Project Report

MRTG Installation, Implementation and Configuration

Prepared by: Parvez Ibrahim

Submitted to: M.H. MacGregor, PhD, PEng, SMIEEE

Associate Professor

Director, MSc in Internetworking

Department of Computing Science

Table of Contents

Acknowledgement

I take pleasure in thanking Dr. Mike MacGregor, for supervising this project. He helped us not only as the supervisor of our project, but also with many other supports to make sure proper progress of the project. During many discussions I had with him, I always felt that I got in to the correct track.

I would also like to thank Dr. Mike MacGregor for all his time and resources he provided us for this project. Without his help and support I think we were not able to complete this project. He is very friendly and kind person.

Introduction and Background

This project is based on MRTG application. Objectives of this project are to Study functions and features of MRTG Application, Implementation and configuration MRTG in the Mint Lab. Setup network scenario in Mint Lab. Monitor network traffic load through MRTG application, Setup Media Server, Proxy Server and FTP Server (video streaming) in the scenario and monitor the network traffic through MRTG and submit the report.

The project begins with the study of MRTG application and it requirements for installation and implementation. A Network was designed, configured and implemented. Network of six routers has been setup and OSPF protocol selected to configure as a communication protocols. Network is divided into three areas. DR and BDR have been elected. Simple Network Management Protocol (SNMP) is also configured on all router for monitoring the traffic and current situation of routers. The Simple Network Management Protocol (SNMP) is an application layer protocol that facilitates the exchange of management information between network devices. It is a part of the Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite.

Four types of Servers have been setup and configured for traffic monitoring and to check load on routers these Servers are Media Server, File Server, Proxy Server and MRTG Server for traffic monitoring. For media server VLC, CuteFTP and WinFTP for file Server and WinGate Server for Proxy. Traffic has been monitored on different network. All the traffic (video streaming, file server, proxy server) was sent through one network and then traffic was distributed to different networks.

Multi Router Traffic Grapher (MRTG) is a tool to monitor the traffic load on network links was downloaded from internet, version mrtg-2.15.2 (latest) is used in our project. We have captured in and out, TCP and UDP packets for which script mrtg.cfg has been created. MRTG.CFG is the basic script file for MRTG which only capture in and out traffic. We have researched what OIDs are required to captured packets for TCP and UDP. After monitoring traffic by MRTG for several weeks we have also explored RRDTOOL which is more user friendly and the graphical images are very good but this tool is difficult to configure than MRTG. To capture data through RRDTOOL we have selected two CGI scripts 14all and routers2 which are also downloaded from the internet and amended as per requirement.

Finally we have installed MRTG/RRD Tool in Norquest college for more than month we have monitored traffic load two weeks one firewall and two weeks on fortigate router. Fortigate also used for the remote access which Norquest used from home mostly after office hours.

We have captured different type of traffic through MRTG. For that we have used different OIDs by editing MRTG.cfg. To monitor traffic load on different routers interfaces we have developed 7 to 8 configuration files these files were developed differently.

At the end of the report we are presenting MRTG result and conclusion. In the MRTG results number of MRTG graph including traffic in/out, daily, weekly, monthly, UDP and TCP are showed.

Chapter No. 1 MRTG and RRDTOOL

Conventional Systems of Data Collection and Presentation

If we see in past, analyzing data was very challenging work. The network analysts use telnet to a router and execute commands that can give them information they need, the information showed in the text format and the network analysts have to compare the data with the other data to analyze the network traffic load which is in no way accurate and efficient system.

Presently there are some tools to analyze the network activities graphically, which is also easily readable format but we can also compare the data with other routers data on the network graphically. This is very important in network analysis. This allows us to not only tell how much traffic is flowing, but where it is being sent on the network. We can also determine the source of bandwidth bottlenecks and use this information, For example, if we see a graph with a lot of traffic flowing out of one router, we can see the same traffic flowing into the other routers on the network, just by glancing at the graphs. This cannot be easily accomplished by the raw text console method.

MRTG and RRDtool

Our assignment for the project is to find a way to connect to the routers, pull out the numbers in the counters, and graph them over time. Before starting the project, we went online and researched existing software packages that are designed to do this. There are many tools available on internet but there are two that stand out above the rest. They do the best job, and have been the most widely used and documented. They are the Multi-Router Traffic Grapher (MRTG) and the Round-Robin Database Tool (RRDtool).

Multi-Router Traffic Grapher

Multi-Router Traffic Grapher is a tool that collects, stores, and graphs data on a given interval. SNMP which reads the traffic counters from network device like router and then it is C program logs the traffic data and creates graphs representing the traffic in and out on the network. MRTG consists of a Perl script. These graphs are embedded into web pages, which can be viewed from any internet browser. See Figure A below is an example of an MRTG graph. In the graph blue is for bytes out and green for bytes in.

Traffic Analysis for 1 – MINT LAB -- routerD

System: routerD in
Maintainer:
Description: FastEthernet0/0
ifType: ethernetCsmacd (6)
ifName: Fa0/0
Max Speed: 12.5 MBytes/s
Ip: 10.1.32.3 ()

The statistics were last updated Friday, 7 September 2007 at 21:29,
at which time 'routerD' had been up for 18 days, 3:41:40.

`Daily' Graph (5 Minute Average)

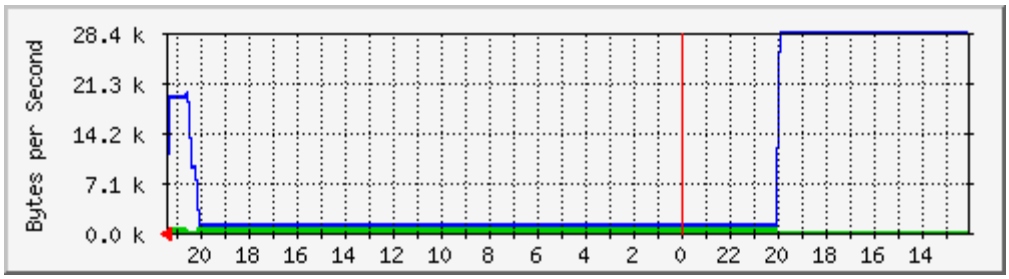


Fig A.

MRTG

Features and Issues

One of the MRTG feature is its configuration file. We can generate configure file manually this means that we can make our own configuration file in notepad or any other text editor or auto-generated by executing cfgmaker file but cfgmaker generates only a basic configuration file for advance level configuration file we need to edit that auto-generated file manually and add more script in it. MRTG's data files do not grow. We can edit the configuration file when ever we want. Daily, weekly and monthly graphs can easily been seen graphically. There some issues that involved with MRTG.

One of the issues with MRTG is that it creates all the graphs every time it runs. In our case, we are graphing number of interfaces on four to six routers. This results a large CPU utilization every 5 minutes. Another issue with MRTG is little flexibility and few customization options. On the other hand RRDTool has nice graphical interface and more flexibility as compare to MRTG.

Round-Robin Database Tool (RRDtool)

RRDtool is similar kind of tool as MRTG. We can also say it is an extension of MRTG's capabilities. RRDtool is not a replacement for MRTG, as RRDTool cannot implement the front end and data acquisition features of MRTG. MRTG can be configured in a way that RRDtool is used as its database which is used at backend and at the frontend we can use CGI script. Graphing with RRDtool is very flexible. Generated graphs can contain any and all information one can require. The RRDtool graphs are similar to the MRTG graphs, except that the RRDtool graphs contain more information. See fig B.

Top Router MINT Lab--

System: Top Router

Maintainer:

Description: FastEthernet 0/0

ifType: FastEthernet 0/0 (71)

Max Speed: 12500000 kBytes/s

Ip: 192.168.0.2 (par-pc.mshome.net)

The statistics were last updated: **Tue Nov 6 22:15:24 2007**

'Weekly' graph (30 Minute Average)

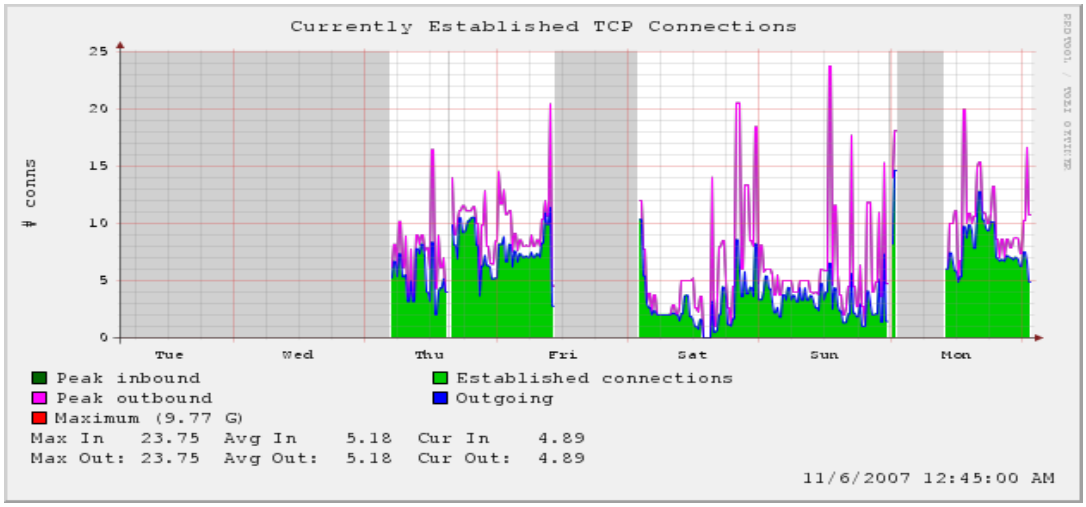


Fig B Graph generated by RRDtool.

RRDtool

RRDtool provide better graphical interface. RRDtool has many features and capabilities as compare to MRTG and we have found RRTool more complex than MRTG to install and configure. The data storage algorithms described below is much more efficient, yet much harder to grasp than those of MRTG. The following is taken from the RRDtool website.

When monitoring the state of a system, it is convenient to have the data available at a constant interval. Unfortunately we may not always be able to fetch data at exactly the time we want to. Therefore RRDtool lets us update the log file at any time we want. It will automatically interpolate the value of the data-source at the latest official time-slot and write this value to the log.

Logging data over a 3 minute interval, but if we want to know the development of the data over the last few hours, the last week, or the last month. RRDtool offers a solution to this problem through its data consolidation feature. When setting up a Round Robin Database (RRD), we can define at which interval this consolidation should occur. There can be multiple consolidation functions for each RRD and they will all be maintained when new data is loaded into the database.

Data values of the same consolidation setup are stored into Round Robin Archives (RRA). This is a very efficient manner to store data for a certain amount of time, while using a known amount of storage space. The use of RRAs guarantees that the RRD does not grow over time and that old data is automatically eliminated. By using the consolidation feature, we can still keep data for a very long time, while gradually reducing the resolution of the data along the time axis.

RRDcgi

One of the most useful features of RRDtool is its RRDcgi module.

RRDcgi is a web scripting module that eliminates the creation of all the graphs on every step of the interval. It is one of the most useful features of the RRDTool. RRDcgi is embedded into a web page and makes a call to RRDtool to graph only the graphs that are requested by the web page. So RRDtool only generates the graphs, when you want to see them. This makes RRDtool much more efficient than MRTG.

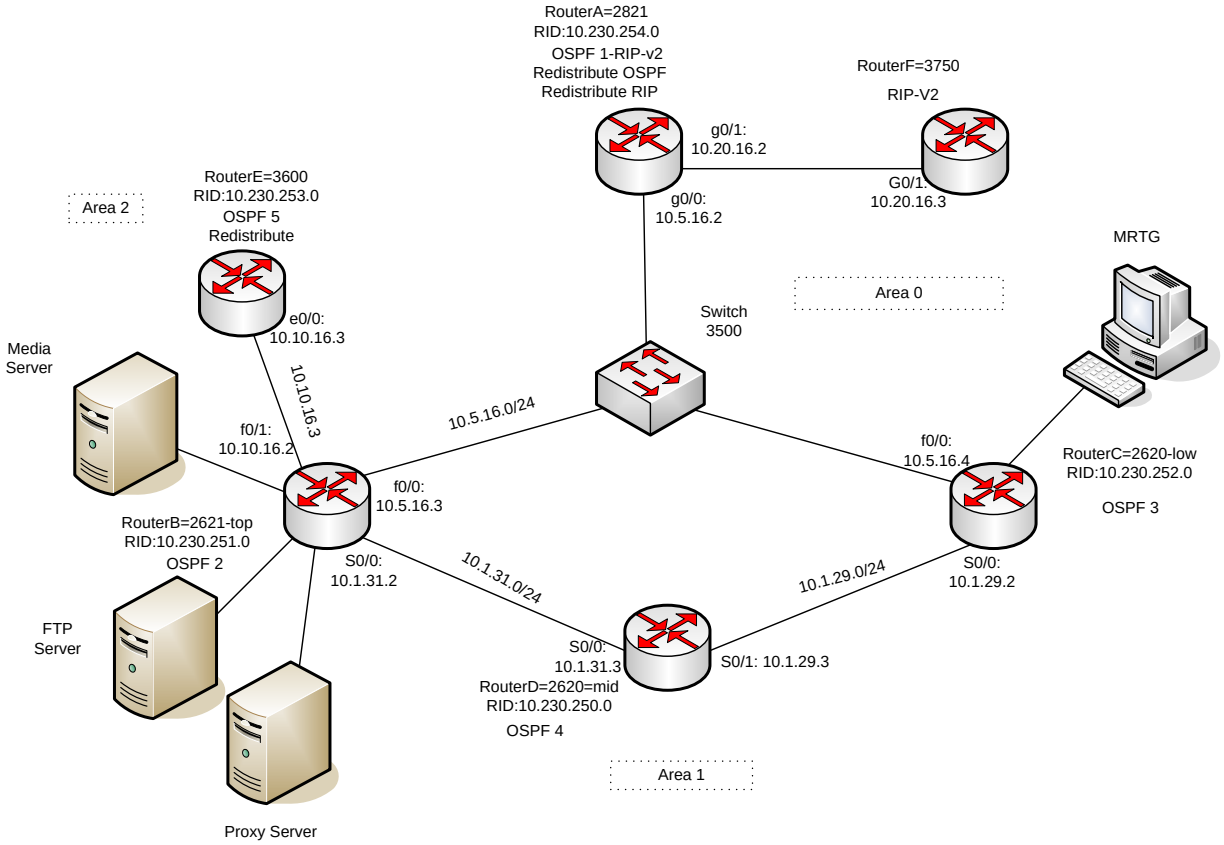
Chapter No. 2 Network and OSPF Configuration

Network:

We have design and implement network of several routers. The protocol configured on this network is OSPF. Network is divided into three areas. The reason of dividing network into 3 areas is that router within an area maintain a database for the area to which it belongs. The router doesn't have detailed information about network topology outside of its area that is why the size of its database is not large. On this network router B elected as DR and router C is BDR. On this network we have also setup one switch. Router A, router B and routers C are directly connected to the switch. Router D is directly connected to router B and router C. Router E is connected to router B and router F is connected to router A. Router A and router E are AS in this network.

For traffic flow we have setup three servers FTP, media server and Proxy server on one router B via switch and on the other router we have setup MRTG on one computer which is connected to the router C. Then we started traffic on the network.

Six routers and one switch have been setup and configured in MINT lab. Following fig C is showing networks created in MINT lab. IP address is assigned to each router port as shown in fig C.



After configuration of routers we have setup also Server to start traffic. We have setup Media Server, FTP Server, Proxy Server under Windows 2003 Server platform. All three servers are connected to switch and on the same network of router B 10.5.16.0. This will help us to send type of type of traffic on the network to check traffic load. MRTG monitoring computer also setup at different network so that we can monitor traffic load on the network.

IP address

Complete IP address allocation is given in the table.

RouterD=2620-mid

Type	IP	MASK
Networks	10.1.31.0 10.1.29.0	255.255.255.0 255.255.255.0
interface Serial0/0	10.1.31.3	255.255.255.0
interface Serial0/1	10.1.29.3	255.255.255.0

RouterB=2621-top

Type	IP	MASK
Networks	10.5.16.0 10.1.31.0 10.10.16.0	255.255.255.0 255.255.255.0 255.255.255.0
Interface FastEthernet0/0	10.5.16.3	255.255.255.0
Interface Serial0/0	10.1.31.2	255.255.255.0
Interface FastEthernet0/1	10.10.16.2	255.255.255.0

RouterF=3750

Type	IP	MASK
Network	10.20.16.0	255.255.255.0
interface GigabitEthernet1/0/1 no switch port	10.20.16.3	255.255.255.0

RouterE=3600

Type	IP	MASK
Network	10.10.16.0	255.255.255.0
interface Ethernet0/0	10.10.16.3	255.255.255.0

RouterA=2821-top

Type	IP	MASK
Networks	10.5.16.0 10.20.16.0	255.255.255.0 255.255.255.0
interface GigabitEthernet0/0	10.5.16.2	255.255.255.0
interface GigabitEthernet0/1	10.20.16.2	255.255.255.0

RouterC=2620-low

Type	IP	MASK
Networks	10.5.16.0 10.1.29.0	255.255.255.0 255.255.255.0
interface FastEthernet0/0	10.5.16.4	255.255.255.0
Interface Serial0/0	10.1.29.2	255.255.255.0

In the second step on all routers OSPF protocol is configured as follows.

Router B

Router ospf 2

Log-adjacency-changes

area 1 stub

area 1 range 10.1.16.0 255.255.240.0

network 10.1.31.0 0.0.0.255 area 1

network 10.5.16.0 0.0.0.255 area 0

network 10.10.16.0 0.0.0.255 area 2

Router C

router ospf 3

log-adjacency-changes

area 1 stub no-summary

area 1 range 10.1.16.0 255.255.240.0

network 10.1.29.0 0.0.0.255 area 1

network 10.5.16.0 0.0.0.255 area 0

Router D

router ospf 4

log-adjacency-changes

area 1 stub no-summary

area 1 range 10.1.16.0 255.255.240.0

network 10.1.16.0 0.0.3.255 area 1

network 10.1.20.0 0.0.3.255 area 1

network 10.1.24.0 0.0.3.255 area 1

network 10.1.29.0 0.0.0.255 area 1

network 10.1.31.0 0.0.0.255 area 1

network 10.1.32.0 0.0.0.255 area 1

network 10.3.31.0 0.0.0.255 area 1

Router A

router ospf 1

log-adjacency-changes

redistribute connected subnets

redistribute rip

network 10.3.31.0 0.0.0.255 area 0

network 10.5.16.0 0.0.0.255 area 0

SNMP

The third step we performed is enabling SNMP access on all routers. This is done by configuring community strings, which act somewhat like passwords. Here's what this look like when configured:

snmp-server community public RO

snmp-server enable traps tty

The default router community is public which is mentioned in the above command.

Multicasting

In the fourth step IP multicasting was configured on router B, router C, router D and router A. For multicasting following commands is used.

ip multicast-routing

ip pim sparse-dense-mode

ip pim rp-address IP address

In order to enable the switch to forward multicast packets it is necessary to set up the following command which is not shown in the running configuration:

ip igmp snooping

a. To configure routers we setup RP for the PIM messages. Sparse Dense mode is required in the router interfaces, it also depends on the topology of the network only dense or sparse mode can be implemented. Independently from the use of RP or the spare and/or dense mode two points are important to enable the multicast

forwarding on the routers. The first one is to enable the IP Multicast routing, and the second is to enable PIM on the interfaces. All the switches connected in the network need to enable the IGMP forwarding, to let the hosts establish a membership with the routers and so they can receive the multicast packets. It is crucial to mention that CGMP is enabled by default on the switches and it is used to perform tasks similar to those performed by IGMP, but when IGMP spoofing is enabled the CGMP is disabled.

b. The IGMP (Internet Group Management Protocol) is used for the router to discover members of the multicast group connected to it and also to join and leave members of the group and for the hosts to establish their membership with the multicast group. PIM (Protocol Independent Multicast) is used between the routers so they can track which multicast packets to forward to each other and to their directly connected LANS. Other protocol used in the multicast process that wasn't seen in this lab, is the DVMRP (Distance Vector Multicast Routing Protocol) and it is used on the multicast backbone of the Internet (MBONE).

c. To enable a routing protocol between the routers for the multicast packets to reach their destination, therefore neither PIM nor IGMP provides a routing mechanism and there is a need to use a routing protocol like OSPF along with the multicast protocols suite.

To show connectivity between routers, how OSPF protocol are configured and how OSPF keep information of routes in a table we have executed some show command on the routers and output is showed below;

Priority

In the fifth step we have set router priority so that one of the router become DR and other can be BDR;

- Router B is elected as DR - with highest priority.
- Router C is BDR - second highest priority.

Output

routerA#sh ip ospf neighbor

Neighbor ID	Pri	State	Dead Time	Address	Interface
10.230.251.0	3	FULL/DR	00:00:39	10.5.16.3	GigabitEthern0

10.230.252.0 1 FULL/BDR 00:00:39 10.5.16.4 GigabitEthern0

routerA#

Following ping results prove communication between routers.

Ping result

routerB#

routerB#ping 10.1.29.2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.29.2, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 364/364/368 ms

routerB#ping 10.230.253.0

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.230.253.0, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

routerD#ping 10.20.16.2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.20.16.2, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 184/184/184 ms

routerD#

routerC#ping 10.1.29.3

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.29.3, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 180/183/184 ms

routerC#

sh ip route command showed that routerc can see other routers on the network:

routerC#sh ip route

```
O E2    10.20.16.0/24 [110/20] via 10.5.16.2, 02:01:13, FastEthernet0/0
O E2 172.20.0.0/16 [110/20] via 10.5.16.3, 20:26:43, FastEthernet0/0
     10.0.0.0/8 is variably subnetted, 12 subnets, 4 masks
C       10.230.252.0/32 is directly connected, Loopback0
O IA    10.10.16.0/24 [110/20] via 10.5.16.3, 20:26:43, FastEthernet0/0
O E2    10.230.254.0/32 [110/20] via 10.5.16.2, 02:01:13, FastEthernet0/0
O E2    10.230.253.0/32 [110/20] via 10.5.16.3, 20:26:43, FastEthernet0/0
O       10.1.16.0/22 [110/65] via 10.1.29.3, 1d00h, Serial0/0
O       10.1.16.0/20 is a summary, 1d00h, Null0
C       10.5.16.0/24 is directly connected, FastEthernet0/0
O       10.1.31.0/24 [110/128] via 10.1.29.3, 1d00h, Serial0/0
O       10.1.24.0/22 [110/65] via 10.1.29.3, 1d00h, Serial0/0
C       10.1.29.0/24 is directly connected, Serial0/0
O       10.1.20.0/22 [110/65] via 10.1.29.3, 1d00h, Serial0/0
O E2 172.16.0.0/12 [110/20] via 10.5.16.2, 02:02:03, FastEthernet0/0
routerC#
```

In this network we have setup two area border routers and overall 3 areas. Outputs are given below.

List all area border routers and their router LSA IDs.

Area Boarder Router

1. Router B LSA: 10.230.251.0
2. Router C LSA: 10.230.252.0

3. List all the AS border routers and AS external LSA originated by them.

Router A: AS Boundary Router

Link State ID: 10.230.254.0

Advertising Router: 10.230.254.0

&

Router E: AS Boundary Router

Link State ID: 10.230.253.0

Advertising Router: 10.230.253.0

After configuration of routers we have setup also Server to start traffic. We have setup Media Server, FTP Server, Proxy Server under Windows 2003 Server platform. This will help us to send type of type of traffic on the network to check traffic load. MRTG monitoring computer also setup at different network so that we can monitor traffic load on the network.

Routers configuration files are saved in appendix B

Chapter No. 3 MRTG / RRDTOOL Configuration and Implementation

MRTG

MRTG is a monitoring tool which can also gives graphical view of the traffic load on network connection. The MRTG produce HTML pages containing graphical images that provide a visual representation of the network traffic after every 5 minutes. The graphs produce by MRTG are daily, weekly, monthly and yearly scales we can see the bandwidth use on the network in these graphs. MRTG is good tool for analyzing network problems because it not only indicates the current status of the network but also compare this with the previous network traffic load. MRTG is based on Perl and C, and runs on Windows and UNIX operating systems.

To get the information from the router or any device on the network for which we're going to monitor the traffic. We have to configure SNMP because MRTG need SNMP enabled on every interface we want to monitor. MRTG is depending on SNMP, to obtain data from routers or other network hardware. Using the variables, MRTG sends SNMP requests every five minutes and stores the responses in a specific format in the log file. This format allows MRTG to present the daily, weekly, monthly, and yearly graphs without the data files forever growing larger. It does this by summarizing the older data as necessary. The graphs file format is Portable Network Graphics (PNG).

In MRTG we can graph the bandwidth in and out of any SNMP enabled network device including routers. With MRTG we can graph following:

- Bandwidth in and out in bits or bytes per second
- Bandwidth in an out of a particular VIP/virtual server or node/real server
- Connection rate (in connections per second)
- Any parameter that has an SNMP counter or gauge object/OID
- Total number of concurrent sessions

MRTG Installation step by step

MRTG is available at http://www.mrtg.org/ for download. First step we have to perform is to download windows version of MRTG and then unzip MRTG folder to C:\mrtg-2.15.2 on the Windows machine.

To get MRTG to work on Windows we need to install PERL. PERL can be downloading from the following site

http://www.activestate.com/Products/Download/Download.plex?id=ActivePerl

Perl was installed on the same Windows machine. We have to make sure that Perl binary directory is listed in the system path.

C:\Perl\bin;%SystemRoot%\system32;%SystemRoot%;

If it is not present in the system path then we have to enter it manually in

[Control Panel]-> [System] -> [Environment]

To see if everything is installed properly we can open a Command Shell and go into *c:\mrt\bin*.
Type: perl mrtg

This should give error message saying about the missing MRTG configuration file but that is ok.

CONFIGURING MRTG

Enabling SNMP

The majority of operating systems do not have the SNMP support enabled by default. The SNMP has to be installed or enabled in order to get the SNMP OID data collection working. The SNMP support offers the SNMP client, which listens for SNMP requests coming from a NMS (network management station) and delivers the requested SNMP values.

Before creating a configuration file for MRTG we should have the following information:

- The IP address or hostname and the SNMP port number of the device which is going to be monitored.
- To monitor something other than bytes in and out, we must also know the SNMPOID of what we want to monitor.
- The read-only SNMP community string for the device. In our case it is **public** that is by default.

We have configured SNMP on all routers available in rack 4 of MINT lab. IOS used in our network with Community string **public**.

Cisco routers/switches offer SNMP support running on a "public" community.

IOS commands that change the SNMP configuration:

(config)#snmp-server community <name> <access-type>
(config)#snmp-server enable traps [notification-type]

Creating .cfg file

After MRTG installation on a specific monitoring Server and SNMP Protocols setup on routers. Next step should be to configure MRTG so that MRTG should start communication with the monitoring device. For that we have to create and configure MRTG.cfg file in which we have to tell MRTG to capture data from that specific port. MRTG.cfg file is required for each monitored host. First thing we have to do is to create a default mrtg.cfg file. The .cfg file defines the SNMP OIDs for each entity that we want to monitor from the destination host. MRTG parses the associated .cfg file and collects the SNMP values for all OIDs defined in the .cfg file.

To create mrtg.cfg file we can run the "CFGMAKER" script or we can also create mrtg.cfg file manually it is on us which method we prefer. If choose to create mrtg.cfg through cfgmaker even then we have to edit that file and add script for the TCP or UDP traffic, we will find mrtg.cfg file in the c:\mrtg\bin\ directory. This script scans a host for the network-interfaces and constructs the mrtg.cfg file example is shown below.

On cmd prompt change to the *c:\mrtg-bin* directory. Type the following command:

Perl cfgmaker public@ interface IP address --global "WorkDir: c:\mrtgdata" --output mrtg.cfg

This creates an initial MRTG config file called mrtg.cfg every time we run above command it will create new mrtg.cfg file in c:\mrtg\bin directory which overwrites any exiting mrtg.cfg file if present in c:\mrtg\bin directory. In MRTG.CFG file all interfaces of the router will be stored by number. These numbers are likely to change whenever we reconfigure router. In order to work around this we can get *cfgmaker* to produce a configuration which is based on IP numbers, or even Interface Descriptions.

Following is the very basic sample configuration mrtg.cfg file which is only good for data in and out.

TargetDevice's IP Address: Interface Number: Community: IP Address

```
Target[IP address]: 1:public@ IP address

This is the interface speed (Default is 100 megabits; for 100Mbit devices use 12500000 and so on...)
MaxBytes[IP address]: 1250000

Title[IP address]: Monitor Traffic load on Router: ether0
```

This section determines how the web page headers will look

```
PageTop[IP address]: <H1>Traffic Analysis for Fastethernet 0/0</H1>
 <TABLE>
 <TR><TD>Router:</TD><TD> Monitor Traffic load on Router </TD></TR>
 <TR><TD>Maintainer:</TD><TD>Administrator</TD></TR>
 <TR><TD>Interface:</TD><TD>ether0(1)</TD></TR>
 <TR><TD>IP:</TD><TD>FastEthernet 0/0(IP address)</TD></TR>
 <TR><TD>Max Speed:</TD>
 <TD>12.5 MB/s (ethernetCsmacd)</TD></TR>
 </TABLE>
This section determine the traffic  if there is any and its description.

 Target[IP address.2]: 2:public@ IP address.1
 MaxBytes[IP address.2]: 125000000
 Title[IP address.2]: Monitor Traffic load on Router : FastEthernet0/0
 PageTop[IP address.2]: <H1>Traffic Analysis for FastEthernet</H1>
  <TABLE>
  <TR><TD>System:</TD><TD> Monitor Traffic load on Router </TD></TR>
  <TR><TD>Maintainer:</TD><TD>Admin</TD></TR>
  <TR><TD>Interface:</TD><TD>FastEthernet0/0</TD></TR>
  <TR><TD>IP:</TD><TD>()</TD></TR>
  <TR><TD>Max Speed:</TD>
  <TD>12.5 MB/s (ethernetCsmacd)</TD></TR>
  </TABLE>
```

To start MRTG and generate graphs to monitor network traffic we have to run following command on command prompt by going in the directory *c:\mrtg-2.15.2\bin*:

perl mrtg mrtg.cfg

It is normal to get errors for the first two times we run above command. The errors will alert about the fact that there have not been any log files in existence before.

To update the MRTG graphs we have to run perl mrtg mrtg.cfg every five minutes this how mrtg will give first lines in our graphs.

Configure MRTG to run all the time

If we want to see the update on MRTG graphs we have to run MRTG manually which is not a professional way. To run the MRTG all the time there is option available in the MRTG. We can set in the MRTG configuration file so that MRTG will not terminate after it was started. Instead it will wait for 5 minutes and then run again automatically. We need to add following option in mrtg.cfg

RunAsDaemon: yes

and at cmd prompt we have to type following command and execute it:

start /Dc:\mrtg-2.15.2\bin perl mrtg --logging=eventlog mrtg.cfg

to run mrtg after 5 minutes automatically.

If we use **wperl** instead of **perl**, no console window will show. MRTG should be running in the background. If it runs into problems it will log the errors in EventLog. To stop MRTG, open the Task Manager and terminate the **wperl.exe** process. To mrtg messages and error we can refer to event log.

It is also possible if we add

```
Target:    perl mrtg --logging=eventlog mrtg.cfg
Start in:  c:\mrtg-2.15.2\bin
```

into windows start-up folder, MRTG will now start whenever you login into windows.

Basic MRTG configuration file

Following is the basic MRTG configuration file to monitor bytes in and out. This mrtg.cfg is created in the MINT lab on Cisco router 2600.

```
# Created by : Parvez Ibrahim
# cfgmaker public@10.1.32.3 --global "WorkDir: c:\mrtgdata" --output mrtg.cfg
### Global Config Options
#  for UNIX
# WorkDir: /home/http/mrtg
#  or for NT
# WorkDir: c:\mrtgdata
### Global Defaults
#  to get bits instead of bytes and graphs growing to the right
```

```
# Options[_]: growright, bits
######################################################################
# System: routerD
# Description: Cisco Internetwork Operating System Software
#       IOS (tm) C2600 Software (C2600-J1S3-M), Version 12.2(15)T7,  RELEASE SOFTWARE (fc2)
#       TAC Support: http://www.cisco.com/tac
#       Copyright (c) 1986-2003 by cisco Systems, Inc.
#       Compiled Sat 09-Aug-03 07:18 by ccai
# Contact: Parvez Ibrahim
# Location: MINT Lab
######################################################################
### Interface 1 >> Descr: 'FastEthernet0/0' | Name: 'Fa0/0' | Ip: '10.1.32.3' | Eth: '00-08-21-bf-4b-40' ###
Target[10.1.32.3_1]: 1:public@10.1.32.3:
SetEnv[10.1.32.3_1]: MRTG_INT_IP="10.1.32.3" MRTG_INT_DESCR="FastEthernet0/0"
MaxBytes[10.1.32.3_1]: 12500000
Title[10.1.32.3_1]: Traffic Analysis for 1 -- routerD
PageTop[10.1.32.3_1]: <h1>Traffic Analysis for 1 -- routerD</h1>
                        <div id="sysdetails">
                                <table>
                                        <tr>
                                                <td>System:</td>
                                                <td>routerD in </td>
                                        </tr>
                                        <tr>
                                                <td>Maintainer:</td>
                                                <td></td>
                                        </tr>
                                        <tr>
                                                <td>Description:</td>
                                                <td>FastEthernet0/0  </td>
                                        </tr>
                                        <tr>
                                                <td>ifType:</td>
                                                <td>ethernetCsmacd (6)</td>
                                        </tr>
                                        <tr>
                                                <td>ifName:</td>
                                                <td>Fa0/0</td>
                                        </tr>
                                        <tr>
                                                <td>Max Speed:</td>
```

<td>12.5 MBytes/s</td>
</tr>
<tr>
<td>Ip:</td>
<td>10.1.32.3 ()</td>
</tr>
</table>

Interface 2 >> Descr: 'Serial0/0' | Name: 'Se0/0' | Ip: '10.1.31.3' | Eth: "
Target[10.1.32.3_2]: 2:public@10.1.32.3:
SetEnv[10.1.32.3_2]: MRTG_INT_IP="10.1.31.3" MRTG_INT_DESCR="Serial0/0"
MaxBytes[10.1.32.3_2]: 193000
Title[10.1.32.3_2]: Traffic Analysis for 2 -- routerD
PageTop[10.1.32.3_2]: <h1>Traffic Analysis for 2 -- routerD</h1>

<table>
<tr>
<td>System:</td>
<td>routerD in </td>
</tr>
<tr>
<td>Maintainer:</td>
<td></td>
</tr>
<tr>
<td>Description:</td>
<td>Serial0/0 </td>
</tr>
<tr>
<td>ifType:</td>
<td>propPointToPointSerial (22)</td>
</tr>
<tr>
<td>ifName:</td>
<td>Se0/0</td>
</tr>
<tr>
<td>Max Speed:</td>
<td>193.0 kBytes/s</td>
</tr>
<tr>

<td>Ip:</td>
<td>10.1.31.3 ()</td>
</tr>
</table>

Interface 3 >> Descr: 'Serial0/1' | Name: 'Se0/1' | Ip: '10.1.29.3' | Eth: "
Target[10.1.32.3_3]: 3:public@10.1.32.3:
SetEnv[10.1.32.3_3]: MRTG_INT_IP="10.1.29.3" MRTG_INT_DESCR="Serial0/1"
MaxBytes[10.1.32.3_3]: 193000
Title[10.1.32.3_3]: Traffic Analysis for 3 -- routerD
PageTop[10.1.32.3_3]: <h1>Traffic Analysis for 3 -- routerD</h1>

<table>
<tr>
<td>System:</td>
<td>routerD in </td>
</tr>
<tr>
<td>Maintainer:</td>
<td></td>
</tr>
<tr>
<td>Description:</td>
<td>Serial0/1 </td>
</tr>
<tr>
<td>ifType:</td>
<td>propPointToPointSerial (22)</td>
</tr>
<tr>
<td>ifName:</td>
<td>Se0/1</td>
</tr>
<tr>
<td>Max Speed:</td>
<td>193.0 kBytes/s</td>
</tr>
<tr>
<td>Ip:</td>
<td>10.1.29.3 ()</td>
</tr>

```
                                </table>
                        </div>
### Interface 4 >> Descr: 'Null0' | Name: 'Nu0' | Ip: '' | Eth: '' ###
### The following interface is commented out because:
### * it is a cisco Null0 interface
# # Target[10.1.32.3_4]: 4:public@10.1.32.3:
# SetEnv[10.1.32.3_4]: MRTG_INT_IP="" MRTG_INT_DESCR="Null0"
# MaxBytes[10.1.32.3_4]: 536870911
# Title[10.1.32.3_4]: Traffic Analysis for 4 -- routerD
# PageTop[10.1.32.3_4]: <h1>Traffic Analysis for 4 -- routerD</h1>
#                       <div id="sysdetails">
#                               <table>
#                                       <tr>
#                                               <td>System:</td>
#                                               <td>routerD in </td>
#                                       </tr>
#                                       <tr>
#                                               <td>Maintainer:</td>
#                                               <td></td>
#                                       </tr>
#                                       <tr>
#                                               <td>Description:</td>
#                                               <td>Null0  </td>
#                                       </tr>
#                                       <tr>
#                                               <td>ifType:</td>
#                                               <td>Other (1)</td>
#                                       </tr>
#                                       <tr>
#                                               <td>ifName:</td>
#                                               <td>Nu0</td>
#                                       </tr>
#                                       <tr>
#                                               <td>Max Speed:</td>
#                                               <td>536.9 MBytes/s</td>
#                                       </tr>
#                               </table>
#                       </div>
### Interface 5 >> Descr: 'Foreign-Exchange-Station-1/0/0' | Name: 'Foreign Exchange Station 1/0/0' | Ip: '' | Eth: '' ###
### The following interface is commented out because:
### * it is a Voice controller
```

```
### * got 'Received SNMP response with error code
###      error status: noSuchName
###      index 1 (OID: 1.3.6.1.2.1.2.2.1.10.5)
###    SNMPv1_Session (remote host: "10.1.32.3" [10.1.32.3].161)
###                     community: "public"
###                    request ID: 826815120
###                   PDU bufsize: 8000 bytes
###                      timeout: 2s
###                      retries: 5
###                      backoff: 1)' from interface when trying to query
# # Target[10.1.32.3_5]: 5:public@10.1.32.3:
# SetEnv[10.1.32.3_5]: MRTG_INT_IP="" MRTG_INT_DESCR="Foreign-Exchange-Station-1/0/0"
# MaxBytes[10.1.32.3_5]: 75000000
# Title[10.1.32.3_5]: Traffic Analysis for 5 -- routerD
# PageTop[10.1.32.3_5]: <h1>Traffic Analysis for 5 -- routerD</h1>
#                         <div id="sysdetails">
#                                 <table>
#                                         <tr>
#                                                 <td>System:</td>
#                                                 <td>routerD in </td>
#                                         </tr>
#                                         <tr>
#                                                 <td>Maintainer:</td>
#                                                 <td></td>
#                                         </tr>
#                                         <tr>
#                                                 <td>Description:</td>
#                                                 <td>Foreign-Exchange-Station-1/0/0  </td>
#                                         </tr>
#                                         <tr>
#                                                 <td>ifType:</td>
#                                                 <td>Voice Foreign eXchange Station (voiceFXS) (102)</td>
#                                         </tr>
#                                         <tr>
#                                                 <td>ifName:</td>
#                                                 <td>Foreign Exchange Station 1/0/0</td>
#                                         </tr>
#                                         <tr>
#                                                 <td>Max Speed:</td>
#                                                 <td>75.0 MBytes/s</td>
#                                         </tr>
```

```
#                                       </table>
#                               </div>
### Interface 6 >> Descr: 'Foreign-Exchange-Station-1/0/1' | Name: 'Foreign Exchange Station 1/0/1' | Ip: '' | Eth: '' ###
### The following interface is commented out because:
### * it is a Voice controller
### * got 'Received SNMP response with error code
###       error status: noSuchName
###       index 1 (OID: 1.3.6.1.2.1.2.2.1.10.6)
###     SNMPv1_Session (remote host: "10.1.32.3" [10.1.32.3].161)
###                       community: "public"
###                      request ID: 826815121
###                     PDU bufsize: 8000 bytes
###                         timeout: 2s
###                         retries: 5
###                         backoff: 1)' from interface when trying to query
## Target[10.1.32.3_6]: 6:public@10.1.32.3:
# SetEnv[10.1.32.3_6]: MRTG_INT_IP="" MRTG_INT_DESCR="Foreign-Exchange-Station-1/0/1"
# MaxBytes[10.1.32.3_6]: 75000000
# Title[10.1.32.3_6]: Traffic Analysis for 6 -- routerD
# PageTop[10.1.32.3_6]: <h1>Traffic Analysis for 6 -- routerD</h1>
#                               <div id="sysdetails">
#                                       <table>
#                                               <tr>
#                                                       <td>System:</td>
#                                                       <td>routerD in </td>
#                                               </tr>
#                                               <tr>
#                                                       <td>Maintainer:</td>
#                                                       <td></td>
#                                               </tr>
#                                               <tr>
#                                                       <td>Description:</td>
#                                                       <td>Foreign-Exchange-Station-1/0/1  </td>
#                                               </tr>
#                                               <tr>
#                                                       <td>ifType:</td>
#                                                       <td>Voice Foreign eXchange Station (voiceFXS) (102)</td>
#                                               </tr>
#                                               <tr>
#                                                       <td>ifName:</td>
#                                                       <td>Foreign Exchange Station 1/0/1</td>
```

```
#                                       </tr>
#                                       <tr>
#                                               <td>Max Speed:</td>
#                                               <td>75.0 MBytes/s</td>
#                                       </tr>
#                               </table>
#                       </div>
### Interface 7 >> Descr: 'Loopback0' | Name: 'Lo0' | Ip: '10.230.250.0' | Eth: " ###
### The following interface is commented out because:
### * it is a Software Loopback interface
#
# Target[10.1.32.3_7]: 7:public@10.1.32.3:
# SetEnv[10.1.32.3_7]: MRTG_INT_IP="10.230.250.0" MRTG_INT_DESCR="Loopback0"
# MaxBytes[10.1.32.3_7]: 536870911
# Title[10.1.32.3_7]: Traffic Analysis for 7 -- routerD
# PageTop[10.1.32.3_7]: <h1>Traffic Analysis for 7 -- routerD</h1>
#                       <div id="sysdetails">
#                               <table>
#                                       <tr>
#                                               <td>System:</td>
#                                               <td>routerD in </td>
#                                       </tr>
#                                       <tr>
#                                               <td>Maintainer:</td>
#                                               <td></td>
#                                       </tr>
#                                       <tr>
#                                               <td>Description:</td>
#                                               <td>Loopback0  </td>
#                                       </tr>
#                                       <tr>
#                                               <td>ifType:</td>
#                                               <td>softwareLoopback (24)</td>
#                                       </tr>
#                                       <tr>
#                                               <td>ifName:</td>
#                                               <td>Lo0</td>
#                                       </tr>
#                                       <tr>
#                                               <td>Max Speed:</td>
#                                               <td>536.9 MBytes/s</td>
```

```
#                                       </tr>
#                                       <tr>
#                                               <td>Ip:</td>
#                                               <td>10.230.250.0 ()</td>
#                                       </tr>
#                               </table>
#                       </div>
### Interface 11 >> Descr: 'Virtual-Access1' | Name: 'Vi1' | Ip: '' | Eth: '' ###
Target[10.1.32.3_11]: 11:public@10.1.32.3:
SetEnv[10.1.32.3_11]: MRTG_INT_IP="" MRTG_INT_DESCR="Virtual-Access1"
MaxBytes[10.1.32.3_11]: 12500000
Title[10.1.32.3_11]: Traffic Analysis for 11 -- routerD
PageTop[10.1.32.3_11]: <h1>Traffic Analysis for 11 -- routerD</h1>
                        <div id="sysdetails">
                                <table>
                                        <tr>
                                                <td>System:</td>
                                                <td>routerD in </td>
                                        </tr>
                                        <tr>
                                                <td>Maintainer:</td>
                                                <td></td>
                                        </tr>
                                        <tr>
                                                <td>Description:</td>
                                                <td>Virtual-Access1  </td>
                                        </tr>
                                        <tr>
                                                <td>ifType:</td>
                                                <td>ppp (23)</td>
                                        </tr>
                                        <tr>
                                                <td>ifName:</td>
                                                <td>Vi1</td>
                                        </tr>
                                        <tr>
                                                <td>Max Speed:</td>
                                                <td>12.5 MBytes/s</td>
                                        </tr>
                                </table>  </div>
```

Advance Level mrtg.cfg

As mentioned that the above script is basic mrtg script. To monitor something other than bytes in and out, we have to add script to capture different type of traffic for example TCP and UDP for that we have to also add SNMPOID in mrtg.cfg file of what we want to monitor. In our case we have added OIDs in mrtg.cfg file for TCP and UDP packet.

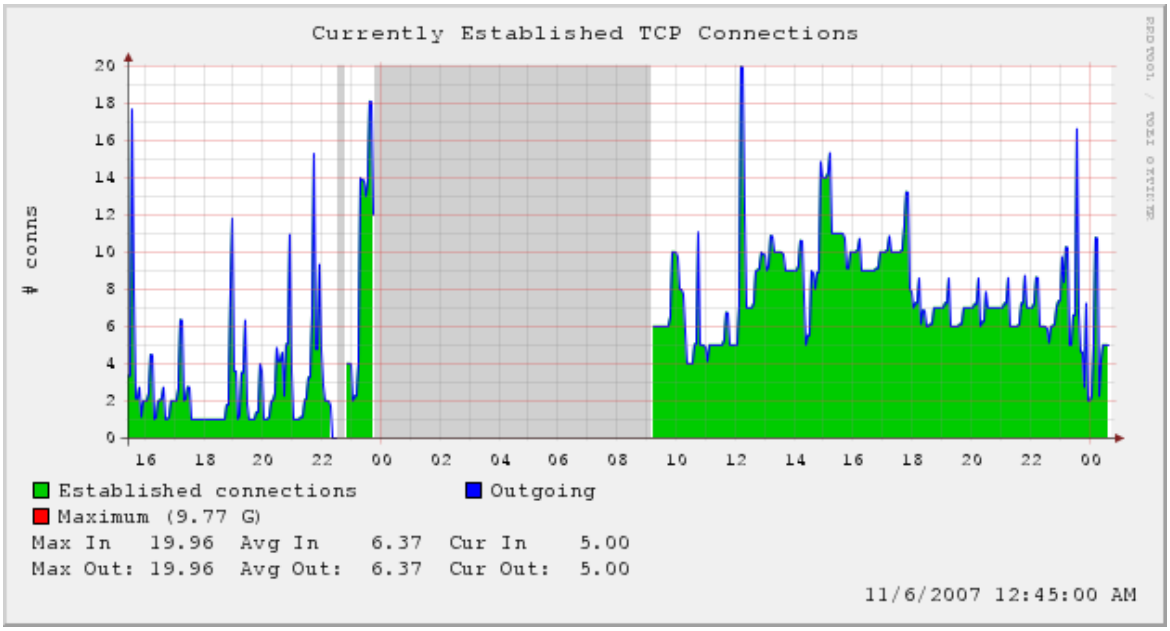
To explicitly define which OID to query by using the following syntax 'OID_1&OID_2: community@router' The following example will retrieve TCP connection traffic from interface 1. MRTG needs to graph two variables, so it is needed to specify two OID's such as TCP and UDP packets or error input and error output.

Following is the just small part of the script which is added in the mrtg.cfg to capture data for TCP and UDP packets. Complete mrtg.cfg is given in appendix A.

```
###The number of TCP connections for which the current state is either ESTABLIS.
Target[tcpopen]:.1.3.6.1.2.1.6.9.0&.1.3.6.1.2.1.6.9.0:public@192.168.0.2
Options[tcpopen]: nopercent,growright,gauge,noinfo
Title[tcpopen]: tcpCurrEstab
PageTop[tcpopen]: tcpCurrEstab
MaxBytes[tcpopen]: 1000000
YLegend[tcpopen]: # conns
ShortLegend[tcpopen]: connections
LegendI[tcpopen]:  Connections:
LegendO[tcpopen]:
Legend1[tcpopen]: tcpCurrEstab

#The total number of UDP datagrams delivered to UDP users.

Target[udp2]:.1.3.6.1.2.1.7.1.0&.1.3.6.1.2.1.7.1.0:public@192.168.0.2
Options[udp2]: nopercent,growright,gauge,noinfo
Title[udp2]: udpInDatagrams
PageTop[udp2]: udpInDatagrams
MaxBytes[udp2]: 1000000
YLegend[udp2]: # udp datagram
ShortLegend[udp2]: udp delivered
LegendI[udp2]:  udp delivered:
LegendO[udp2]:
Legend1[udp2]: udpInDatagrams
```



TCP Connections

For our project we have used following OID which include both TCO and UDP SNMP OIDs.

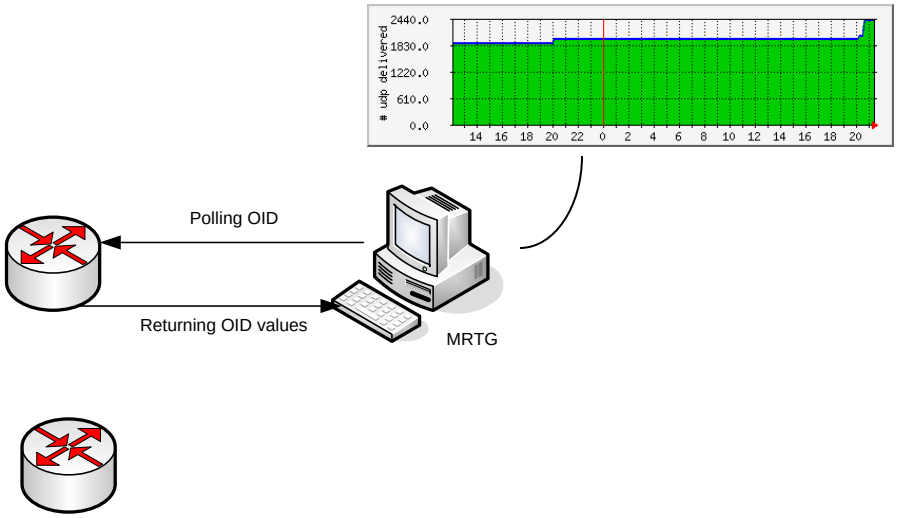
TCP OID

Name	OID
tcpRtoAlgorithm	.1.3.6.1.2.1.6.1
tcpRtoMin	.1.3.6.1.2.1.6.2
tcpRtoMax	.1.3.6.1.2.1.6.3
tcpMaxConn	1.3.6.1.2.1.6.4
tcpActiveOpens	.1.3.6.1.2.1.6.5
tcpPassiveOpens	.1.3.6.1.2.1.6.6
tcpAttemptFails	.1.3.6.1.2.1.6.7
tcpEstabResets	.1.3.6.1.2.1.6.8
tcpInErrs	.1.3.6.1.2.1.6.14
tcpOutRsts	.1.3.6.1.2.1.6.15
tcpHCInSegs	.1.3.6.1.2.1.6.17
tcpHCOutSegs	.1.3.6.1.2.1.6.18

UDP OID

Name	OID
udpInDatagrams	.1.3.6.1.2.1.7.1
udpNoPorts	.1.3.6.1.2.1.7.2
udpInErrors	.1.3.6.1.2.1.7.3
udpOutDatagrams	.1.3.6.1.2.1.7.4
udpHCInDatagrams	.1.3.6.1.2.1.7.8
udpHCOutDatagrams	.1.3.6.1.2.1.7.9

Figure is showing how MRTG polling OID



For each OID referenced in the configuration file, MRTG creates the following graphs:

- **Daily graph**—5 minute average data points with approximately 33 hours of data presented.
- **Weekly graph**—30 minute average data points with approximately 8 days of data presented.
- **Monthly graph**—2 hour average data points with approximately 5 weeks of data presented.
- **Yearly graph**—1 day average data points with approximately 1 year of data presented.

Configuration summary

a. Perl, MRTG and the Web Server are installed.

b. Configuration files (.cfg) for all monitored hosts have been generated using CFGMAKER. These files will be guiding the MRTG process in the SNMP acquisition.

c. HTML index files for all hosts are generated and copied in the folders where the image c:\mrtgdata files are. Mostly recommended is to name it "index.html" and use a folder for each monitored host to store its own related data.

d. Next step: Get the MRTG process running and visualize the variation graphs.

Integration of RRDTool with MRTG

MRTG works with RRDTOOL which improves its performance and graphing flexibility. RRDtool is used as the logger to MRTG. It stores data samples on each of the network switch interfaces (ports) in a separate RRD. To minimize size of the database files, RRD uses the consolidation mechanism. It guarantees that the database does not grow over time and that old data is automatically eliminated.

Installing and configuring the RRDTooL

The installation steps are:

1. Download the RRDTOOL package and unzip it to the chosen RRD folder. A free copy of RRDTOOL can be downloaded form www.rrdtool.com/download.html

2. The package contains the binary files. The src folder contains four subfolders where some RRDTools are available in .exe format. We have to copy these files in the rrdtool\bin\ folder for an easier access. We have also included the rrdtool\bin in the system path.

3. Register RRDTOOL package with the currently installed Perl distribution (at least Perl 5.6). Go to the "perl-shared" folder and run the following command:

ppm install rrds.ppd

RRDTOOL is now ready for use. Next step is to configure the MRTG instances to write SNMP data into the RRD databases.

Running MRTG instances with RRD database support

The MRTG instances with RRD database support can be ran in the same manner as we mentioned above for example.

Run from the command prompt the following command:

perl mrtg mrtg.cfg //we have to execute this command every 5 minutes

or

start /Dc:\mrtg\bin perl mrtg --logging=eventlog mrtg.cfg

// add RunAsDaemon: Yes in mrtg.cfg config file and run this command for one time only

MRTG will read the SNMP data at the specified interval and will add the values to the database instead of just updating the image-graphs.

Integration /Configuring MRTG with RRD database support

When using mrtg with RRDtool we are actually replacing *rateup* with the RRDtool perl module *RRDs.pm*.

We have to reconfigure MRTG.cfg for RRDTool steps are given below:

1. Build the MRTG .cfg file for the monitored target using the CFGMAKER

2. Update the .cfg file with the following configuration

Logformat: rrdtool // To enable RRDtool support in mrtg

Workdir: folder name // rrdtool repository folder

PathAdd: path to the rrdtool bin folder // For the location of the rrdtool executable

LibAdd: path to the rrdtool perl-shared folder // For the location of the perl module

RunAsDaemon: Yes // add this command in MRTG configuration file so that MRTG will not terminate after it was started.

After applying above modification in the configuration file following effects take place when we run mrtg again with the new modifications in config file:

1. Mrtg will take all your old .log files and convert them to .rrd format. The .log files are still in the same folder in case process doesn't work we can still use .log files with mrtg.cfg old settings.
2. Mrtg will use rrdtool to update its databases. These will have a new format called *rrd* which is totally different from the. *log* format.
3. Mrtg will not create any webpages of graphs anymore. It will only query the routers for traffic information and update its *rrd* databases.

The advantage of whole integration is that MRTG become much faster. Expect the runtime would drop to 20%. Logging process of RRDtool is very fast. The whole concept behind RRDTool MRTG integration is that it is more efficient to create graphs and webpages on demand by using a cgi script.

To generate the graph from rrd files we have to run CGI script files. We have setup and configure two CGI script files routers2.cgi and 14all.cgi.

Routers2.cgi

Installation and Configuration

Following are the step-by-step procedure we applied in windows operating system we did this installation on windows 2003, windows XP and windows Vista successfully.

1. Prerequisite software.

In order to use routers2.cgi, we install certain prerequisite software. Prerequisite software is necessary to installed before step no. 2.

1.1. Setup Web Server

For our project we have configured IIS on three machines all machines have different windows operating systems Windows Server 2003, windows xp and windows Vista.

Windows Server 2003 configuration: Web Server

1. Enable IIS in add remove program
2. Enable world wide web service
3. In computer management select Services and applications

4. Then select Internet Information Service → web Sites → Default web site
5. Right click on Default web site → properties then click on Home directory
6. Then click on configuration.
7. In Application configuration window we should change application mapping
8. Add following
   - assign drive:/path/perl.exe %s %s to .pl
   - assign drive:/path/perl.exe %s %s to .cgi
   - assign drive:/path/rrd_cgi.exe %s %s to .rrdcgi
9. Start the Web Service →Web Site → right-click → Start.
10. This sets PERL to execute locally whenever an app with those extensions is selected
11. Modify web server so that the directory has execute permissions

Windows Vista

1. Enable Internet Information Services
2. Enable World Wide Web Services
3. In computer management select Services and applications
4. Then select Internet Information Service → web Sites → Default web site
5. Select Handler Mapping → Add script and add following one by one
6. Then click on configuration.
7. Add following
   - assign drive:/path/perl.exe %s %s to .pl
   - assign drive:/path/perl.exe %s %s to .cgi
   - assign drive:/path/rrd_cgi.exe %s %s to .rrdcgi
8. Start the Web Service →Web Site → right-click → Start.
9. This sets PERL to execute locally whenever an app with those extensions is selected

In internet browse enter localhost WebServer is displaying its default page to the browser

1.2. ActivePerl

Already install and configured ActivePerl from the http://www.activestate.com/ web site. Application is up and running.

1.3. MRTG and RRDTool

These are already downloaded from Tobi's sites at http://www.mrtg.org/ and http://www.rrdtool.org/ and configured in our system

2. Installation Process of cgi script:

To install routers.cgi, first uncompress the folder using any uncompress software. Second step execute the installation script

install.pl

during installation we have to provide paths of the prerequisite software required for cgi script.

After router2.cgi installation we have to edit router2.cgi as per windows environment and our settings. In the first step we have to change the path '#!/usr/local/bin/perl' to #!C:\Perl\bin\PERL.EXE. In the second step we have to install the RRDs perl libraries. While editing script we have to keep this in mind that the script is UNIX based so we have to check all the syntax which is only used in UNIX. For example windows uses backstrokes '\' as path separators, URLs use forward strokes '/'. So, when editing the routers2.conf file, we should use the correct strokes when defining URL paths and filename paths and we should use appropriate changes for directory paths, permissions obviously we need the Windows version of Perl installed and made CGI-able.

Creating Directories

Now we need to make directories where we put our files. We need following directories:

Icons Dir: In this directory all the .gif files should be installed. This directory should not be under the CGI-BIN directory.

Graphs Dir: Graphs directory is important it the directory where all graphs are stored it also a working directory. The directory needs writeable permission with read and executes rights. It is important that this directory should have writable right because when routers2.cgi query data from rrd files the script first write the graphs in the graph directory.

CGI-BIN Dir: This is where the routers.cgi file should go, with read and executes permission. This directory should be visible to the web browser with exec-cgi flag set.

MRTG CONF Dir: This is the directory where MRTG .conf files are stored. This directory should already present at the time of MRTG installation. In our configuration the path is c:\mrtg\bin

DATABASE Dir: Where the RRDTool .rrd databases are kept. This should already exist. It should not be under web server root.

Configuring the Perl script

All scripts used with RRDTool are mostly UNIX based and these scripts need to be reconfigured as per our own environment. This is now done via separate configuration file. We should modify the script so that it knows where the conf file is and modify the path of the perl executable in the first line of the script. ($CONFFILE = "....")

The conf file should have at least 2 sections, [web] and [routers.cgi]. The first needs an entry 'backurl' to define where the 'Main Menu' button takes us. The other section defines all the other options that used to be hardcoded into the script.

A fourth section, [targetnames] and fifth section, [targettitles] are optional. A fourth section, [targetnames] which allows us to override the default short description for the routers and interfaces. A fifth section, [targettitles] allows to give the 'long' description for each interface.

This configuration file should be kept in the inetpub\wwwroot - and then modify the script routers2.cgi so that it knows where to find this file. The line to modify is clearly indicated at the start of the script.

3. **Installing the files**

All the files should be installed to the c:\inetpub\wwwroot directory, permission read and execute except graph directory. This directory should be given write permission as well. In the package we will find routers2.cgi.pl which should be renamed to routers2.cgi. Copy the *.gif files to the ICONS directory.

4. **Test web link**

To test web interface enter http://localhost/routers2.cgi in the web browser. This will display menu page. The graph will probably be all grey at the moment, since the data has not yet been collected. If the page shows error message about 'rrd file not found' that's mean probably MRTG is not running successfully to create the rrd database.

14all.cgi script

14all.cgi is a CGI script to create html pages and graphics for mrtg. 14all.cgi parses the mrtg config file and uses most of the information to create;

- main index page: one link for every "Directory [...]: adir"
- group index pages: one for every "Directory..." with daily graphics
- statistic pages: one for every target with daily/weekly/monthly/yearly graphics according to "Suppress[...]: ..."

Installation and Setup

We have install 14all.cgi in a directory where the web server allows execution of cgi scripts. The rights to this directory should be readable and executable by the user.

Check the first line of the cgi script: It has to contain the full path to the perl interpreter. It should look like this:

#!C:\Perl\bin\perl

Library include path

14all.cgi script needs the file MRTG_lib.pm which is part of mrtg and present in mrtg\lib\mrtg2 directory.

We have to edit the script and change line 13 to contain the path to this file line 13 in script file should be look like;

use lib qw(C:\mrtg\lib\mrtg2);

We have to mentioned mrtg config file name into the script. There is a section where the perl variable *$cfgfile* is set. Change the appropriate line. The path should be absolute.

$cfgfile = 'C: \mrtg\bin\mrtg.cfg';

This how we can make changes in the 14all.cgi file then we have to run this script in the web browser by typing http://localhost/14all.cgi this will display the graphs.

Chapter 4 MRTG Results

Methodology: The MRTG Test

The MRTG provide graphical view of the data in and out from the router interface. Enabling SNMP in the router it collects measurements from the router MIB and then create daily graph every 5 minutes. In that graph MRTG show the traffic load on the link and how much bandwidth is available.

MRTG logs from all links along the path and the knowledge of the capacity of all links. We have applied the MRTG test to a pair of paths for which we have such data. One path, which traverses 1 hops and 100 Mb/s Ethernet connecting the router D.

The other path is from Router B network, with 2 hops. This path also has a link of 100 Mb/s. We have monitored these paths over a period of several days and for a total monitoring time of 24 hours. The data send though IDTG. In this experiment we have setup one machine at the sender end and one machine at the receiver end. Since MRTG data provides an average over a period of 5 minutes.

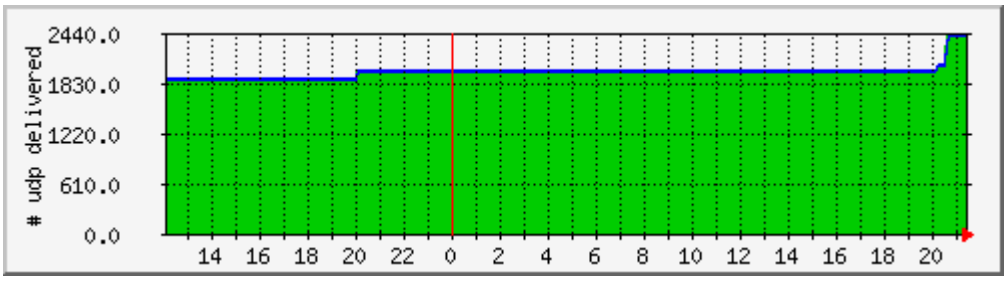
Occasionally, we actively increase the cross traffic traversing the monitored path. The objective of this induced load is to discover the responsiveness of the monitoring tool to changing network conditions. The sender of the cross traffic is different from the machine running the tool. The cross traffic uses UDP (though similar results were obtained with TCP cross traffic).

MRTG Test Results

Figures 4 and 5 shows results of our test. The plot shows the available bandwidth over a period of a day as measured by MRTG. In figure 4, traffic sent at 2:00 pm at a rate of 1830 B/s continuously till 8:00 pm. From 8:00 pm to 8:00 pm more traffic injected @ of 1850 B/s. From 8:00 pm to 9:00 pm data injected @ 2404 B/s.

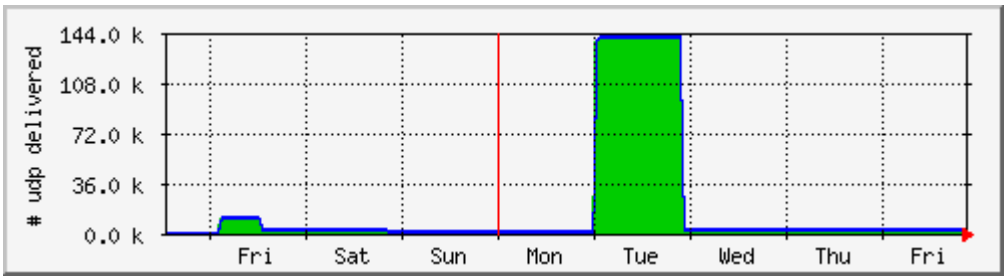
Figure No. 4

`Daily' Graph (5 Minute Average)



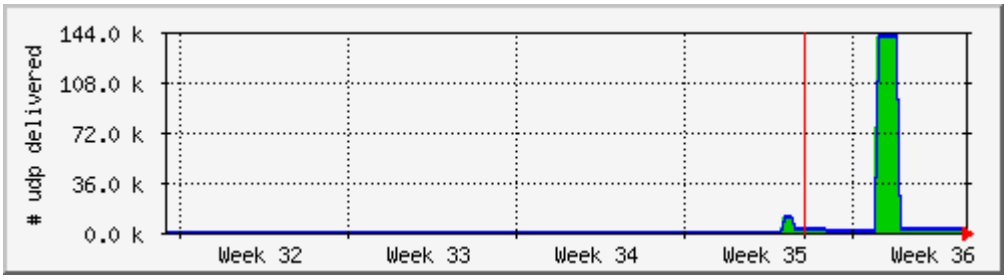
	Max	Average	Current
UDP :	2404.0 B/s	1946.0 B/s	2404.0 B/s

`Weekly' Graph (30 Minute Average)



	Max	**Average**	**Current**
udp:	140.5 kb/s	18.4 kb/s	2382.0 B/s

`Monthly' Graph (2 Hour Average)



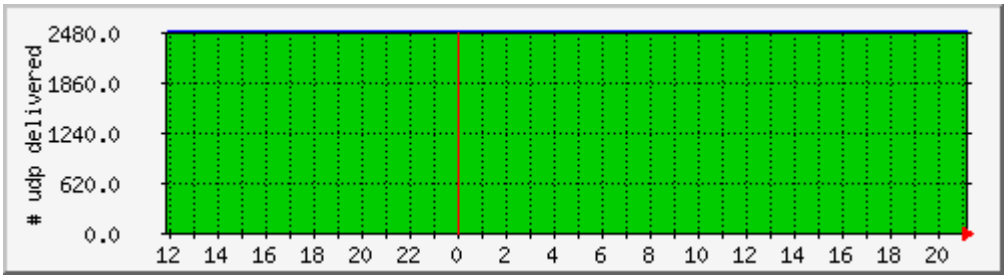
Max	Average	Current
140.5 kb/s	18.4 Kb/s	1956.0 B/s

In figure 5, traffic injected at 12:00 pm at a rate of 2480 Kb/s continuously till 21:00 pm. The rest of the time, we only monitored the path.

Figure No. 5

UDP datagram. The statistics were last updated **Thursday, 27 September 2007 at 21:30**

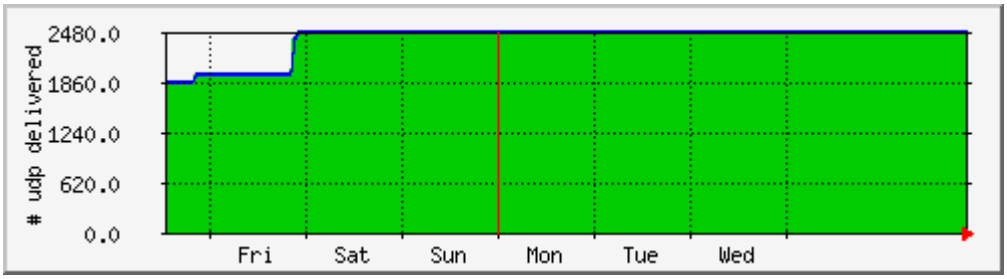
`Daily' Graph (5 Minute Average)



	Max	Average	Current
udp:	2476.0 B/s	2476.0 B/s	2476.0 B/s

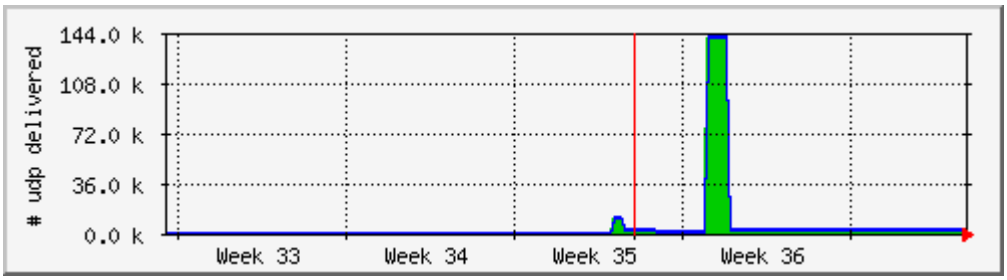
Weekly graph shows that the traffic is injected at the rate of 2480 B/s. On Friday traffic was injected @ 1860.0 B/s. We can see in the weekly graph that traffic load on Friday is low as compare to other day where the traffic load was @ 2480.0 B/s.

`Weekly' Graph (30 Minute Average)



	Max	Average	Current
udp:	2476.0B	2390.0B	2476.0B

`Monthly' Graph (2 Hour Average)



	Max	Average	Current
udp:	140.5 kb	10.8 kb	2476.0B

Following daily graph represents the traffic send through FTP during the course of one day. On average the server on the other end receives data max @ 57.563 Kb/s. From this graph we can conclude that the max data received @ 57.563 kb/s after 12:00 pm.

But not many files receive between the hours of 21:00 pm to 23:00 pm we can see the green spikes falls after 18:00 pm and there are few spikes because we have injected more traffic in the network. After 12:00 pm green spikes pick up again due to more files send from the client end.

Top Router MINT Lab-

System: Top Router

Maintainer:

Description: FastEthernet 0/0

ifType: ethernetCsmacd (71)

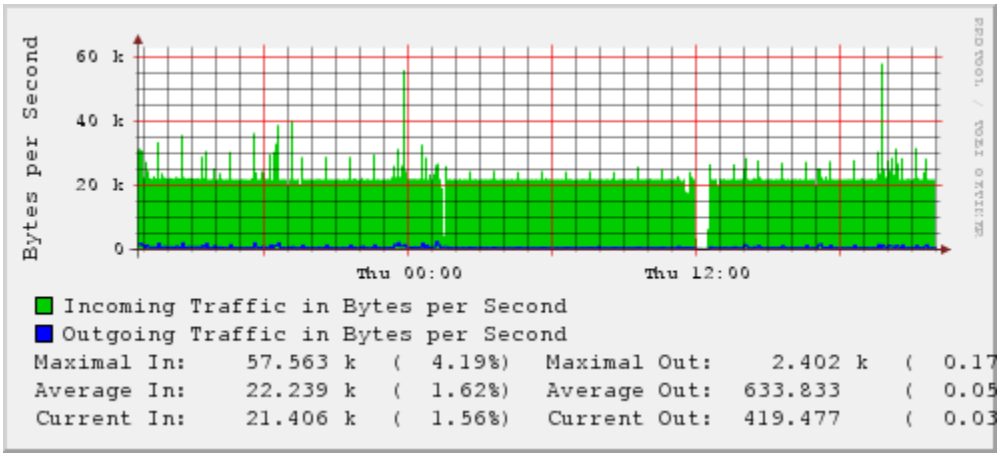
ifName: FastEthernet 0/0

Max Speed: 12.5 MBytes/s

Ip:

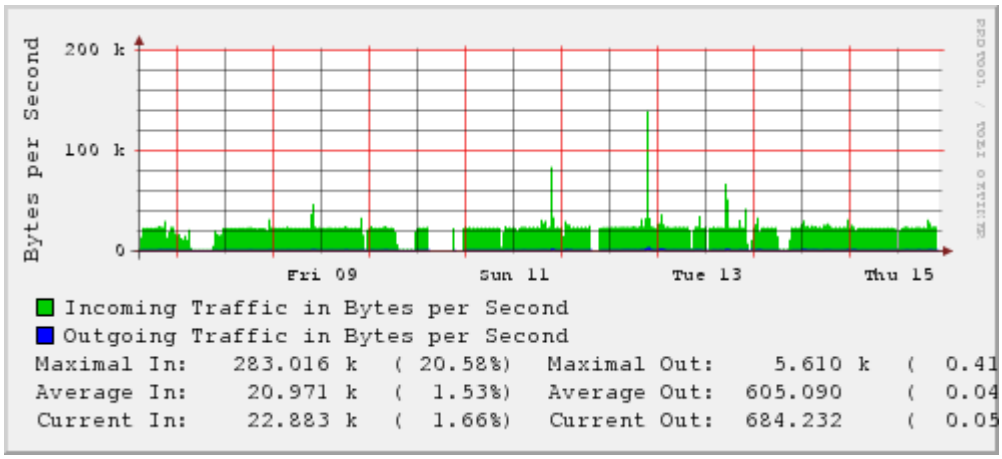
The statistics were last updated: **Thu Nov 15 22:00:00 2007**

'Daily' graph (5 Minute Average)

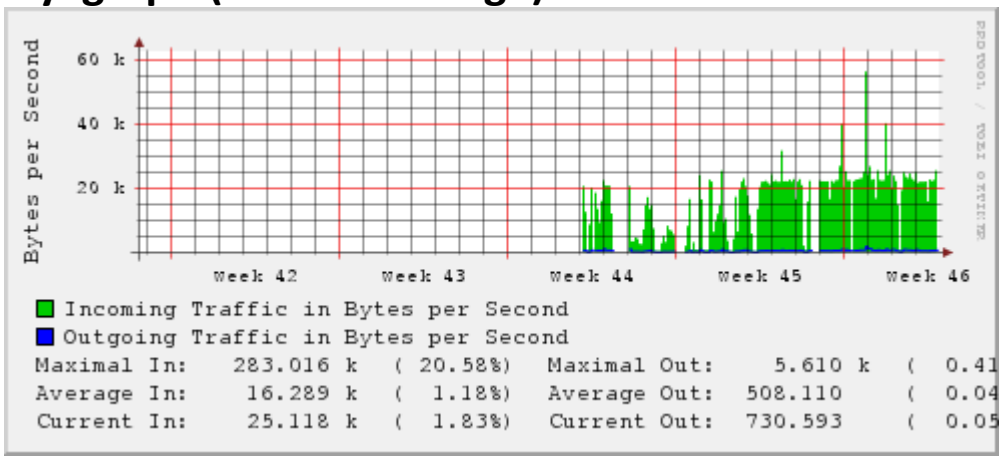


Weekly graph shows that maximum data received in @ 140.0 kb/s and maximum data send is very low. In the weekly graph there are some high spikes on Sunday and Tuesday because we injected more data so that we can see traffic load.

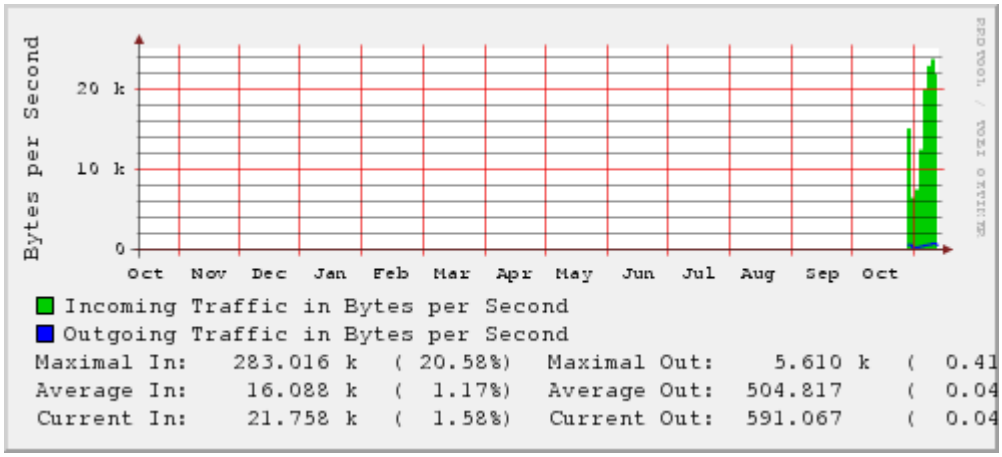
'Weekly' graph (30 Minute Average)



'Monthly' graph (2 Hour Average)



'Yearly' graph (1 Day Average)



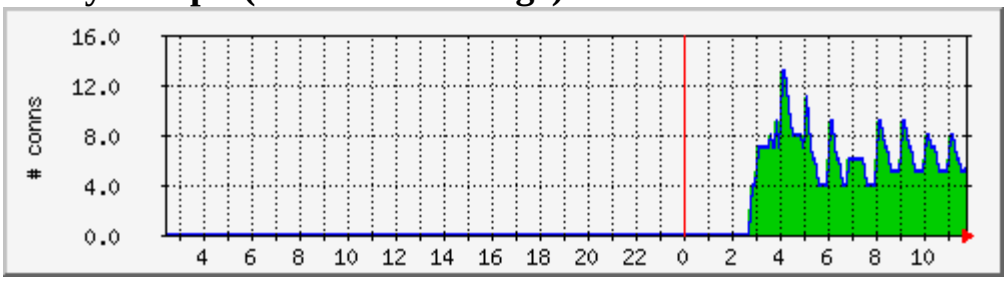
Established TCP Connections

TCP is used mostly for Internet's most popular application protocols like www, E-mail, File Transfer Protocol, Secure Shell, and some streaming media applications. Following graph showing the TCP connections. TCP provides connections that need to be established before sending data. In this experiment we have setup Proxy Server and FTP Server and media streaming to captured TCP connections then we monitored the MRTG for 2 to 3 weeks.

The following daily graph is showing TCP connection. In this graph which is showing 5 current connection and 13 max connections during the day. Average connections during the day are 6.0. Connections established means that graph is representing TCP three way handshakes completed. This graph is going to be update after every 5 minutes.

tcpCurrEstab. The statistics were last updated **Thursday, 20 September 2007 at 11:47**

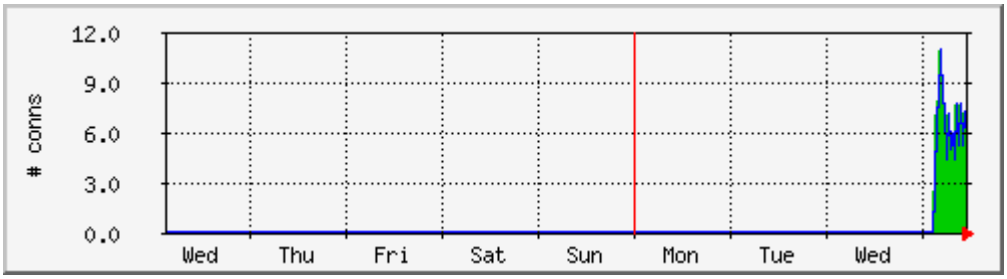
`Daily' Graph (5 Minute Average)



	Max	Average	Current
Connections:	13.0 connections	6.0 connections	5.0 connections

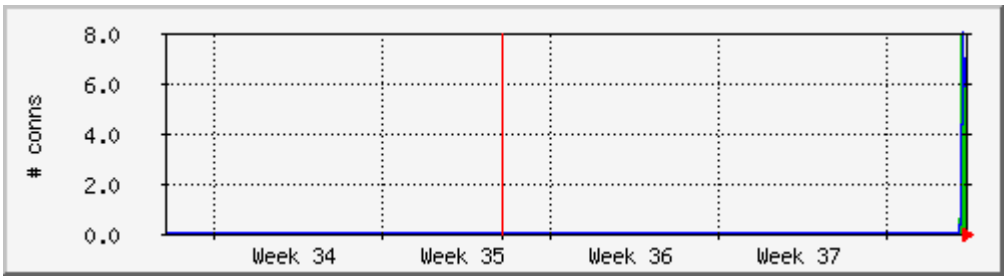
Following graph is the weekly graph which update in 30 minute average showing current TCP connection 5, max connections 10 and average 6 connections.

`Weekly' Graph (30 Minute Average)



	Max	Average	Current
Connections:	10.0 connections	6.0 connections	5.0 connections

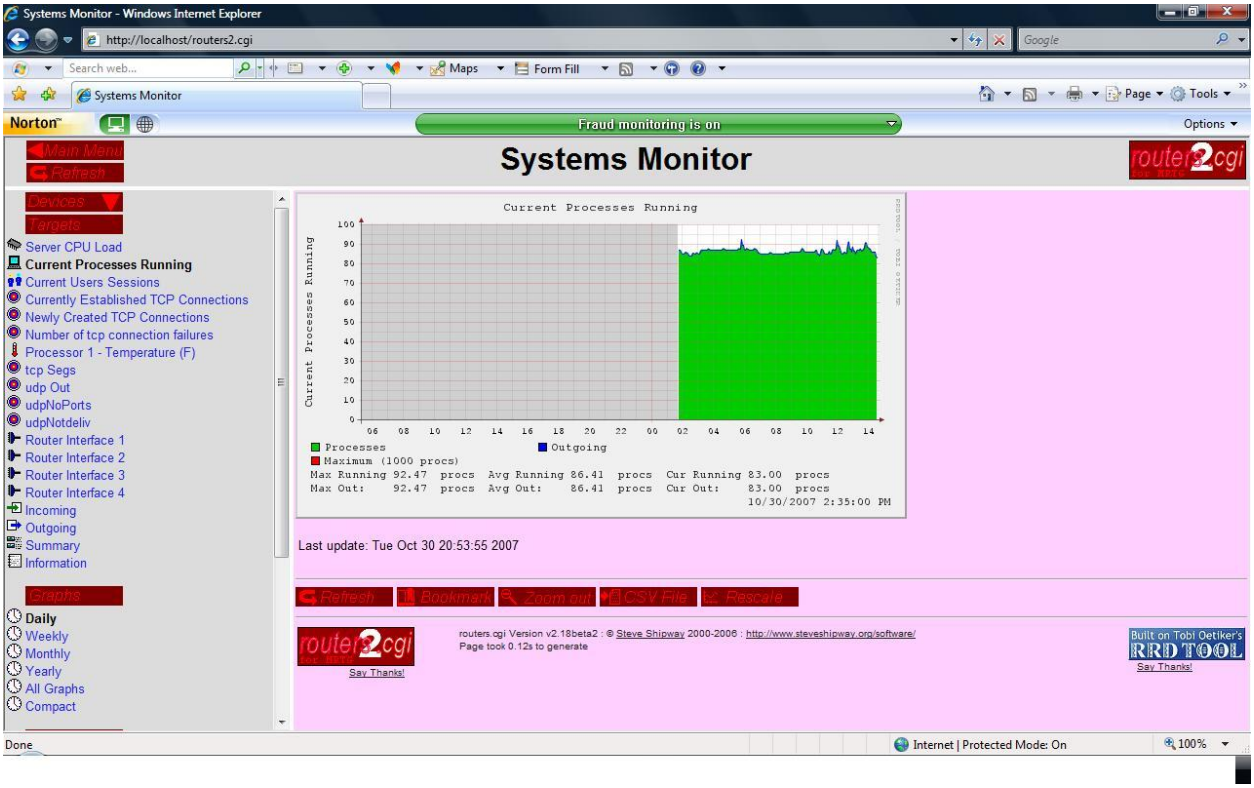
`Monthly' Graph (2 Hour Average)



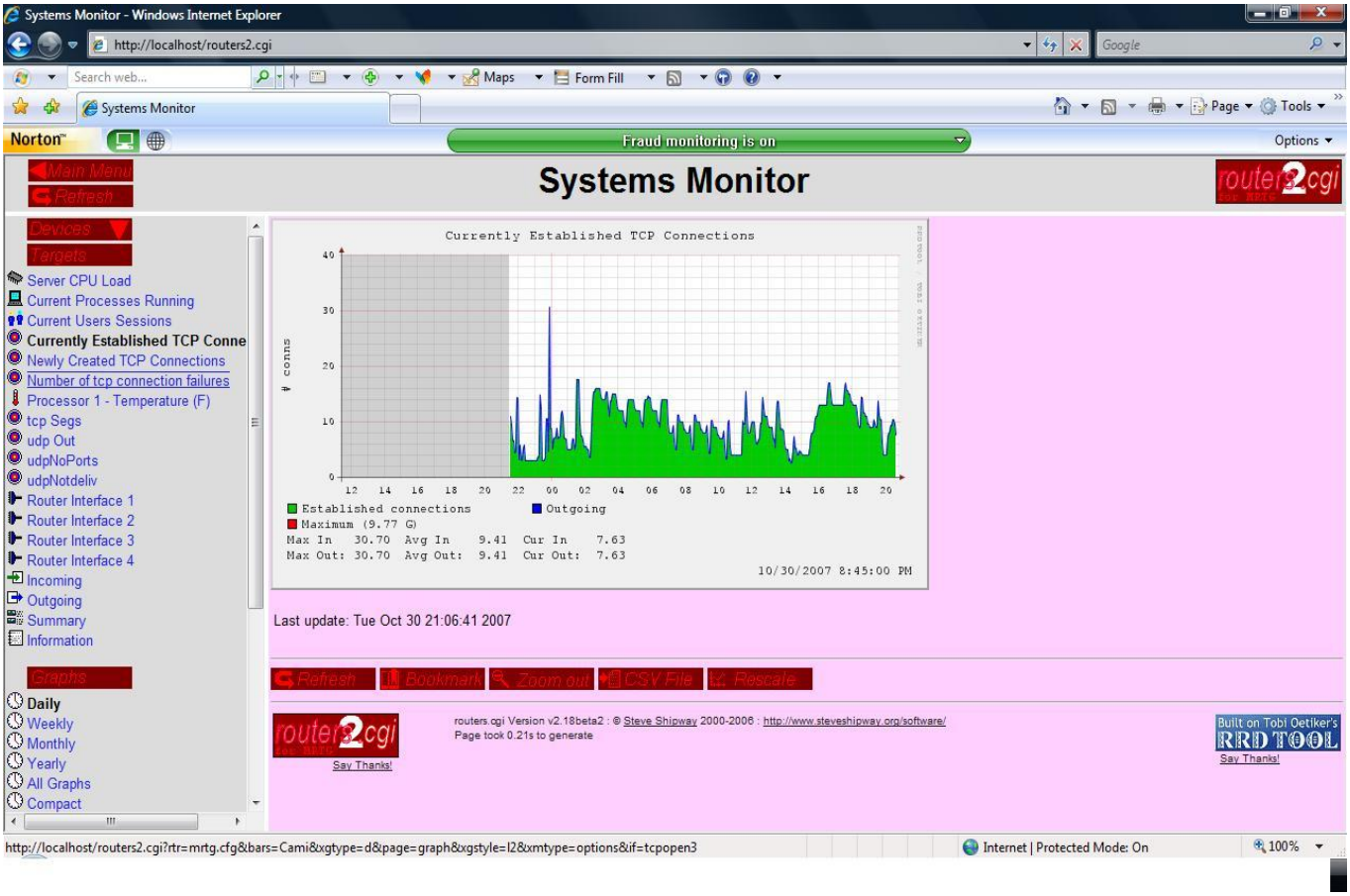
	Max	Average	Current
Connections:	8.0 connections	5.0 connections	6.0 connections

Following graphs are taken from RRDTool by using routers2.cgi script at the frontend. We can see the difference in graphical presentation in RRDTool graphs is much better than the MRTG. The graph is showing current processes running on the Server. The front end script running is routers2.cgi. Basically it create a web page on which we have several options on the left side of the page there is a menu from which we can select option for example we can select any

interface, tcp connection established, current sessions, newly created TCP connections, UDP data, incoming and outgoing traffic and then we can also pick which graph we want to see for example daily, weekly, monthly or yearly and we have several other option which we can use. We can easily say RRDTool graph is much better than MRTG.



Following graph is another example of RRDTool. In this graph we can see currently established TCP connection when three way handshakes completed. Here basically we are trying to showing the presentation of the RRDTools as compare to MRTG graphs.



In appendix c more graphs are stored.

Chapter 5. MRTG Implementation in NorQuest College

MRTG SETUP IN NORQUEST COLLEGE

NorQuest College Network Architecture

NorQuest College has seven campuses out of seven four sites are in Edmonton area. The Main, East Court, Capital center campuses are located in the downtown of Edmonton, whereas other campuses are Westmount Campus (westend), Stony plain Campus, Wetaskiwin Campus and Fort Saskatchewan Campus. Total network users in NorQuest are more than 1200 including students.

Following applications are used in NorQuest College;

WebCT (distance learning)

eLive (Virtual Classrooms)

Agresso (student, HR, finance)

Voyager (Library website)

VOIP

Terminal Services

Can8 (Language learning)

Web server

File and Print servers

Exchange server (e-mail)

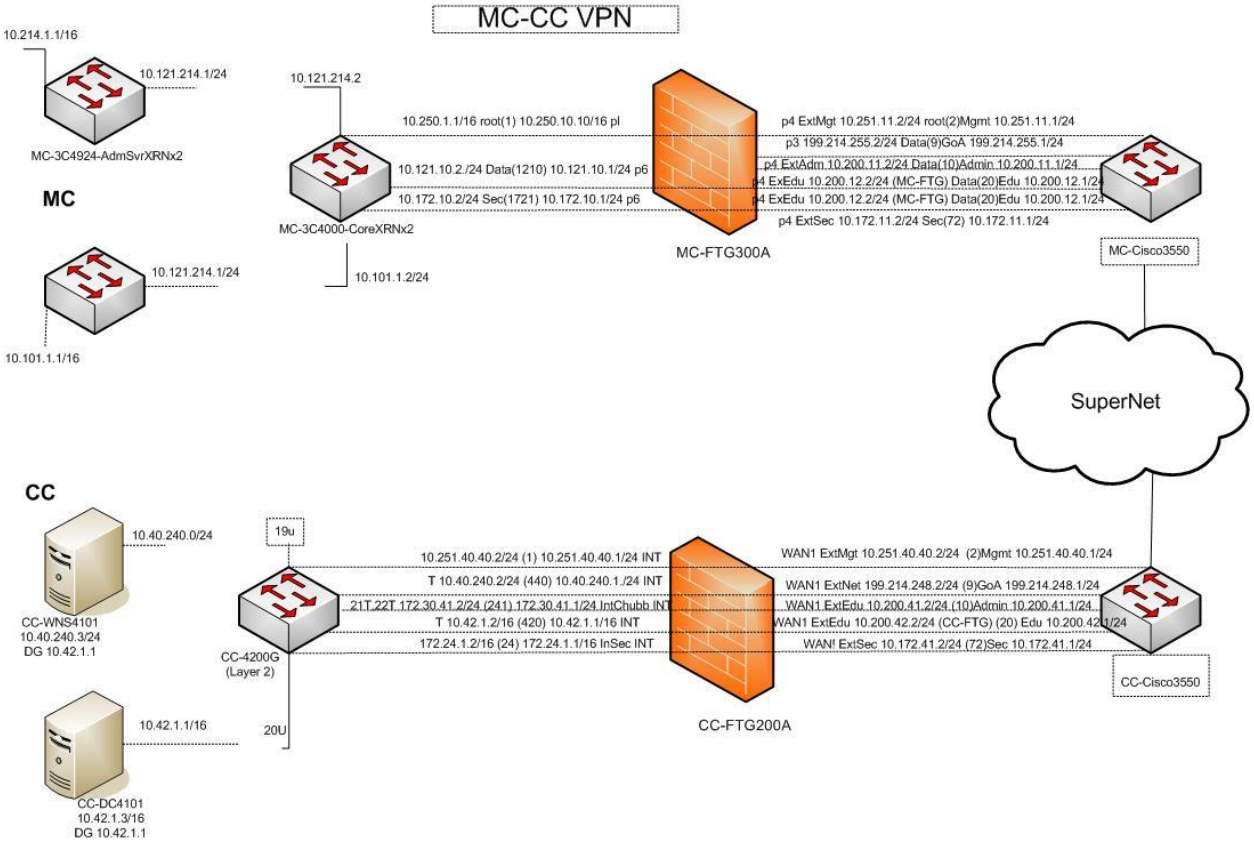
Magic (help desk)

The Main campus of NorQuest College work as central hub which is connected with all the campuses located within or out of the city. Three Main Domains are Norquest.ca (Parent domain) and Admin and Edu Domains (child domains). Admin Server is dedicated for staff members and Edu are dedicated for professors and students. Under each Domain have two sub-

domain controllers for redundancy offering Directory services, DNS and DHCP services to local and remote users. Admin and Edu domains have Windows 2000 and 2003 Server OS installed.

Fortigate Router

Fortigate, a routing and firewall device performing edge routing installed at each site of the College. At Main campus, due to the heavy traffic flow Fortigate FTG3600 is used for internet access and FTG300A is used for data Voice and video, at West mount and Stony plain sites Fortigate FTG300A, at Capital Center and Wetaskiwin sites Fortigate FTG200A have been installed. All the remote sites have been connected to Main campus by FTG300A, an ASIC-based layer 7 firewall that does antivirus, spam control, intrusion, and content filtering, FTG300A is linked to the internal network of the College through Juniper Netscreen device and 3COM core switch 3C-4050.



SuperNet

All campuses of the college are connected through SuperNet to Main Campus. All sites are connected through SuperNet then SuperNet via white Bell SuperNet box which is connected to media converter from fiber to twisted-pairs. It is then connected to port 1 of the Axia Cisco Catalyst 3550. The port 2 of the Cisco Catalyst 3550 switch which is connected to the Fortigate FTG300A at Main Campus.

Main Campus Network Setup

Devices installed in this campus are layer 2 and layer 3 switches. Servers and services are duplicated for redundancy purpose. Approximately 75 to 100 switches are installed on the network. The network is subnetted on the admin & Edu Domains as a result of using VLAN to provide smaller broadcast domains and having 2 NICs or 2 or more IP addresses on most servers. VLANs have been assigned on a functional basis across different floors e.g. Admin VLAN, Edu VLAN. This would provide traffic isolation and enhance security.

Core Devices

Core switches 3C4050 and two 3C4924 3Com products are used at the backbone of the college's network at Main campus catering all the traffic flow of ADM and EDU networks including Web Server, Mail Server, and Internet Server to the 2 riser core switches 3COM 3C4924 & 3C 5500 one for Adm and one for Edu network, these riser core switches connects to the other 3Com switches placed at the each floor of the college, these switches then connects to all the user workstations.

MRTG at NorQuest College (Main Campus)

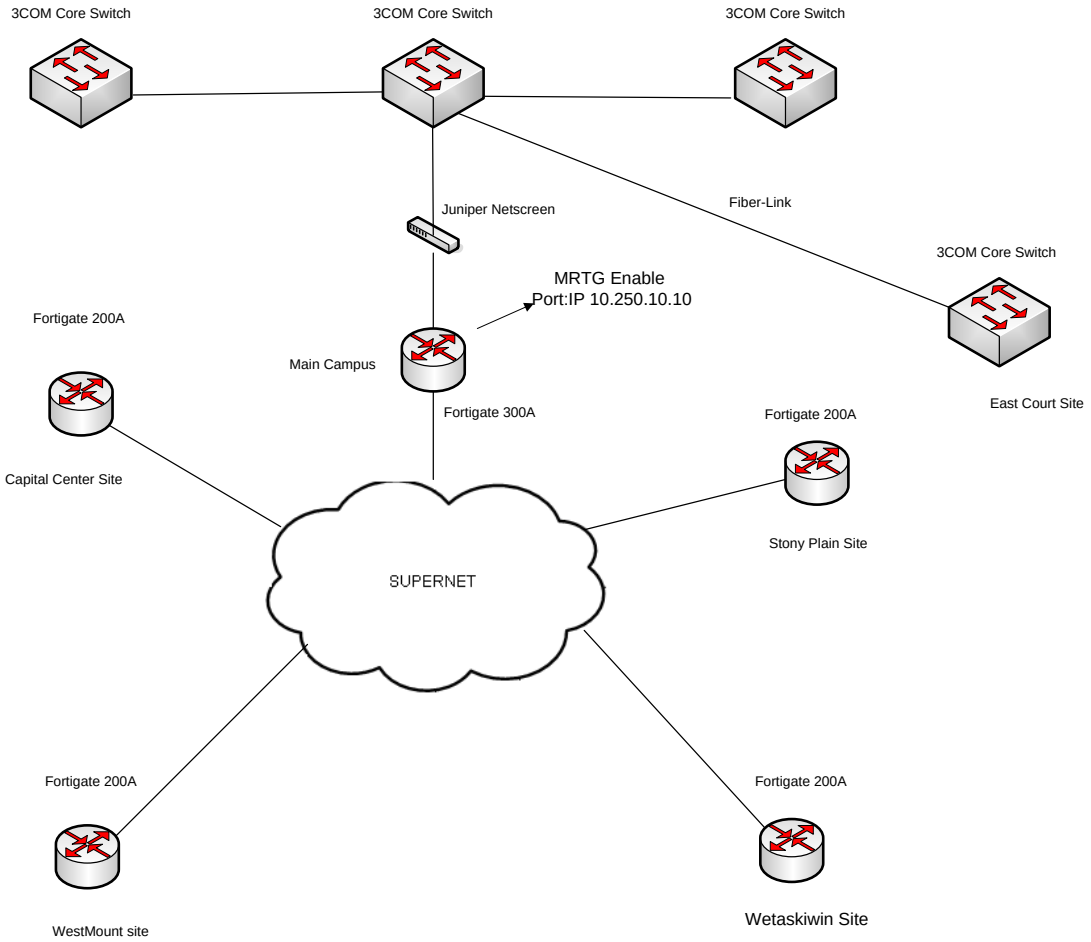
We have to select device which is most suitable for monitoring traffic on which all type of packets should be available. We were trying to select device on which traffic should be present 24 hours. Fortigate product FTG300A is the best device available in the NorQuest College Network. TCP IP, UDP, and SMTP traffic flow is present on this device. Input Ethernet port IP 10.250.10.10 of the FTG300A is select for MRTG monitoring for that SNMP has also been enabled on FTG300A. Second port where heavy traffic flow across the network could be at the main core of the network (backbone) where all the servers including servers in DMZ are placed and used by internal users and also remote users as well. The core switch in the middle among the three switches is an appropriate point to monitor. This core switch directly connected to the Fortigate 300A device and handles all the traffic from remote user, this switch also directly connected to East court site through fiber optic link and handles traffic flow from the users and servers installed at the sites, this switch also connected to the core Admin riser switch and

handles the traffic flow from the admin staff users. This core switch is also connected with two other core switches to handles the traffic flow from Edu servers, Edu users, Admin servers and NorQuest servers (DMZ zone). 10.121.214.2 is the IP of Ethernet port of the core switch which is selected for heavy traffic flow on this port, SNMP has also been enabled on the switch.

Setup MRTG WebServer in NorQuest College

Windows 2003 Server installed on Dell GSX260 machine. This machine has 250 GB hard disk, 512 MB Ram and one network interface card. IIS 6.0 and MRTG both are configured on this machine. How to configure IIS and MRTG on the machine is already explained in chapter 2. MRTG monitoring Server has been connected to the switch and have given access to the core switch 3C4050 and Fortigate 300A device.

Main Network



Traffic load @ NorQuest College

To monitor the traffic load on NorQuest college network we have configure MRTG to run all the time because after office hours many users use NorQuest College network remotely. Users in NorQuest College use variety of applications for example educational applications, Internet, Library resources, Magic, email, database, etc. Between office hours many staff and students internally and remotely use college network.

After several days observation on different network devices and port we have found that fortigate and 3COM Core Switch are the best devices to monitor because most of the traffic passed through these devices ports. We create MRTG scripts, enabled SNMP and established webserver to capture traffic for both devices.

Traffic Analysis at Fortigate FTG300A Port 1

System: MC-FTG300A in

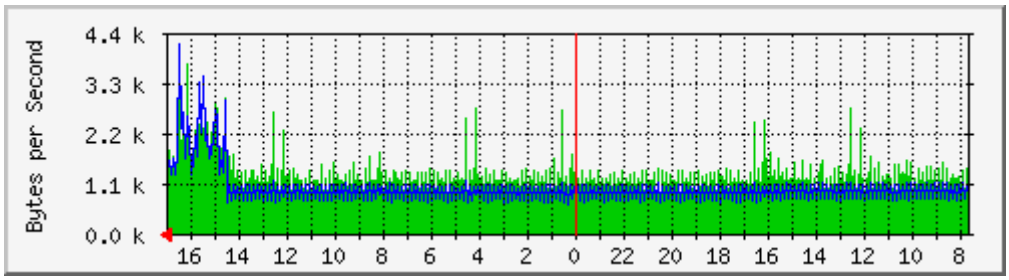
Description: port1

ifType: ethernetCsmacd (6)

Max Speed: 13.1 MBytes/s

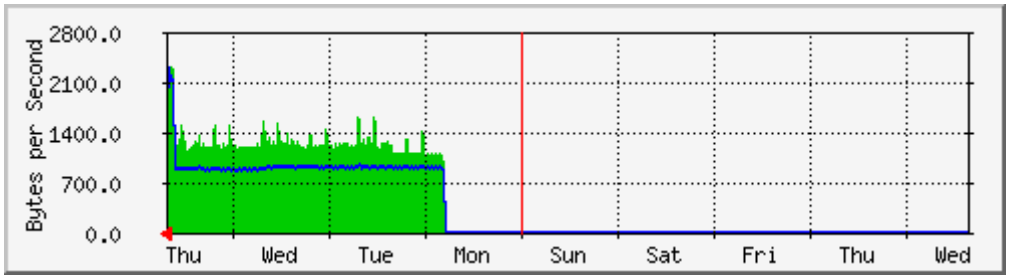
Ip: 10.250.10.10 ()

`Daily' Graph (5 Minute Average)



	Max	Average	Current
In	3711.0 B/s (0.0%)	1339.0 B/s (0.0%)	1616.0 B/s (0.0%)
Out	4136.0 B/s (0.0%)	971.0 B/s (0.0%)	1282.0 B/s (0.0%)

`Weekly' Graph (30 Minute Average)



	Max	Average	Current
In	2677.0 B/s (0.0%)	1274.0 B/s (0.0%)	2678.0 B/s (0.0%)
Out	2554.0 B/s (0.0%)	930.0 B/s (0.0%)	2555.0 B/s (0.0%)

Observation

On Fortigate FTG 300A port 1, router is link to the 3Com Core switch which is for network management data between main campus and remote sites used to manage the whole network from a single site, there is not much traffic flow on this link, continuously used for monitoring purposes.

We have observed through graph that data-out from the Netmon station @ main campus sent out to remote devices and get data in, the difference between the variations of both lines can be noticed and this is due to the amount of information received from remote network devices. The max data in the daily graph is 37110 B/s, current data is 1616.0 B/s and average is 1339.0 B/s. Where as data out 4136.0 B/s, current is 917.0 B/s and the average is 1282.0 B/s.

Traffic Analysis at Fortigate FTG300A- Port 3

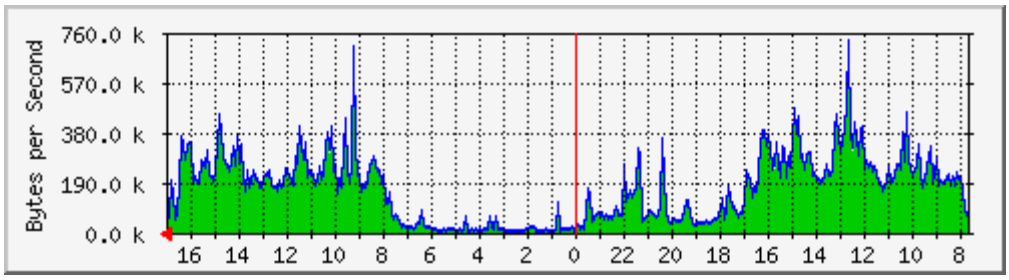
System: MC-FTG300A in

Description: port3

ifType: ethernetCsmacd (6)

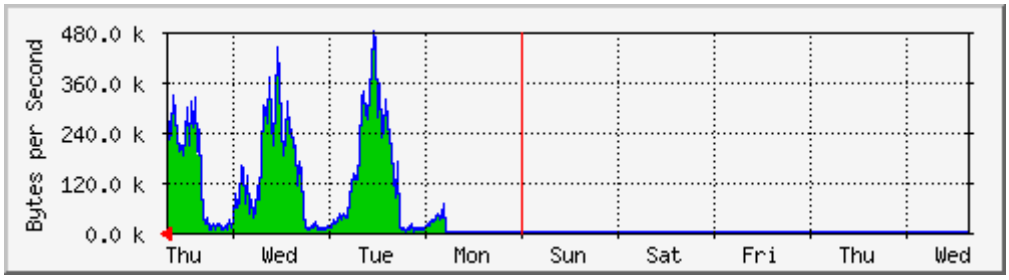
Max Speed: 13.1 MBytes/s

`Daily' Graph (5 Minute Average)



	Max	Average	Current
In	709.9 kB/s (5.4%)	156.5 kB/s (1.2%)	31.2 kB/s (0.2%)
Out	724.9 kB/s (5.5%)	161.0 kB/s (1.2%)	32.5 kB/s (0.2%)

`Weekly' Graph (30 Minute Average)



	Max	Average	Current
In	464.6 kB/s (3.5%)	125.4 kB/s (1.0%)	294.6 kB/s (2.2%)
Out	477.8 kB/s (3.6%)	128.9 kB/s (1.0%)	300.8 kB/s (2.3%)

Observation

The above graphs are showing traffic load on fortigate port 3 data in and out during different time intervals of the week days week days. Following are the statistics of the traffic. Traffic analysis has been categorized in three time intervals from 7:30 am – 5:00 pm, 5:00 pm – 12:00 am and 12:00 am – 6:00 am.

a From 7:30 am – 5:00 pm

During 7:30 am – 5:00 pm there is a high spikes, max data-in 709.9 kb/s and max data-out is 724.9 kb/s occurred @ 12:30 pm this is peak network usage hours in NorQuest College due lab classes, office working and remote connections from other sites.

b From 5:00 pm – 12:00 am

As the graph is showing as the day reached to end traffic is started decreasing we don't see much traffic flow. We can also see after some traffic load after office hours due to some staff or remote users are still working in the office

c From 12:00 am – 6:00 am

During these hours low traffic load is very low across the network which is near to zero bytes per second, during this time period no user accessing the network locally except few remote users, same reflect in the weekly and monthly graphs.

Traffic Analysis at Fortigate FTG300A- Port 4

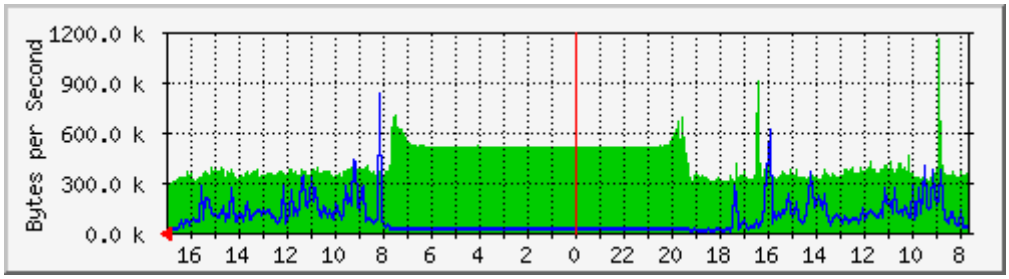
System: MC-FTG300A in

Description: port4

ifType: ethernetCsmacd (6)

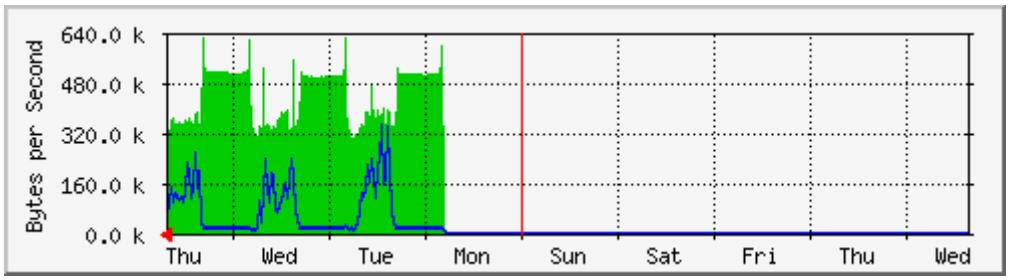
Max Speed: 13.1 MBytes/s

`Daily' Graph (5 Minute Average)



	Max	Average	Current
In	1160.6 kB/s (8.9%)	420.7 kB/s (3.2%)	289.6 kB/s (2.2%)
Out	821.8 kB/s (6.3%)	79.4 kB/s (0.6%)	19.6 kB/s (0.1%)

`Weekly' Graph (30 Minute Average)



	Max	Average	Current
In	623.2 kB/s (4.8%)	440.3 kB/s (3.4%)	339.2 kB/s (2.6%)
Out	342.1 kB/s (2.6%)	66.0 kB/s (0.5%)	50.3 kB/s (0.4%)

Observation

Using Fortigate FTG 300A router port 4 that is linked with the Supernet through Cisco catalyst switch 3550 (edge device). Data including admin and edu internet management , voice and video are travelling in and out by this port to the outside (supernet), high network traffic can be observed on this port. In graph, data-out during midnight is close to zero as during that time no network activity normally happens in the main campus, but data-in is high due to the remote user access and video streaming from one remote site to the main campus.

Max data-in in the daily graph is 1160.6 kB/s, average is 420.7 kB/s and current 289.6 kB/s whereas data-out max is 821.8 kB/s, average 79.4 kB/s and current 19.6 kB/s. We can see high data-in load after office early hour and before office because many users remotely connected through this port and use internet and different applications. In the weekly graph we can see high data-in load.

Traffic Analysis at Fortigate FTG300A- Port 6

System: MC-FTG300A in

ifName: ethernetCsmacd (6)

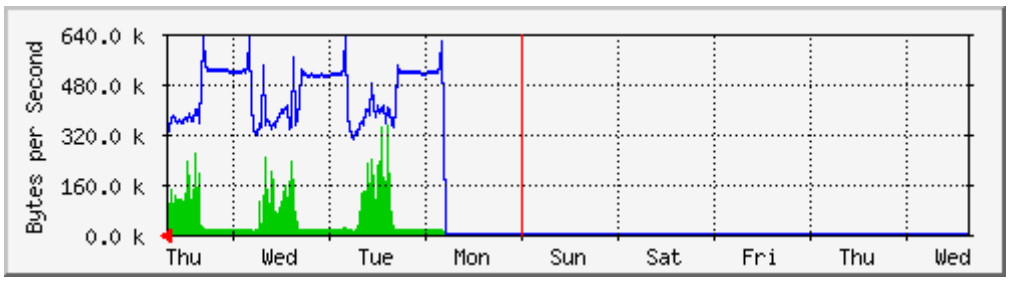
Ip: 10.214.10.1 ()

Max Speed: 13.1 MBytes/s

`Daily' Graph (5 Minute Average)

	Max	Average	Current
In	825.3 kB/s (0.6%)	81.4 kB/s (0.1%)	19.9 kB/s (0.0%)
Out	1182.7 kB/s (0.9%)	427.2 kB/s (0.3%)	292.4 kB/s (0.2%)

`Weekly' Graph (30 Minute Average)



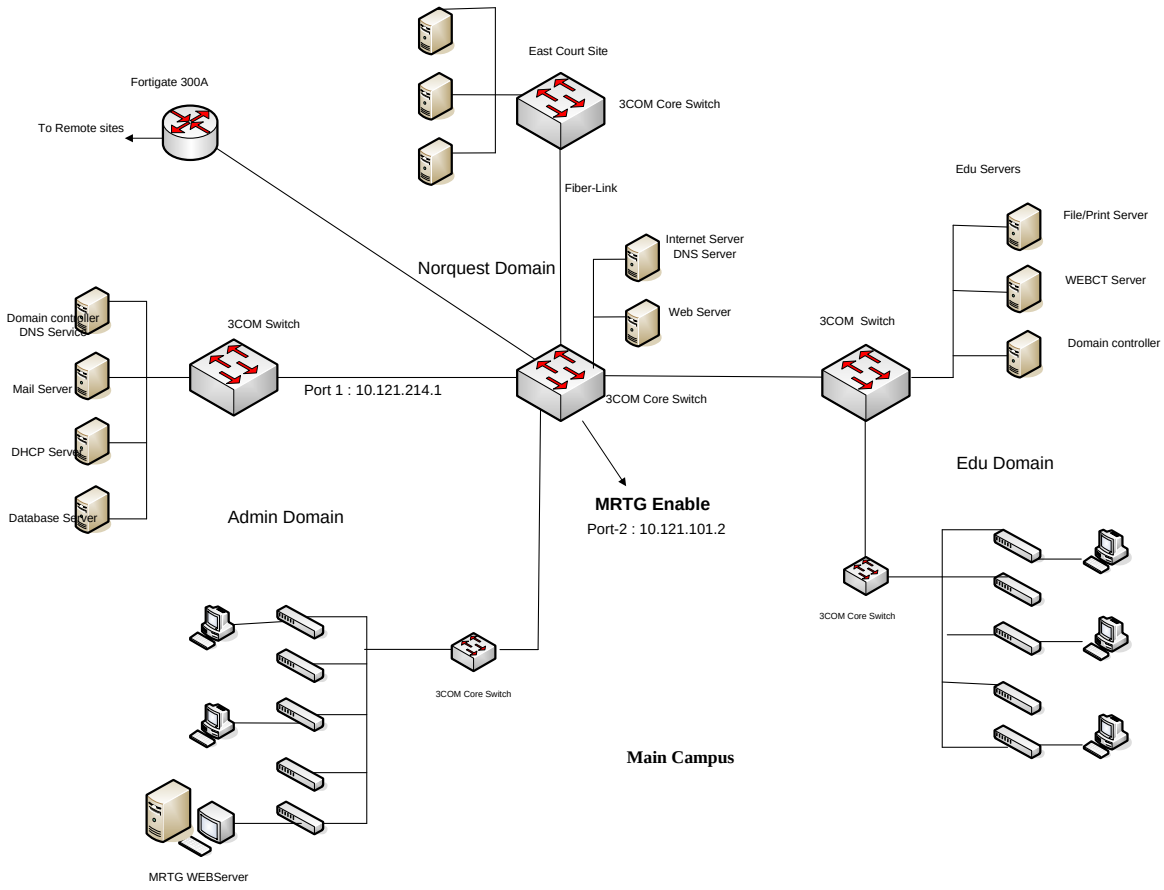
	Max	Average	Current

In	346.3 kB/s (0.3%)	68.4 kB/s (0.1%)	50.1 kB/s (0.0%)
Out	630.5 kB/s (0.5%)	446.7 kB/s (0.3%)	343.0 kB/s (0.3%)

Observation

In the daily and weekly graph data out is more than data in because from this port 6 clients connected basically send request for financial application that is why blue spikes are higher than green. Port6 of the Fortigate FTG 300A router is connected to the 3Com core switch, used for data, video and voice. Video streaming is used from security cameras from one of the remote site to main campus. Type of traffic flow is emails, applications, internet, files and print access. Remote users accessing network by using VPN and terminals services. Domain controllers replicate each other at regular time interval.

Network Structure of the Main Campus (downtown)



Traffic Analysis at 3COM-Core-Switch 3C4050 Port 2

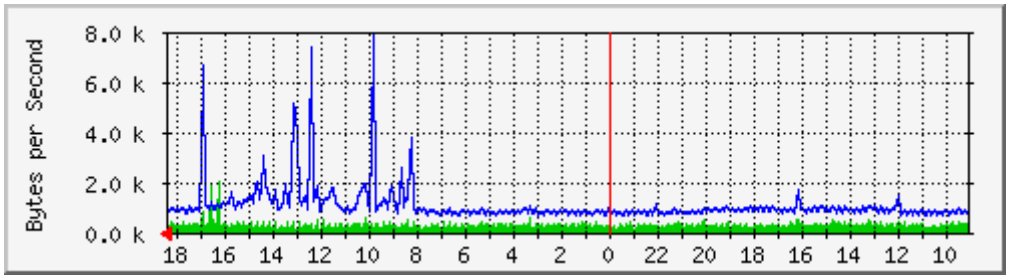
System: MC217A-EDU-SVR-XRN in MC-217A

Description: RMON-Port-02-on-unit-1

ifType: ethernetCsmacd (6)

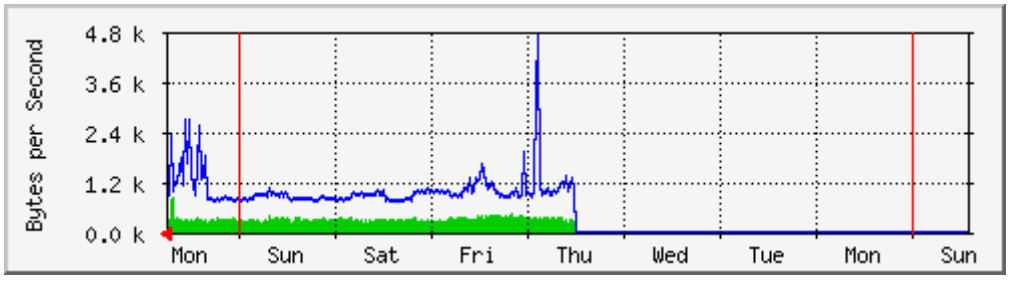
Max Speed: 125.0 MBytes/s

`Daily' Graph (5 Minute Average)



	Max	Average	Current
In	2078.0 B/s (0.0%)	323.0 B/s (0.0%)	200.0 B/s (0.0%)
Out	7897.0 B/s (0.0%)	1036.0 B/s (0.0%)	751.0 B/s (0.0%)

`Weekly' Graph (30 Minute Average)



	Max	Average	Current

In	828.0 B/s (0.0%)	324.0 B/s (0.0%)	323.0 B/s (0.0%)
Out	4703.0 B/s (0.0%)	990.0 B/s (0.0%)	863.0 B/s (0.0%)

Observation

3COM core switch port 2 is connected with Core Switch EDU SVR. All Edu Servers in the Edu domain are connected with EDU SVR. Core switch EDU SVR is also connected with another Edu RSR switch to which all baseline switches at each floor of the building are connected. Main core switch is also connected with Fortigate FTG 300A router device and ADM SVR core switch.

It control the network traffic from remote sites to "Edu and Admin" sub network and vice versa, also transfers network packets between those two sub networks. A variation in the graphs is due to the many services and servers Edu & Admin are available on the network and used at different timings.

We can observe from the graph that traffic out is higher then traffic in because usage of network for the resources including educational applications in the Edu domain.

Traffic Analysis at 3COM-Core-Switch 3C4924 Port 1

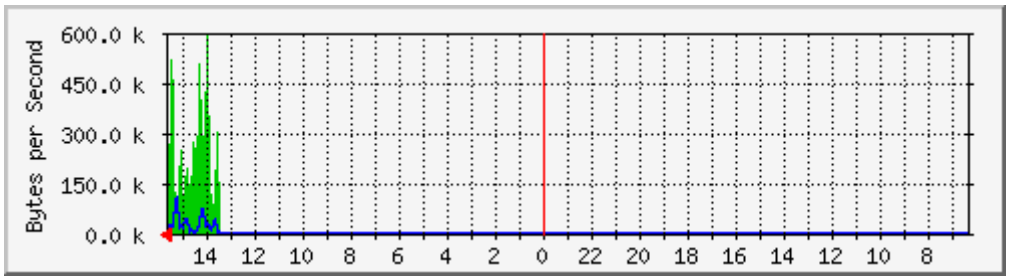
System: MC-ADM-SVR-XRN in MC-217A

Description: RMON-Port-01-on-unit-1

ifType: ethernetCsmacd (6)

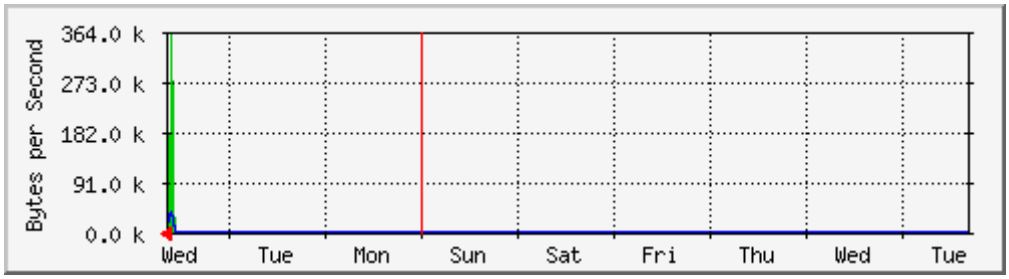
Max Speed: 12.5 MBytes/s

`Daily' Graph (5 Minute Average)



	Max	Average	Current
In	596.3 kB/s (4.8%)	265.8 kB/s (2.1%)	202.6 kB/s (1.6%)
Out	106.2 kB/s (0.8%)	28.1 kB/s (0.2%)	12.4 kB/s (0.1%)

`Weekly' Graph (30 Minute Average)



	Max	Average	Current
In	360.8 kB/s (2.9%)	223.5 kB/s (1.8%)	276.8 kB/s (2.2%)
Out	46.8 kB/s (0.4%)	24.1 kB/s (0.2%)	46.8 kB/s (0.4%)

Observation

In this graph we are capturing traffic on port 1 of the ADM SVR 3COM core switch is connected to the main 3COM core switch and servers in Admin Domain. Core Switch is also connected to ADM riser switch to which all baseline switches at each floor of the building are connected and serves the staff members in the admin domain.

The network traffic from admin domain to Edu domain is flow from this core switch, this core switch is also designated for remote users through fortigate FTG-300A device to the admin servers. Students are restricted to use only Edu side domain. They are not allowed using admin domain. Traffic-in (green) is higher then traffic out (blue) is showing that usage from Edu domain and remote sites are higher then the data retrieval.

Traffic Analysis At Domain Controller, File And Mail Servers

Traffic Analysis for 16777219 – Admin Domain Controller

Traffic Analysis for 65539 – DC11

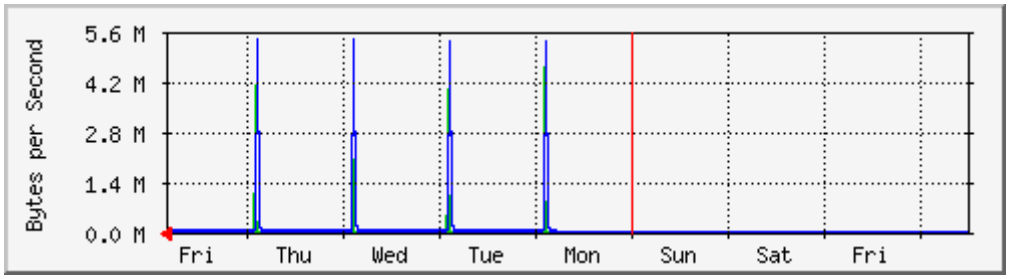
System: DC11 in

ifType: ethernetCsmacd (6)

Max Speed: 12.5	In	4353.2 kB/s (3.5%)	65.9 kB/s (0.1%)	14.1 kB/s (0.0%)	MBytes/s
Ip: 10.214.240.3	Out	5454.5 kB/s (4.4%)	76.4 kB/s (0.1%)	15.9 kB/s (0.0%)	

(mail1.norquest.ca)

`Weekly' Graph (30 Minute Average)



Max **Average** **Current**

Observation

College Domain Controller which is DC11 has Windows 2000 Server installed. It is located at the admin sub-network for authentication to office employees only. The Server runs the login scripts for user login once they get connection. DNS server is also configured on this machine for the local resources and also as forwarder for the external resources.

High Spikes every day excluding weekends shows that in the early office hours users login to the server from their workstations and their login scripts executes, on particular time server's Ethernet Interface transfers high volume of packets, rest of the day no significant activity happens at the server results very low volume of traffic. On Saturday and Sunday limited users access the server produce low traffic which is close to zero.

Traffic Analysis for 65539 – File and Print server

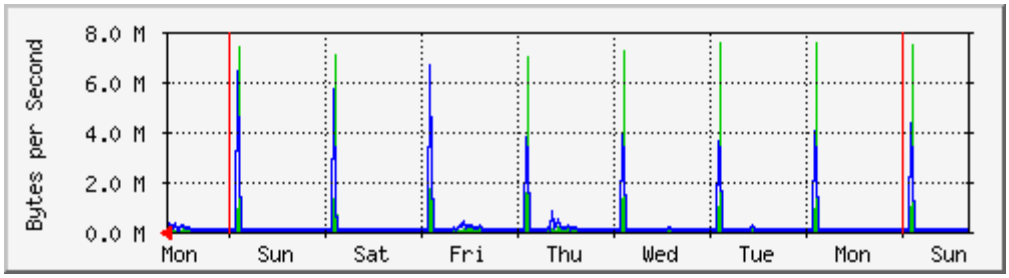
System: FAP21 in

ifType: ethernetCsmacd (6)

Max Speed: 12.5 MBytes/s

Ip: 10.214.240.11 (mail1.norquest.ca)

`Weekly' Graph (30 Minute Average)



	Max	Average	Current
In	7580.5 kB/s (6.1%)	196.8 kB/s (0.2%)	207.5 kB/s (0.2%)
Out	6601.7 kB/s (5.3%)	171.0 kB/s (0.1%)	357.4 kB/s (0.3%)

Observation

File and Print Server is FAP21 on this machine Windows 2003 server installed. This located at the EDU sub network give access to the sharable resources available on the server to the Edu which is dedicated to student & instructors users only. This server is also performing as a print server allow sharable printing to different groups of students and instructors, also few applications such as success maker and plato is also installed on this server for small group of students use. Student's home directory resides on this server which is mapped automatically after the login scripts executes

Spikes including weekends is showing that in the early office hours students login to the server from their workstations and their login scripts executes which mapped the home directory, this particular time server's Ethernet interface transfers high volume of packets, rest of the day very low activity performed on the server including file and print access. On Saturday and Sunday students and instructors still access the server locally and remotely.

Traffic Analysis for 16777220 – MAIL Server

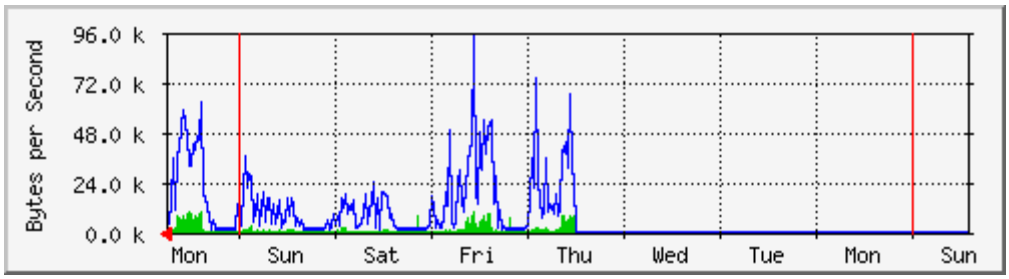
MAIL01 in

ifType: ethernetCsmacd (6)

Max Speed: 12.5 MBytes/s

Ip: 10.214.240.11 (mail1.norquest.ca)

`Weekly' Graph (30 Minute Average)



	Max	Average	Current
In	9952.0 B/s (0.1%)	1786.0 B/s (0.0%)	728.0 B/s (0.0%)
Out	93.2 kB/s (0.7%)	14.5 kB/s (0.1%)	5953.0 B/s (0.0%)

Observation

Mail01 is the Mail Server of the college. Windows2000 server and Microsoft Exchange 2000 are installed. This is located at Admin sub network. On this server all the staff mail boxes are created.

Graph shows high variation during each day including weekends that shows users continuously connects and access their mail boxes, meeting schedules and calendars, results server's Ethernet interface continuously transfers high volume of packets during the day. We can observe low activity in graph during the Saturday, Sunday (limited users access remotely) and no activity from midnight to early morning. There is a gap between blue and green lines it's because many users are receiving the mails and some users are actually sending mails.

Conclusion

To monitor the network traffic graphically in the MINT lab with MRTG monitoring tool we have setup design and implement network of several routers. The protocol configured on this network is OSPF. We have divided this network into three areas. The reason of dividing network into three areas is that router within an area must maintain a database for the area to which it belongs. The router doesn't have detailed information about network topology outside of its area that is why the size of its database is reduced. On this network router B elected as DR and router C is BDR. As we have mentioned in chapter 2. On this network we have also setup one switch. Router A, router B and routers C are directly connected to the switch. Router D is directly connected to router B and router C. Router E is connected to router B and router F is connected to router A. Router A and router E are AS in this network.

After implementation of the network based on OSPF protocols. We installed MRTG on Windows 2003 Server now the main task for us to configure MRTG on the Server and capture data from the router port. As MRTG was new for us and there was several difficulties to configure and implement MRTG in a proper way. The first step we performed to create basic MRTG.cfg and then convert it to advance level configuration file in which we configure TCP and UDP OIDs to capture data. We setup different MRTG.cfg file because we want to capture data from different ports of the network to see how MRTG works in the large networks. To send different traffic to the network FTP Server, Proxy Server and media streaming were setup and also I-DTG traffic generator was used. To capture traffic and generate graph in MRTG SNMP should be configured on all routers which we want to monitor. SNMP is an application layer protocol that facilitates the exchange of management information between network devices. It is a part of the Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite. To execute MRTG there are two options one is we can run MRTG after every 5 minutes to update graph or run MRTG all the time. In our project we have configured MRTG to run all the time so that graphs update automatically. During monitoring traffic through graphs generated by MRTG. We vary the traffic load on the network and also inject different size of the data so that we can see how MRTG behave and analyze its accuracy. We analyze that MRTG is very accurate tool available nowadays. Data inject with variation MRTG accurately showed the variation in the graph the only limitation analyzed is that it update the graph after 5 minutes which should one of the disadvantage in MRTG.

In this project have also configured RRDTool which is more complex and difficult to configure as compared to MRTG. RRDtool provide better interface to the user. As RRDtool have many features and capabilities as compare to MRTG. The data storage algorithm is much more efficient, yet much harder to grasp than those of MRTG. When configure RRDTool and make changes in the mrtg.cfg we are actually replacing *rateup* with the RRDtool PERL module *RRDs.pm*. MRTG will take all old .log files and convert them to .rrd format. The .log files don't get touched in the process, so if things don't work out they are still there. MRTG will use rrdtool to update its databases. These will have a new format called *rrd* which is totally different than

the native *log* format of the classic MRTG. MRTG will not create any webpages of graphs anymore. It will only query the routers for traffic information and update its *rrd* databases. To see RRDTool results we have to configure script files. In our project we setup two script file 14all.cgi and routers2.cgi files. To run these script file we have to configure web server IIS or Apache. We have configured IIS Server. When we run these scripts we can see many options on the index page and the graph is much better than MRTG alone.

MRTG reports the amounts of traffic sent or receive by a router interface. It collects its data from the router Management Information Base (MIB) using SNMP, generating a reading after every 5 minutes. Given the capacity of the link, the MRTG data allows us to compute the average available bandwidth every 5 minutes. Despite its low resolution, the MRTG data is the most accurate way to verify the available bandwidth estimation tools.

References: http://www.mrtg.org

http://oss.oetiker.ch/rrdtool

http://net-snmp.sourceforge.net