TECHNISCHE UNIVERSITÄT BERLIN
Fakultät IV – Elektrotechnik und Informatik
Fachgebiet Intelligente Netze und Management verteilter Systeme
Prof. Anja Feldmann, Ph.D.
A. Mehmood, T. Hühn, L. Iannone, W. Mühlbauer, G. Schaffrath

Praktikum RouterLab SS 2009
Work Sheet 10: Workload Generation and Monitoring using Netflow/SNMP

The World Wide Web (WWW) is the most essential component of today's Internet. Users generate requests and web-servers provide the desired contents. Therefore it is necessary for any TEST-LAB setup to generate web-traffic which possesses similar characteristics as those observed in the real Internet. Examples of such characteristics are inter-arrival times, distribution of server file sizes, number of concurrent sessions, etc. This process of artificially generating web-traffic is known as web-workload generation. The purpose of this worksheet is twofold: First we learn how to generate web-traffic using different tools for experimentation purposes. Second we will learn an additional method of monitoring traffic using *flows*. By defination *'A flow is an aggregation of similar packets that are close in time'*. The most common type of flows are Five-Tuple Flows which is the result of aggregating packets with the same source/destination IP address and port number and by the transport protocol: (SRCIP, SRCPORT,DSTIP,DSTPORT,PROTOCOL). Note that according to this definition flows are uni-directional! A flow ends when there has not been a matching packet for a certain time called inactivity timeout (usually 15 seconds). In the case of TCP, a flow can also be terminated by a FIN or a RST packet. Flows carry with them information like start and end time of a flow or number of bytes and number of packets contained in a flow. For this purpose we will use *Netflow Version 5* on Cisco Router. Furthermore, we will learn how to monitor traffic using tools such as SNMP.

Table 1: Assignment of devices to groups

Group	Ham-Cloud	Muc-Cloud
Router	ham-rc1	muc-rc1
Switches	ham-sc1, ham-sc2	muc-sc1, muc-sc2
Loadgens	loadgen10[2/3/4]-ham	loadgen10[2/3/4]-muc
IP range	10.1.0.0/16	10.2.0.0/16

Question 1: (20 Points) *Basic Setup Configuration*

In this question, we will build the basic setup for the generation of web workload.

(a) First of all you need to establish connectivity between your web-client and web-server as shown in Figure 1. Your configuration must satisfy the following conditions:

1. Configure VLANs as shown in Figure 1.
2. Both client and server as shown in Figure 1 must ping each other under the condition that there is no (!) overlap in IP address range used for the links between i) web-client and rc1. ii) rc1 and monitoring Loadgen. iii) rc1 and web server.
3. You should be able to do traceroute from web client to web-server through rc1.

(b) Submit a logical topology map that shows your IP address assignment.

Question 2: (20 Points) *Workload Generation with iperf*

Frequently, it is desired to test the "quality" of a link or path over which packets are sent. For this purpose *iperf* can be used.

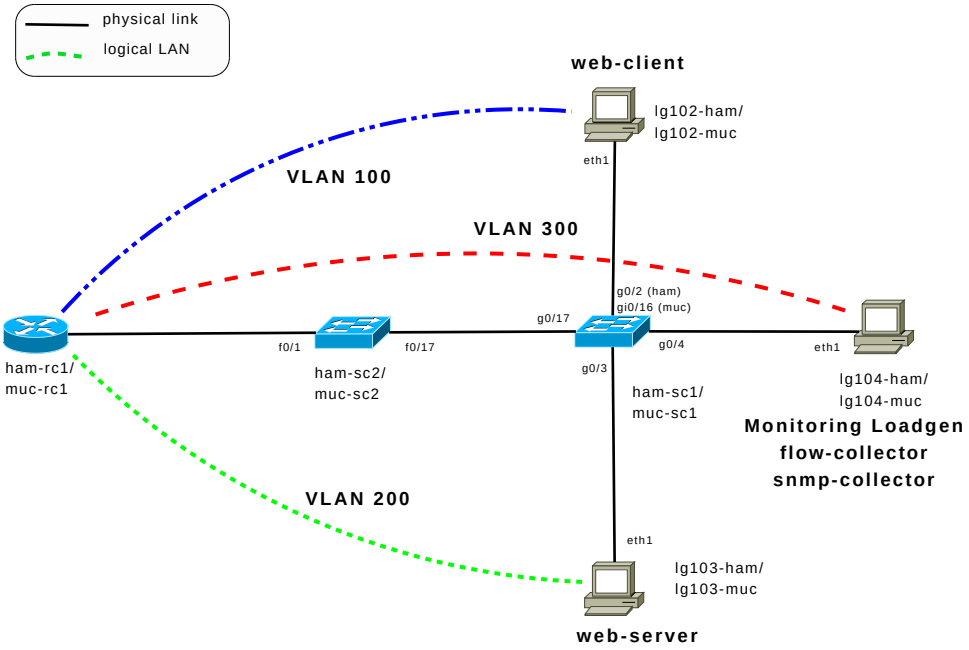

Figure 1: Topology – Basic Configuration for Web Workload gereration

(a) Install *iperf* using apt-get install iperf on both web-client and web-server[1]. Read the man page of *iperf* and find out how it can be used in client and server mode. First start one *iperf* process as server on the web-client *loadgen* and then start another process on the same web-client loadgen. Observe the following parameters for multiple iterations.

- TCP throughput/bandwidth.
- Maximum segment size (MSS)
- TCP window size

(b) Now start one *iperf* process on web-client as client and one *iperf* process on web-server as server and observe the same parameters as in Question 2a) for multiple iterations.
Submit a table for at least five iterations showing results of Question 2a) and 2b).

(c) Did you observe differences in the values that you obtain in Question 2a) and Question 2b).Explain which parameters are changing. What are the possible factors which are causing these changes?

(d) Can *iperf* perfrom bidirectional bandwidth measurement? If yes how?

Question 3: (20 Points) *Web work load Generation with Harpoon*

(a) In order to monitor traffic and to find out how much bandwidth is currently being used, you need tools such as *mrtg* or *iptraf*. For this, we will use only *iptraf*. Install *iptraf* on your loadgens using apt-get install iptraf. Read man page of *iptraf* and find out how it can be used.

(b) *Harpoon* is a web-work load traffic generator which is used for realistic web traffic generation in a lab environment. In this question you will learn how to generate traffic in greater volume. We have provided you a compiled version of *Harpoon* that you can find at */usr/local/harpoon* of your loadgen image. There you also find a folder *RouterLab* with the two files *web-client.xml* and *web-server.xml* which will be used for traffic generation. Analyse these files carefully and explain in short what 'active sessions', 'file sizes' and 'interconnection times' means.

(c) Read the manual of Harpoon at `(http://pages.cs.wisc.edu/~jsommers/harpoon/)` and find out what options are avaiable.

(d) For generating default traffic with Harpoon, you do not need to modify 'web-server.xml' file but you need to modify the following lines at the end of 'web-client.xml' file according to your IP assignment.

[1] if you get some errors while installing iperf/iptraf, ignore them

```
<address_pool name="client_source_pool">
    <address ipv4="x.x.x.x/32" port="0" />
</address_pool>

<address_pool name="client_destination_pool">
    <address ipv4="y.y.y.y/32" port="10000" />
</address_pool>
```

Here x.x.x.x/32 is the web client IP address and y.y.y.y/32 is the web server IP address.

First start Harpoon server process on web server *loadgen* using the following command:

```
./harpoon -f RouterLab/web-server.xml -v10 -w300
```

Use *netstat -an* to confirm that your web server is now in 'Listening mode'. Then start the Harpoon client process on web client *loadgen* using the following command:

```
./harpoon -f RouterLab/web-client.xml -v10 -w300 -c
```

You can stop any Harpoon process using ctrl-c. Use iptraf and observe how much traffic is being generated in both directions. Also observe which IP addresses and ports are used for web-traffic. Use the following commands on rc1 and sc1 to observe traffic.

```
show interfaces <interface> | include bits
```

By default interface statistics are updated every 5 minutes. You should change this interval to 30 seconds on all interfaces that you are monitoring using the following command in interface configuration mode.

```
load-interval 30
```

Submit the web traffic data rate you observe with these default Harpoon files in both directions.

(e) Find out how different web traffic data rates can be generated by reading the Harpoon documentation. Try different iterations[2] and find out what maximum throughput you can achieve over the setup of Figure 1. Submit your web-client.xml configurations[3] you use to achieve maximum throughput.

Question 4: (20 Points) *Configuring Netflow on Cisco Router*

You have already used tcpdump many times for packet-level capturing. Now you will learn how to capture aggregated data from routers in the form of flows.

(a) Configure *Netflow Version 5* on rc1 with the following commands. You need to configure the IP address 100.100.100.100/32 on interface loopback 0. Choose port 7777 as destination port for exporting flows to the flow-collector loadgen.

- At global configuration mode:

```
ip flow-export source Loopback0
ip flow-export destination <destination_ip>  <destination_port>
```

- At interface configuration mode on interface from where you want to get flows:

```
interface <sub-interface>
ip flow ingress
```

To verify that flows are exported correctly use *sh ip flow export* command.

(b) You need a tool which can capture flow data. For this purpose you can use *flow-tools*[4] which is a software package for collecting and processing NetFlow data from Cisco and Juniper routers. For this worksheet you need only *flow-capture* for collecting flows from rc1 router and *flow-report* which generate reports from flow data. Install *flow-tools* using apt-get install flow-tools on your monitoring/flow-collector loadgen. Read the man pages of different flow-tools utilities and capture the flows in */tmp/flows* directory of your monitoring loadgen for at least five minutes of web traffic using the Harpoon web workload generator as in Question 3. Now generate a flow report[5] from these files. Explain what you see in the flow report. Submit the output of flow-report along with the commands used for generating flow-report.

(c) Submit graphs(no text files) for 'Packets per flow distribution', 'Octets/Bytes per flow distribution' and 'Flow Time distribution' from the data you obtain in the flow-report with

[2] You need to modify web-client.xml only

[3] Submit only the parameters you change not the whole file

[4] (http://www.splintered.net/sw/flow-tools/)

[5] Use flow-cat along with flow-report

1. default Harpoon configuration.
2. Harpoon configuration with maximum throughput obtained in Question 3e).

Question 5: (20 Points) *Network Management with SNMP*

So far we have learned how to get aggregated information from the router in the form of netflow and now we will learn how to get detailed information from the routers and switches using SNMP.

The management protocol SNMP — Info 10.1

SNMP is the protocol most commonly used to exchange management information between components on the Internet.

The communication model follows the Client-Server paradigm. The following parties participate in the protocol: One *Manager* (oder *Management Station*) and several *Management Agents*, which are located inside of the monitored component (also called *Managed Node*). The Manager acts as a client and requests pieces of information from the Agent (acting as the server), using the *get-request*. Alternatively, it can also transfer information there (*set-reqest*). Access to the information is protected using a simple shared secret, the *community string*

The information is structured based on an information model: the *Structure of Managed Information, SMI*. According to the SMI, the agents in the managed nodes have a characteristic *Management Information Base (MIB)*, a structure of variables. The Agent-MIB represents a subtree of the standardized Internet-MIB tree.

Info 1 provides an overview of network management using SNMP. Accomodate yourself in the following questions about any necessary details on the SNMP management, and about the invocation and functionality of the tools involved.

In this question, we take the first steps with the protocol SNMP and the associated tools, and we examine the information and communication model.

(a) The command line programs `snmpget`, `snmpset` and `snmpwalk` (All present on the loadgens) are useful tools to perform simple management tasks. Read up on their functionality and basic invocation syntax in the `man` pages. Hint: Use the option `-v1` the avoid conflicts with different versions of SNMP, and use `-Of` to always output full OIDs.

(b) Enable SNMP on the [ham—muc]-rc1 router. Some hints on the required communication commands can be found in Info 2.

(c) Now request the value of the Agent MIB object with the OID `iso.org.dod.internet.mgmt.mib-2.system.sysUpTime.0`.

Enabling snmp on Cisco Devices — Info 10.2

On Cisco routers, the commands to activate and configure an SNMP agent in configuration mode are initiated with `snmp-server`.

```
snmp-server community <your_secret_community_string> RO
```

(d) SNMP is based on UDP. What consequences does this have for different error situations? E.g., what behaviour did you observe when there was no SNMP agent running on the router? What happens in case of packet loss? What happens if you pass a wrong community string?

(e) Analyze the SNMP packets using `Wireshark`. What pieces of information are transmitted in the packets, which protocol fields are present?
How does the OID look? Does `snmpget` show it in the same way? Can you also provide the numeric OID to `snmpget`?

Include your tracefile with your answer.

(f) Set the router MIB variables for *system contact* and *system location* to arbitrary values.
Do an `snmpwalk` on `iso.org.dod.internet.mgmt.mib-2.system`.
Capture the packet exchange, like in exercise (e). What types of request do you see now? The same ones as before? Explain how *snmpwalk* works!

For submission details please check the FAQ:
`http://www.net.t-labs.tu-berlin.de/teaching/ss09/RL_labcourse/`
Due Date: Friday, July 10th, 2009, 08:00 am