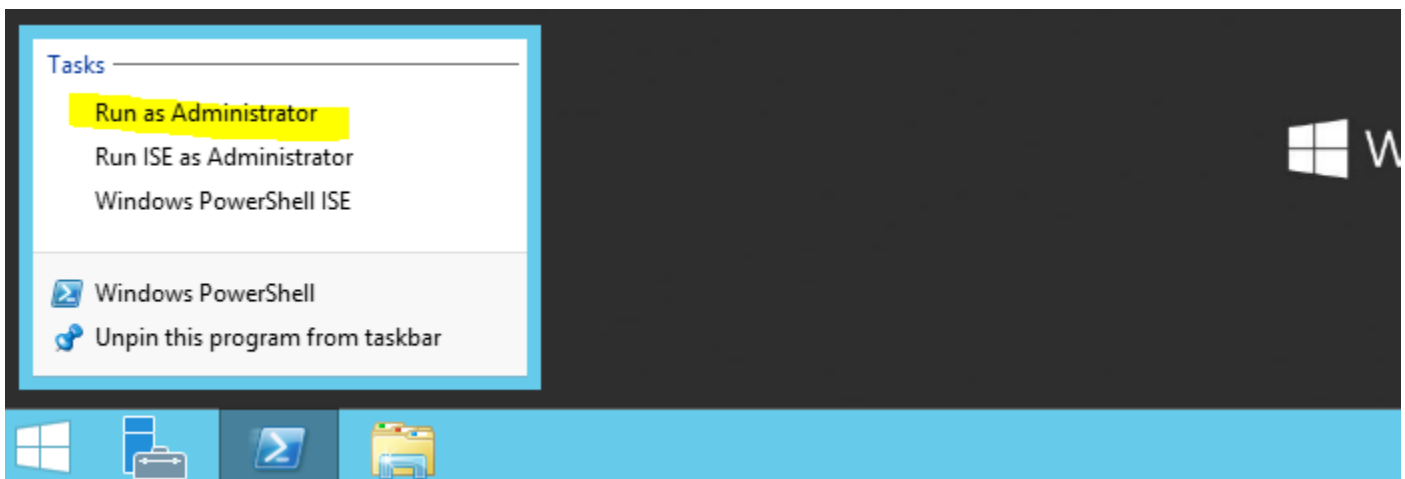
4. Follow the instructions







5. Run **PowerShell** as **Administrator**.



6. Type the following commands to configure the Microsoft KM-TEST Loopback Adapter:

```
$adapter=Get-NetAdapter -InterfaceDescription "Microsoft KM-TEST Loopback Adapter"
Rename-NetAdapter -InterfaceDescription "Microsoft KM-TEST Loopback Adapter" -NewName
"Loop1"
New-NetIPAddress -InterfaceIndex $adapter.ifIndex -AddressFamily "IPv4" -IPAddress
"172.16.0.2" -PrefixLength 24
Route add 172.16.0.0 mask 255.240.0.0 172.16.0.2 metric 1
```

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2014 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> $adapter=Get-NetAdapter -InterfaceDescription "Microsoft KM-TEST Loopback Adapter"
PS C:\Windows\system32> Rename-NetAdapter -InterfaceDescription "Microsoft KM-TEST Loopback Adapter" -NewName "Loop1"
PS C:\Windows\system32> New-NetIPAddress -InterfaceIndex $adapter.ifIndex -AddressFamily "IPv4" -IPAddress "172.16.0.2"
-PrefixLength 24

IPAddress           : 172.16.0.2
InterfaceIndex      : 15
InterfaceAlias      : Loop1
AddressFamily       : IPv4
Type                : Unicast
PrefixLength        : 24
PrefixOrigin        : Manual
SuffixOrigin        : Manual
AddressState        : Tentative
ValidLifetime       : Infinite ([TimeSpan]::MaxValue)
PreferredLifetime   : Infinite ([TimeSpan]::MaxValue)
SkipAsSource        : False
PolicyStore         : ActiveStore

IPAddress           : 172.16.0.2
InterfaceIndex      : 15
InterfaceAlias      : Loop1
AddressFamily       : IPv4
Type                : Unicast
PrefixLength        : 24
PrefixOrigin        : Manual
SuffixOrigin        : Manual
AddressState        : Invalid
ValidLifetime       : Infinite ([TimeSpan]::MaxValue)
PreferredLifetime   : Infinite ([TimeSpan]::MaxValue)
SkipAsSource        : False
PolicyStore         : PersistentStore

PS C:\Windows\system32> Route add 172.16.0.0 mask 255.240.0.0 172.16.0.1 metric 1
OK!
PS C:\Windows\system32> _
```

Lab sheet 1.2: provide a screenshot of the PowerShell screen.

To allow SnmpB to receive traps, create a rule in Windows Firewall with Advanced Features to allow the program SnmpB through the firewall.

7. Type the following command to add a rule to allow SNMP PDUs through the firewall:

```
New-NetFirewallRule -DisplayName "SNMP" -Direction Inbound -Action Allow -Protocol UDP -LocalPort 161-162
```

```
Select Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2014 Microsoft Corporation. All rights reserved.

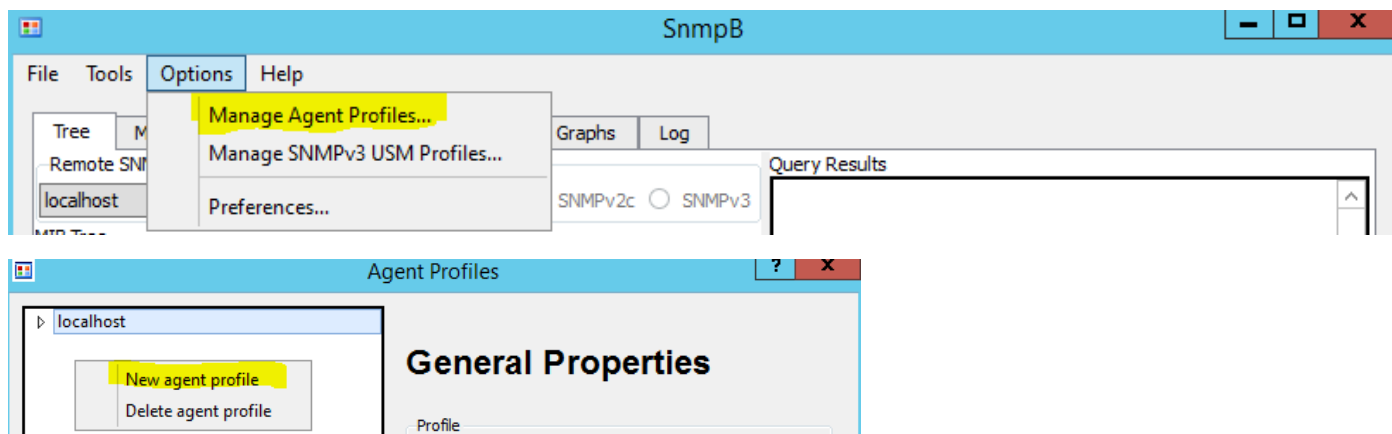
PS C:\Windows\system32> New-NetFirewallRule -DisplayName "SNMP" -Direction Inbound -Action Allow -Protocol UDP -LocalPort 161-162

Name                : {b1fb2fdb-2095-4ade-be6c-f9bb785d443a}
DisplayName          : SNMP
Description          :
DisplayGroup         :
Group                :
Enabled              : True
Profile              : Any
Platform             : {}
Direction            : Inbound
Action               : Allow
EdgeTraversalPolicy  : Block
LooseSourceMapping   : False
LocalOnlyMapping     : False
Owner                :
PrimaryStatus        : OK
Status               : The rule was parsed successfully from the store. (65536)
EnforcementStatus    : NotApplicable
PolicyStoreSource    : PersistentStore
PolicyStoreSourceType : Local
```

Lab sheet 1.3: provide a screenshot of the PowerShell screen.

Part 2: Configure SNMP Manager

1. Run the **SnmpB** program.
2. Go to **Manage Agent Profiles** and add a profile called **R1**, using the following parameters:

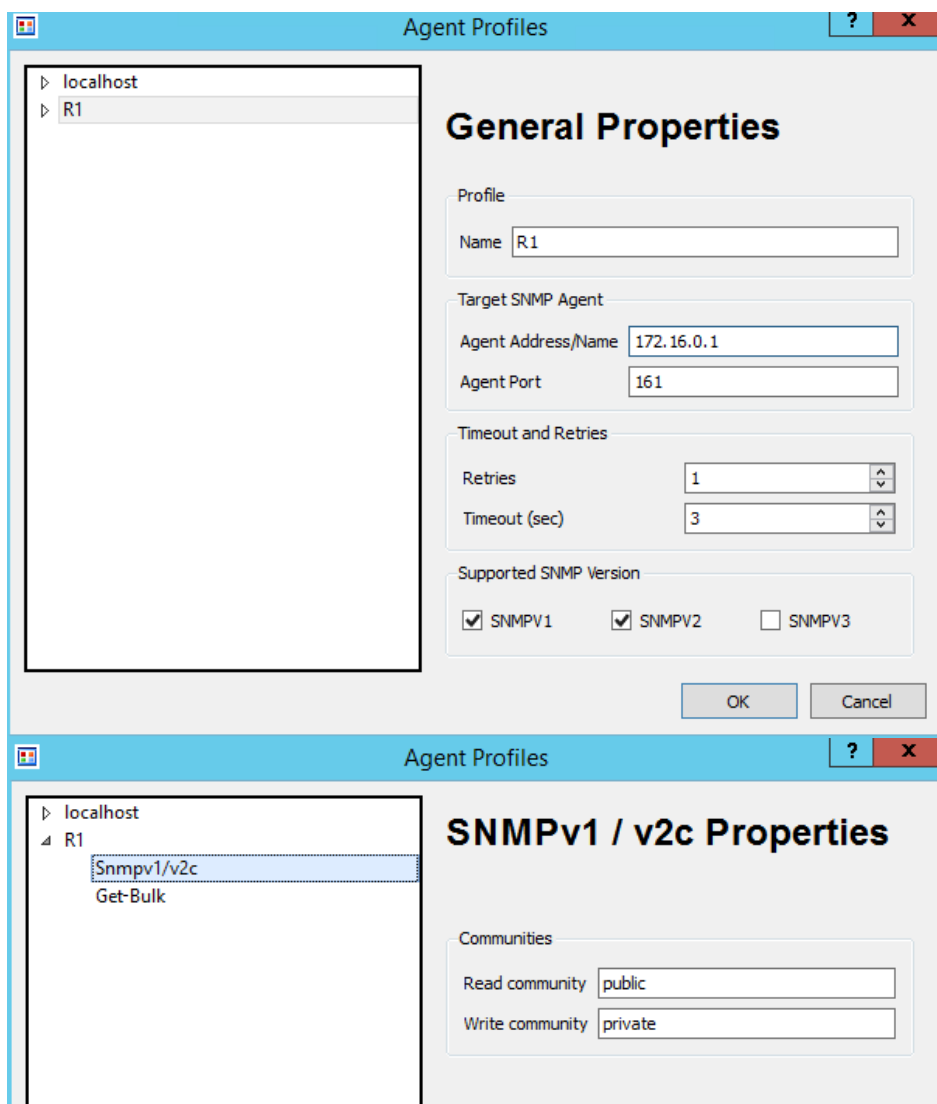


Name: R1

Agent address:
172.16.0.1

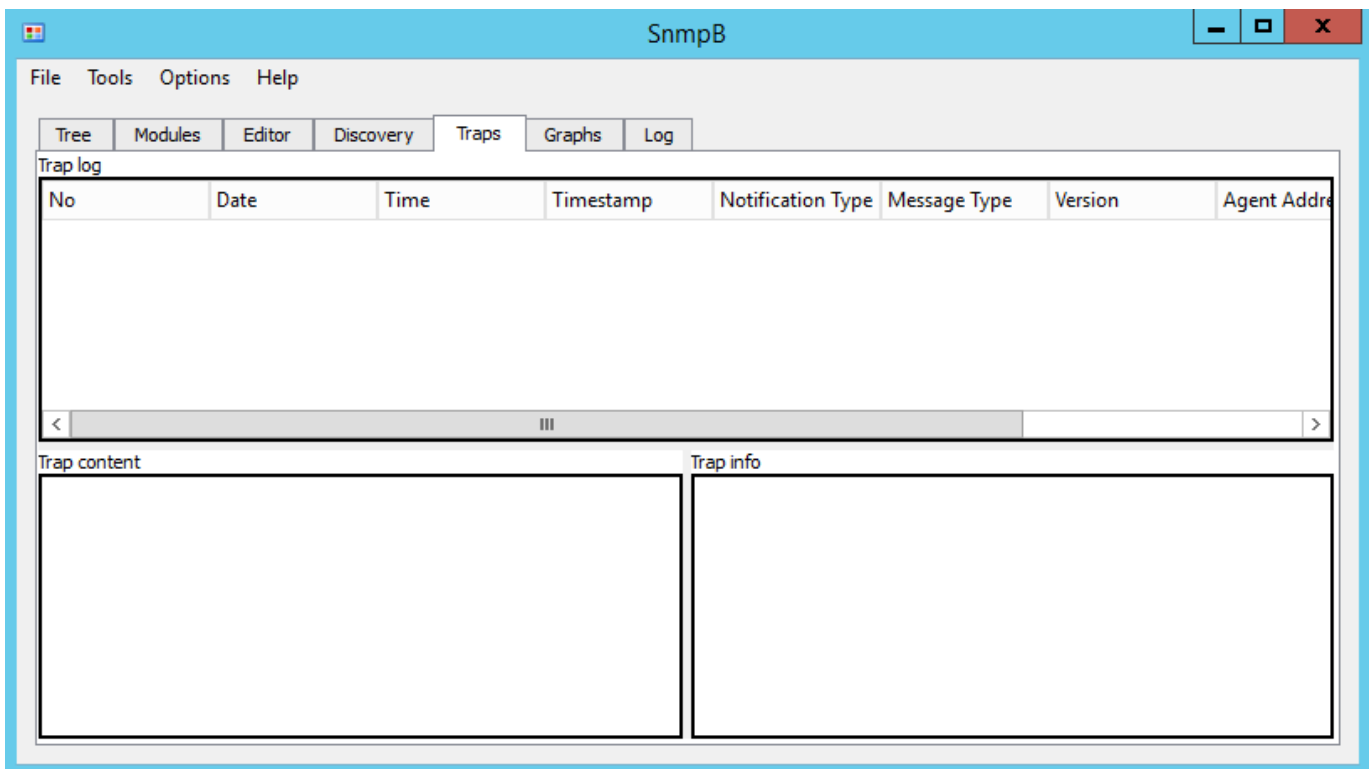
Supported SNMP Version:
SNMPv1, SNMPv2

Read community: **public**



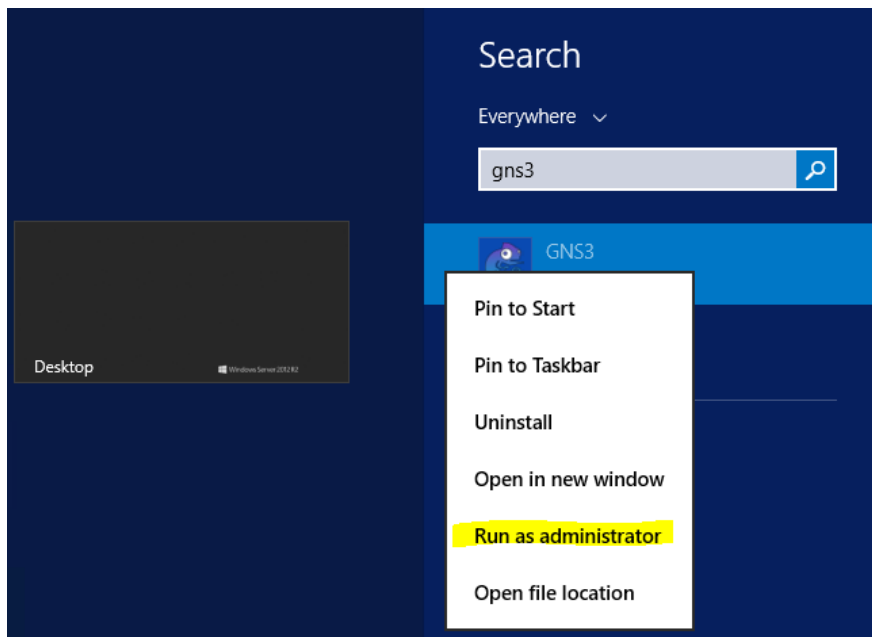
Lab sheet 2.1: provide a screenshot of the SnmpB Agent Profiles screen.

3. Go to the Traps tab and leave the SnmpB window running to watch for traps.

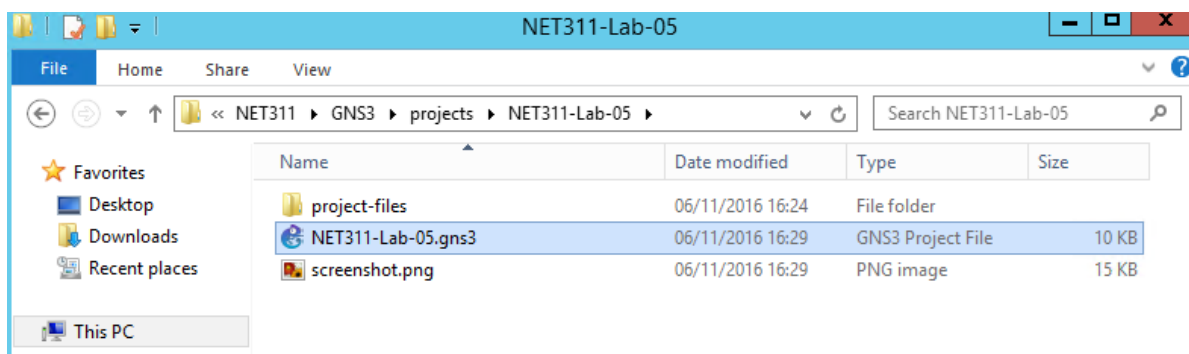


Part 3: Configure SNMP Agent on Cisco Router

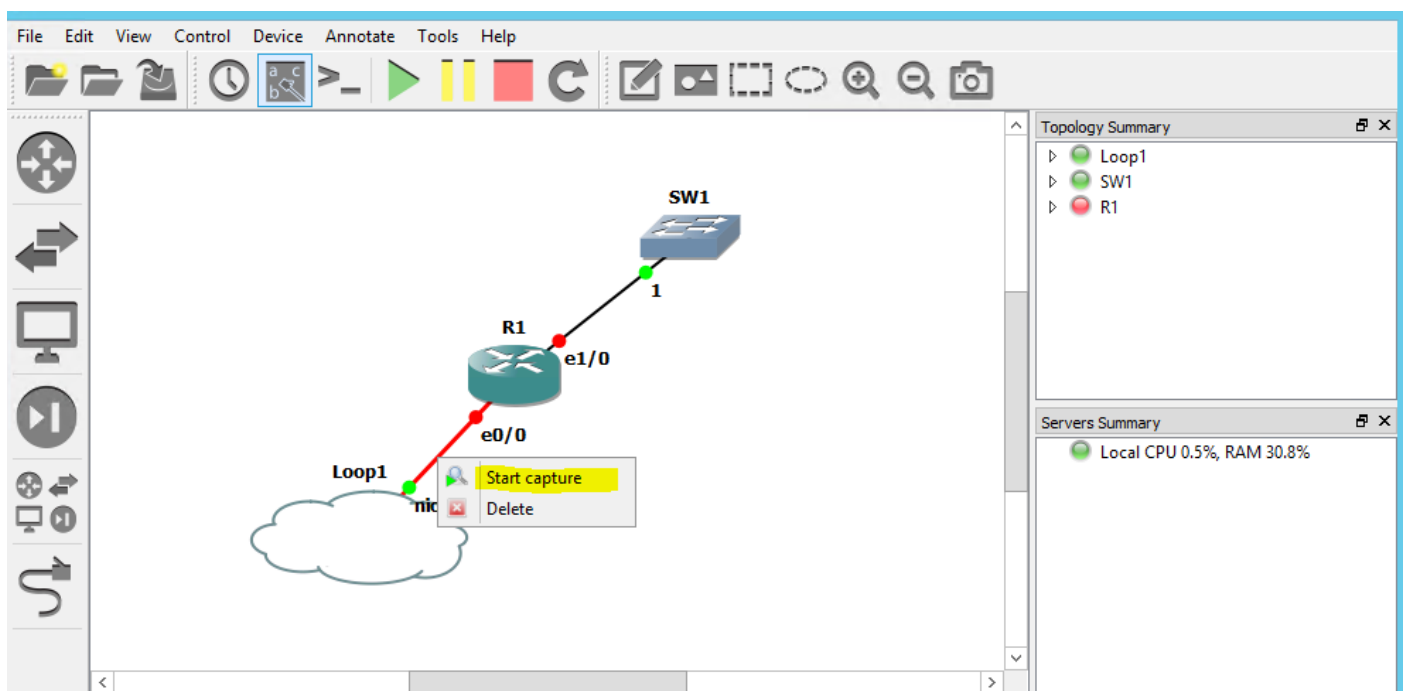
1. Run **GNS3** as an **administrator**.

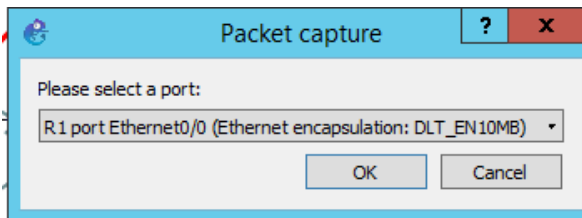


2. Open the GNS3 project **NET311-Lab-05.gns3**

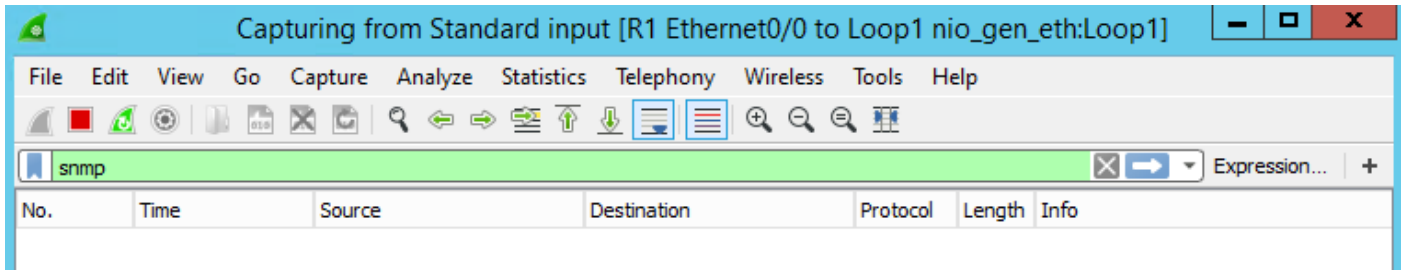


3. Right-click on the link from **Loop1** to **R1** and click **Start capture**.

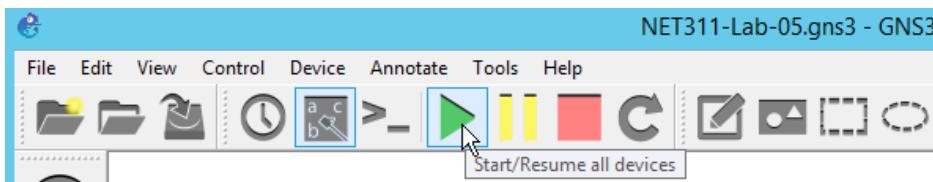




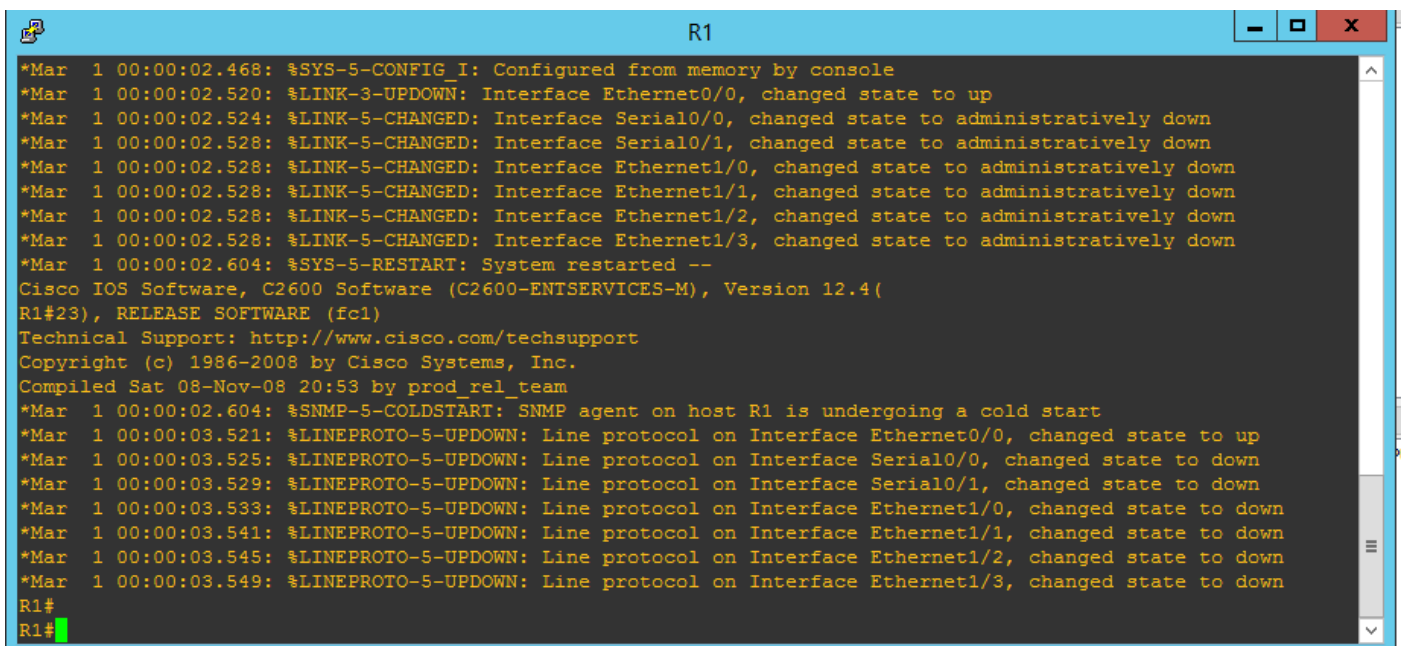
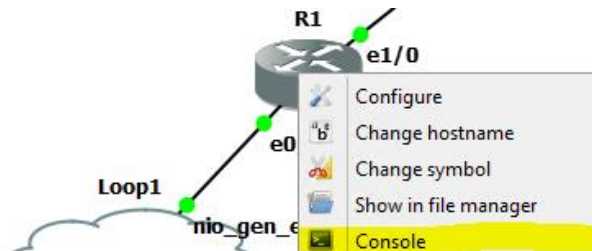
4. Add the filter **snmp** then click **Enter**.



5. Run the network by clicking on the green icon.



6. After the network is started, **right-click** on the **R1** router to access its console.



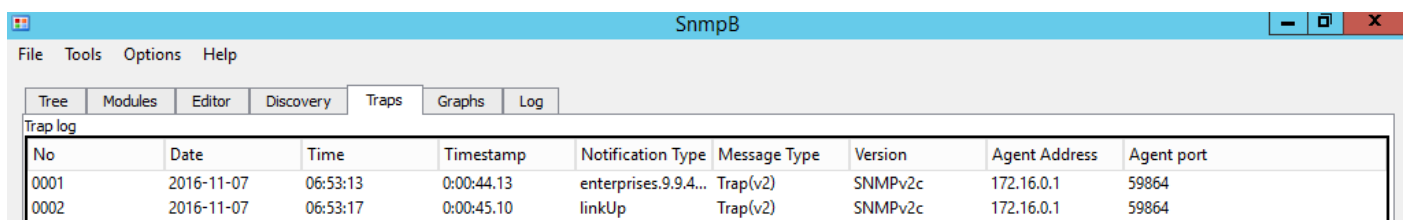
7. Type the following commands to configure the SNMP agent on R1 router. **Use your name** as a contact:

```
config t
snmp-server community public ro SNMP_ACL
snmp-server location Lab4
snmp-server contact Mostafa Dahshan
snmp-server host 172.16.0.2 version 2c public
snmp-server enable traps
ip access-list standard SNMP_ACL
permit 172.16.0.2
exit
```

8. Type the following commands to configure the network interface e0/0.

```
int e0/0
ip address 172.16.0.1 255.255.255.0
no shutdown
```

9. Go to SnmpB and check for traps sent from R1.

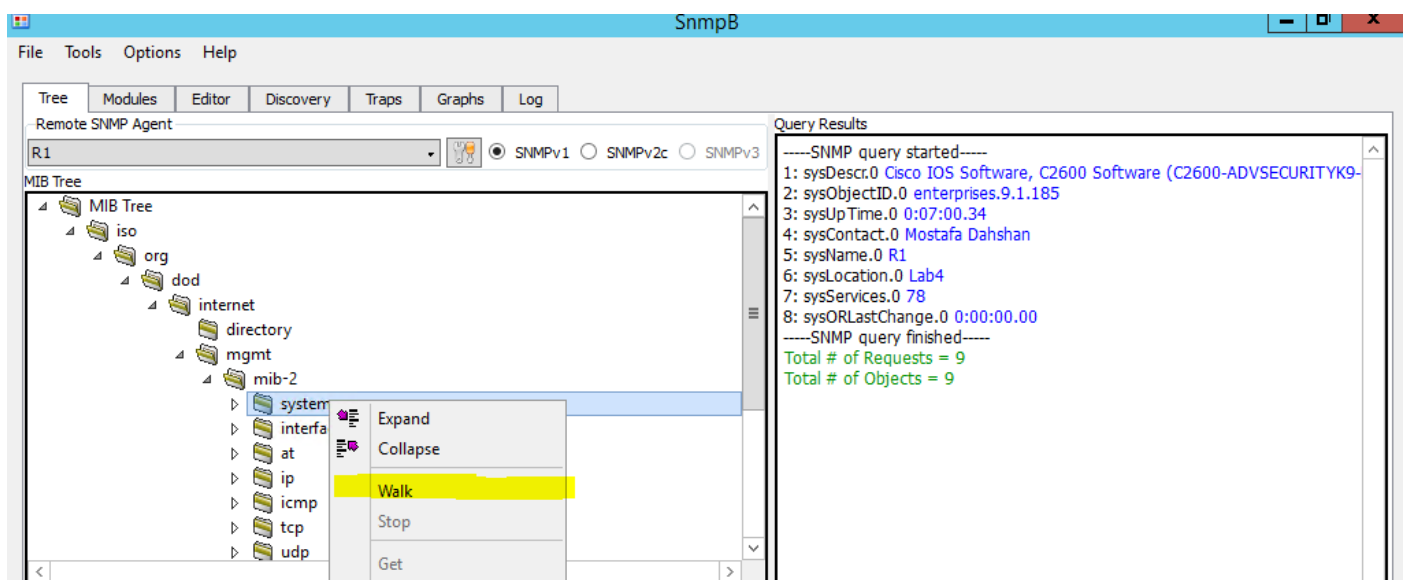


The screenshot shows the SnmpB application window with the 'Traps' tab selected. Below the tabs is a 'Trap log' table with the following data:

No	Date	Time	Timestamp	Notification Type	Message Type	Version	Agent Address	Agent port
0001	2016-11-07	06:53:13	0:00:44.13	enterprises.9.9.4...	Trap(v2)	SNMPv2c	172.16.0.1	59864
0002	2016-11-07	06:53:17	0:00:45.10	linkUp	Trap(v2)	SNMPv2c	172.16.0.1	59864

Lab sheet 3.1: provide a screenshot of the SnmpB Traps screen showing initial traps.

10. Go to SnmpB Tree window and perform a Walk on system using SnmpB profile.

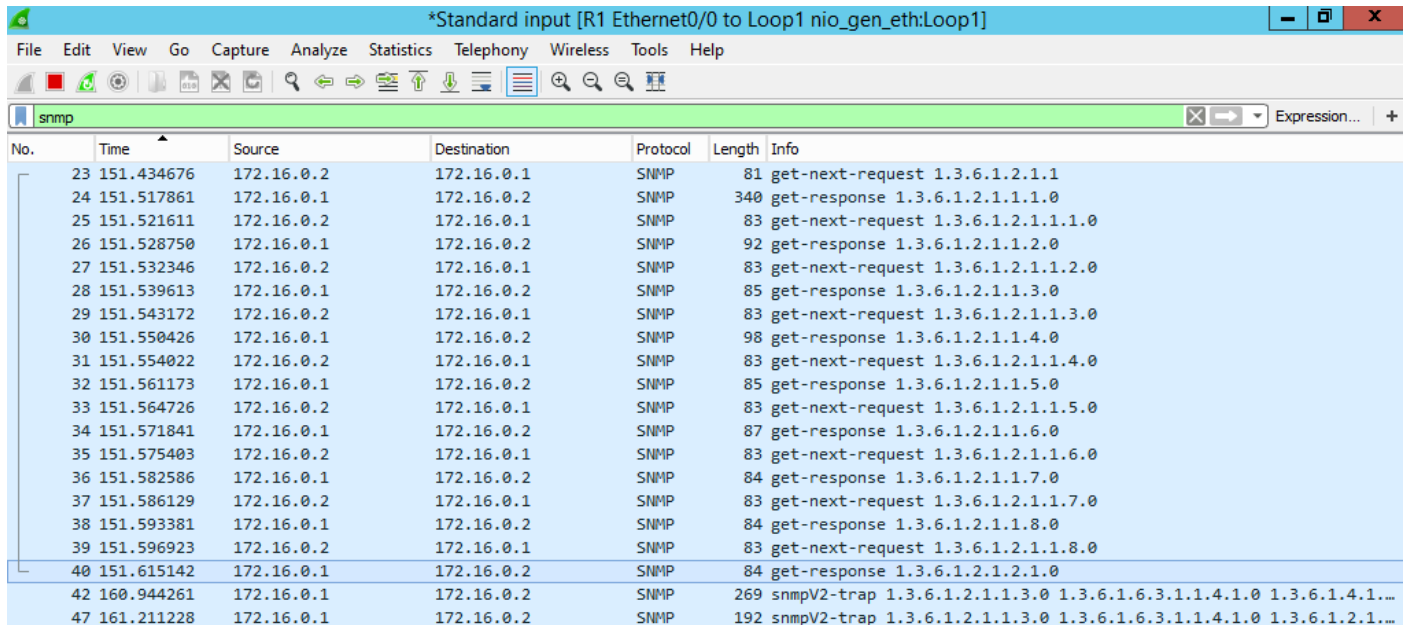


The screenshot shows the SnmpB application window with the 'Tree' tab selected. The 'Remote SNMP Agent' is set to 'R1'. The 'MIB Tree' is expanded to 'system'. A context menu is open over the 'system' node, with the 'Walk' option highlighted. The 'Query Results' pane on the right shows the output of the walk:

```
-----SNMP query started-----
1: sysDescr.0 Cisco IOS Software, C2600 Software (C2600-ADVSECURITYK9-
2: sysObjectID.0 enterprises.9.1.185
3: sysUpTime.0 0:07:00.34
4: sysContact.0 Mostafa Dahshan
5: sysName.0 R1
6: sysLocation.0 Lab4
7: sysServices.0 78
8: sysORLastChange.0 0:00:00.00
-----SNMP query finished-----
Total # of Requests = 9
Total # of Objects = 9
```

Lab sheet 3.2: provide a screenshot of the SnmpB Tree windows showing the result of Walk.

11. Go to Wireshark and check the captured SNMP packets.



No.	Time	Source	Destination	Protocol	Length	Info
23	151.434676	172.16.0.2	172.16.0.1	SNMP	81	get-next-request 1.3.6.1.2.1.1
24	151.517861	172.16.0.1	172.16.0.2	SNMP	340	get-response 1.3.6.1.2.1.1.1.0
25	151.521611	172.16.0.2	172.16.0.1	SNMP	83	get-next-request 1.3.6.1.2.1.1.1.0
26	151.528750	172.16.0.1	172.16.0.2	SNMP	92	get-response 1.3.6.1.2.1.1.2.0
27	151.532346	172.16.0.2	172.16.0.1	SNMP	83	get-next-request 1.3.6.1.2.1.1.2.0
28	151.539613	172.16.0.1	172.16.0.2	SNMP	85	get-response 1.3.6.1.2.1.1.3.0
29	151.543172	172.16.0.2	172.16.0.1	SNMP	83	get-next-request 1.3.6.1.2.1.1.3.0
30	151.550426	172.16.0.1	172.16.0.2	SNMP	98	get-response 1.3.6.1.2.1.1.4.0
31	151.554022	172.16.0.2	172.16.0.1	SNMP	83	get-next-request 1.3.6.1.2.1.1.4.0
32	151.561173	172.16.0.1	172.16.0.2	SNMP	85	get-response 1.3.6.1.2.1.1.5.0
33	151.564726	172.16.0.2	172.16.0.1	SNMP	83	get-next-request 1.3.6.1.2.1.1.5.0
34	151.571841	172.16.0.1	172.16.0.2	SNMP	87	get-response 1.3.6.1.2.1.1.6.0
35	151.575403	172.16.0.2	172.16.0.1	SNMP	83	get-next-request 1.3.6.1.2.1.1.6.0
36	151.582586	172.16.0.1	172.16.0.2	SNMP	84	get-response 1.3.6.1.2.1.1.7.0
37	151.586129	172.16.0.2	172.16.0.1	SNMP	83	get-next-request 1.3.6.1.2.1.1.7.0
38	151.593381	172.16.0.1	172.16.0.2	SNMP	84	get-response 1.3.6.1.2.1.1.8.0
39	151.596923	172.16.0.2	172.16.0.1	SNMP	83	get-next-request 1.3.6.1.2.1.1.8.0
40	151.615142	172.16.0.1	172.16.0.2	SNMP	84	get-response 1.3.6.1.2.1.2.1.0
42	160.944261	172.16.0.1	172.16.0.2	SNMP	269	snmpV2-trap 1.3.6.1.2.1.1.3.0 1.3.6.1.6.3.1.1.4.1.0 1.3.6.1.4.1...
47	161.211228	172.16.0.1	172.16.0.2	SNMP	192	snmpV2-trap 1.3.6.1.2.1.1.3.0 1.3.6.1.6.3.1.1.4.1.0 1.3.6.1.2.1...

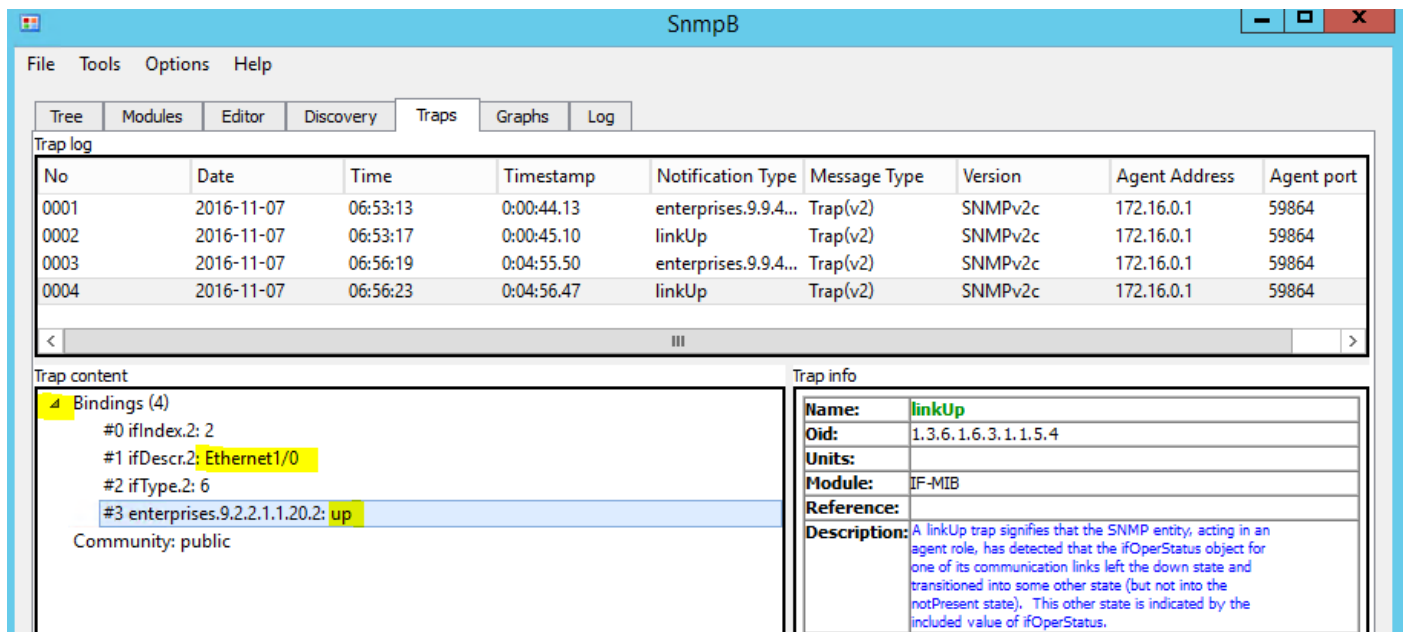
Lab sheet 3.3: provide a screenshot of the Wireshark window showing captured SNMP packets.

To experiment with traps, change the status of the network interface **e1/0** and watch for the trap in SnmpB and Wireshark.

12. Go to R1 console in GNS3 and run the following commands:

```
int e1/0
no shutdown
```

13. Go to SnmpB Traps window and look for the **LinkUp** trap received from R1.



No	Date	Time	Timestamp	Notification Type	Message Type	Version	Agent Address	Agent port
0001	2016-11-07	06:53:13	0:00:44.13	enterprises.9.9.4...	Trap(v2)	SNMPv2c	172.16.0.1	59864
0002	2016-11-07	06:53:17	0:00:45.10	linkUp	Trap(v2)	SNMPv2c	172.16.0.1	59864
0003	2016-11-07	06:56:19	0:04:55.50	enterprises.9.9.4...	Trap(v2)	SNMPv2c	172.16.0.1	59864
0004	2016-11-07	06:56:23	0:04:56.47	linkUp	Trap(v2)	SNMPv2c	172.16.0.1	59864

Trap content		Trap info	
Bindings (4) #0 ifIndex.2: 2 #1 ifDescr.2: Ethernet1/0 #2 ifType.6 #3 enterprises.9.2.2.1.1.20.2: up Community: public		Name: linkUp Oid: 1.3.6.1.6.3.1.1.5.4 Units: Module: IF-MIB Reference: Description: A linkUp trap signifies that the SNMP entity, acting in an agent role, has detected that the ifOperStatus object for one of its communication links left the down state and transitioned into some other state (but not into the notPresent state). This other state is indicated by the included value of ifOperStatus.	

Lab sheet 3.4: provide a screenshot of the SnmpB Traps screen showing the Bindings in the LinkUp trap.

14. Go to R1 console in GNS3 and run the following commands:

```
int e1/0
shutdown
```

15. Go to SnmpB Traps window and look for the **LinkDown** trap received from R1.

The screenshot shows the SnmpB application window. The 'Traps' tab is selected, displaying a table of trap logs. The table has columns: No, Date, Time, Timestamp, Notification Type, Message Type, Version, Agent Address, and Agent port. The last entry (No. 0005) is a 'linkDown' trap received from 172.16.0.1 at 06:58:31 on 2016-11-07. Below the table, the 'Trap content' section shows bindings for the selected trap, including 'Ethernet1/0' and 'administratively down'. The 'Trap info' section on the right provides details about the 'linkDown' trap, including its OID, units, module, and a description.

No	Date	Time	Timestamp	Notification Type	Message Type	Version	Agent Address	Agent port
0001	2016-11-07	06:53:13	0:00:44.13	enterprises.9.9.4...	Trap(v2)	SNMPv2c	172.16.0.1	59864
0002	2016-11-07	06:53:17	0:00:45.10	linkUp	Trap(v2)	SNMPv2c	172.16.0.1	59864
0003	2016-11-07	06:56:19	0:04:55.50	enterprises.9.9.4...	Trap(v2)	SNMPv2c	172.16.0.1	59864
0004	2016-11-07	06:56:23	0:04:56.47	linkUp	Trap(v2)	SNMPv2c	172.16.0.1	59864
0005	2016-11-07	06:58:31	0:06:29.76	linkDown	Trap(v2)	SNMPv2c	172.16.0.1	59864

Trap content

- Bindings (4)
 - #0 ifIndex.2: 2
 - #1 ifDescr.2: Ethernet1/0
 - #2 ifType.2: 6
 - #3 enterprises.9.2.2.1.1.20.2: administratively down
- Community: public

Trap info

- Name: linkDown
- Oid: 1.3.6.1.6.3.1.1.5.3
- Units:
- Module: IF-MIB
- Reference:
- Description: A linkDown trap signifies that the SNMP entity, acting in an agent role, has detected that the ifOperStatus object for one of its communication links is about to enter the down state from some other state (but not from the notPresent state). This other state is indicated by the included value of ifOperStatus.

Lab sheet 3.5: provide a screenshot of the SnmpB Traps screen showing the Bindings in the LinkDown trap.

16. Go to Wireshark window and double click on the last captured SNMP trap.

Wireshark · Packet 121 · wireshark_pcapng_-_20161107065035_a03448

- ▷ Ethernet II, Src: c8:01:01:84:00:00 (c8:01:01:84:00:00), Dst: 02:00:4c:4f:4f:50 (02:00:4c:4f:4f:50)
- ▷ Internet Protocol Version 4, Src: 172.16.0.1, Dst: 172.16.0.2
- ▷ User Datagram Protocol, Src Port: 59864 (59864), Dst Port: 162 (162)
- ▣ Simple Network Management Protocol
 - version: v2c (1)
 - community: public
 - ▣ data: snmpV2-trap (7)
 - ▣ snmpV2-trap
 - request-id: 5
 - error-status: noError (0)
 - error-index: 0
 - ▣ variable-bindings: 6 items
 - ▣ 1.3.6.1.2.1.1.3.0: 38976
 - Object Name: 1.3.6.1.2.1.1.3.0 (iso.3.6.1.2.1.1.3.0)
 - Value (Timeticks): 38976
 - ▣ 1.3.6.1.6.3.1.1.4.1.0: 1.3.6.1.6.3.1.1.5.3 (iso.3.6.1.6.3.1.1.5.3)
 - Object Name: 1.3.6.1.6.3.1.1.4.1.0 (iso.3.6.1.6.3.1.1.4.1.0)
 - Value (OID): 1.3.6.1.6.3.1.1.5.3 (iso.3.6.1.6.3.1.1.5.3)
 - ▣ 1.3.6.1.2.1.2.2.1.1.2:
 - Object Name: 1.3.6.1.2.1.2.2.1.1.2 (iso.3.6.1.2.1.2.2.1.1.2)
 - Value (Integer32): 2
 - ▣ 1.3.6.1.2.1.2.2.1.2.2: 45746865726e6574312f30
 - Object Name: 1.3.6.1.2.1.2.2.1.2.2 (iso.3.6.1.2.1.2.2.1.2.2)
 - Value (OctetString): 45746865726e6574312f30
 - ▣ 1.3.6.1.2.1.2.2.1.3.2:
 - Object Name: 1.3.6.1.2.1.2.2.1.3.2 (iso.3.6.1.2.1.2.2.1.3.2)
 - Value (Integer32): 6
 - ▷ 1.3.6.1.4.1.9.2.2.1.1.20.2: 61646d696e6973747261746976656c7920646f7776e

No.: 121 · Time: 479.395536 · Source: 172.16.0.1 · Destination: 172.16.0.2 · Protocol: SNMP · Length: 1.3.6.1.2.1.2.2.1.1.2 1.3.6.1.2.1.2.2.1.2.2 1.3.6.1.2.1.2.2.1.3.2 1.3.6.1.4.1.9.2.2.1.1.20.2

Close Help