Performance & Monitoring Network

Muhammad Zen Samsono Hadi, ST. MSc.

Functions of Network Management

- **Fault management**
  - Network state monitoring
  - Failure logging, reporting and tracking etc.
- **Configuration management**
  - device and software configuration
  - version control (compare, apply and rollback, backup) etc.
- **Accounting management**
  - billing and traffic measurement etc.
- **Performance management**
- **Security Management**
  - Access control, worm/attack detection and alert etc.

Performance Management-Why

- **Why needed and important?**
  - **Capacity planning**
    - **when do we need to upgrade our link and device?**
  - **Ensure network availability**
  - **Verify network performance, verify QoS (we expected)**
  - **Ensure SLA compliance (customer expected)**
  - **Better understanding and control of network**
  - **Optimization, make the network runs better!**
- **Proactive or reactive?**
  - **Know problem before users and boss**
  - **Solve the problem before their complain**

  Or
  - **Wait for problem to happen, and customers complain?**

  - **As a NOC, we should be proactive, NOC means NO Complain!**

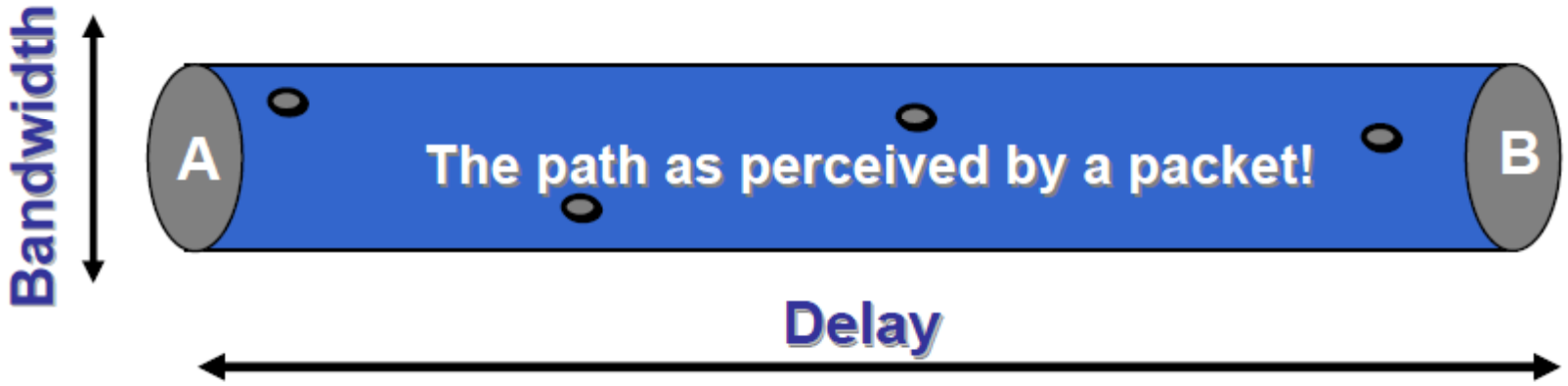
Performance Management-What

- **What's performance management?**
  - **understanding the behavior of a network and its elements in response to traffic demands**
  - **Measuring and reporting of network performance to ensure that performance is maintained at a acceptable level**

Performance Management-How

- How to measure the network performance
  - Delay, jitter, packet loss, bandwidth usage etc.
- The steps and process of performance management:
  - Data collection
  - Baseline the network
  - Determining the threshold for acceptable performance
  - Tunning
- Technologies and tools needed
  - Data collection technologies such as: sniffing & netflow
  - QoS
  - Tools: ping, mrtg, iperf, wget, etc.

Analogi Jaringan



- **Bandwidth, dianalogikan pipe**
- **Delay, mempresentasikan panjang pipa**
- **Jitter, variasi delay pada pipa**
- **Loss, menggambarkan kebocoran pada pipa**

Delay (Latency)

- ***Delay*** **adalah waktu yang dibutuhkan oleh sebuah paket data terhitung dari saat pengiriman oleh *transmitter* sampai saat diterima oleh *receiver***
- **Delay untuk komunikasi suara :**

  a. Propagation delay **(*delay* yang terjadi akibat transmisi melalui jarak antar pengirim dan penerima)**

  b. Serialization delay **(*delay* pada saat proses peletakan bit ke dalam *circuit*)**

  c. Processing delay **(*delay* yang terjadi saat proses *coding, compression, decompression* dan *decoding*)**

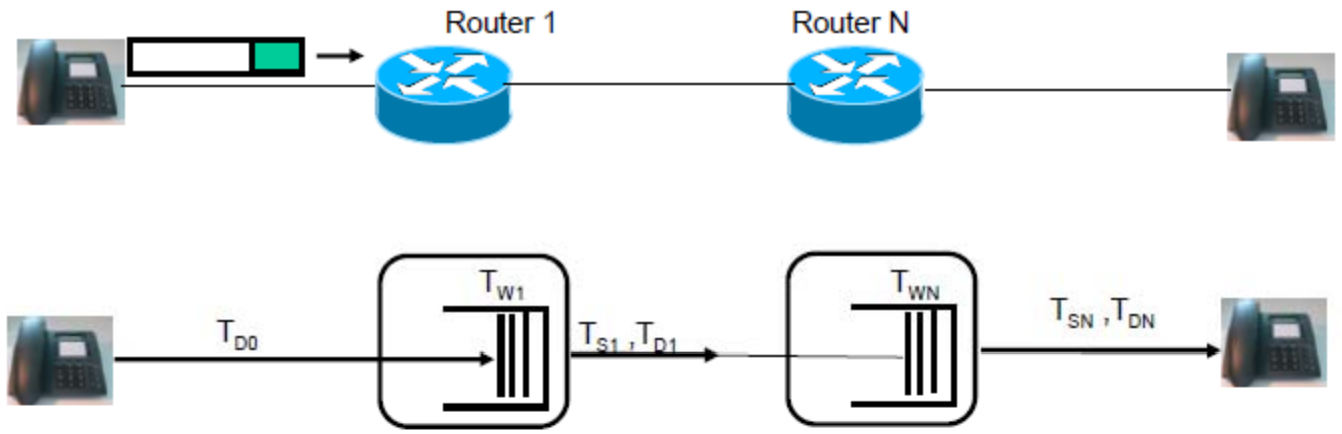
  d. Packetization delay **(*delay* yang terjadi saat proses paketisasi *digital voice sample)***

  e. Queuing delay **(*delay* akibat waktu tunggu paket sampai dilayani)**

  f. Jitter buffer **(*delay* akibat adanya *buffer* untuk mengatasi *jitter*)**

- **Tools: ping, traceroute, tcpdump.**

Perhitungan Delay



• Total delay time of a packet (without CPE) for N links and Routers

$$T_G = T_{D0} + ... + T_{DN} + T_{w1} + ... + T_{wN} + T_{S1} + ... + T_{SN}$$

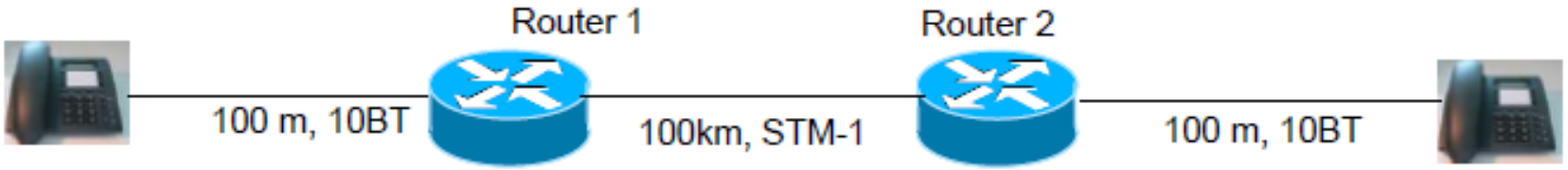
Delay time of physical lines
(**5 usec/km**)

Waiting time
In equipment buffers

Serialisation time t:
Ts=packet length/ output bitrate

- **limit for Voice over IP Roundtrip delay R = 2 * TG = 150 msec (ITU G.113)**

Contoh



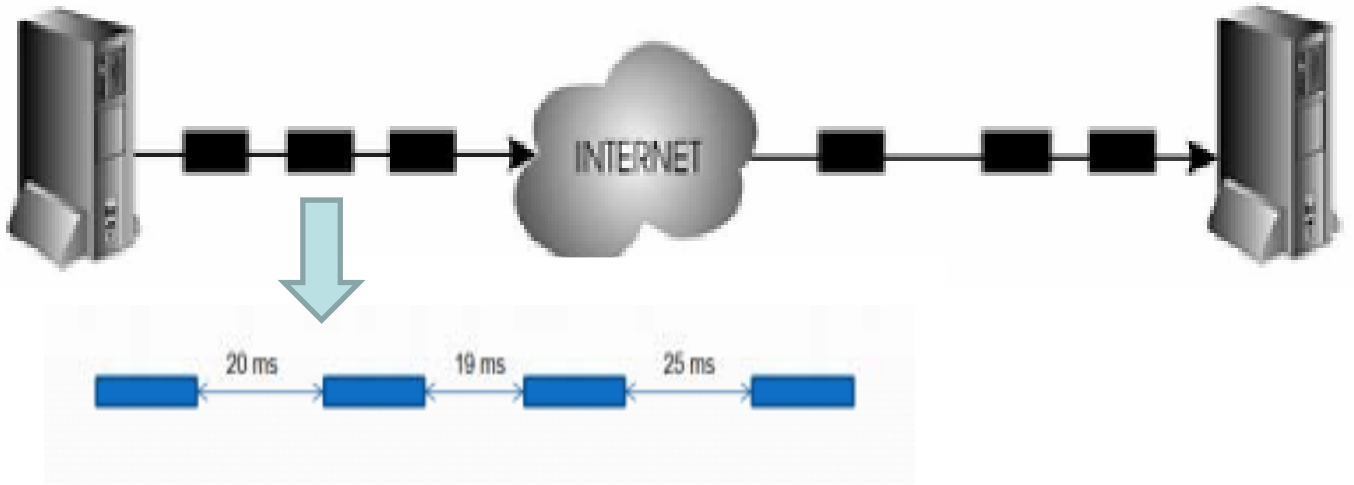
- **VoIP-Phones dihubungkan dengan 10Mbps ke Router. Panjang paket adalah 212 Byte. Kedua Router dihubungkan melalui WAN dengan STM-1 (155Mbps). Pada masing-masing input dan output Router adalah buffer untuk serialization.**

  Waiting time untuk Router adalah 10msec.

  Berapa total end to end delay (tanpa CPE phone) dan roundtrip delay ?

Jitter

- ***Jitter*** **adalah variasi** ***delay*****, yaitu perbedaan selang waktu kedatangan antar paket di terminal tujuan.**
- **Untuk mengatasi** ***jitter*** **maka paket data yang datang dikumpulkan dulu dalam** ***jitter buffer*** **selama waktu yang telah ditentukan sampai paket dapat diterima pada sisi penerima dengan urutan yang benar.**
- **Nilai jitter yang direkomendasikan oleh ITU – T Y.1541 adalah dibawah 50 ms.**
- **Tools: ping, iperf, dll. J1 = abs(t2-t1), J2=abs(t3-t2), ….**



Packet Loss

- ***Packet loss*** **adalah banyaknya paket yang hilang selama proses transmisi ke tujuan.**
  - **Terjadi tabrakan data atau antrian penuh**
  - **Link atau hardware disebabkan CRC error**
  - **Perubahan rute (temporary drop) atau blackhole route (persistent drop)**
  - **Interface or router down**
  - **Misconfigured access-list**
  - **...**
- **1% packet loss tidak dapat digunakan.**
- ***Packet loss*** **dinyatakan dalam persen (%) dengan nilai yang direkomendasikan pada ITU-T Y.1541 tidak boleh lebih dari 0.1 %.**
- **Tools: ping etc.**

$$Packet\ loss\ = \frac{(Packets\ \_\ trasnsmitt\ ed - Packets\ \_\ received\ )}{Packets\ \_\ trasnmitte\ d} x100\%$$

Throughput

- ***Throughput*** **adalah jumlah bit yang diterima dengan sukses perdetik melalui sebuah sistem atau media komunikasi (kemampuan sebenarnya suatu jaringan dalam melakukan pengiriman data).**
- **Throughput diukur setelah transmisi data (host/client) karena suatu sistem akan menambah delay yang disebabkan *processor limitations*, kongesti jaringan, *buffering inefficients*, error transmisi, *traffic loads* atau mungkin desain hardware yang tidak mencukupi.**
- **Aspek utama throughput yaitu berkisar pada ketersediaan bandwidth yang cukup untuk menjalankan aplikasi.**
- **Hal ini menentukan besarnya trafik yang dapat diperoleh suatu aplikasi saat melewati jaringan.**
- **Tool : MRTG, iperf**

$$waktu\_download\_terbaik = \frac{ukuran\_file}{bandwidth}$$

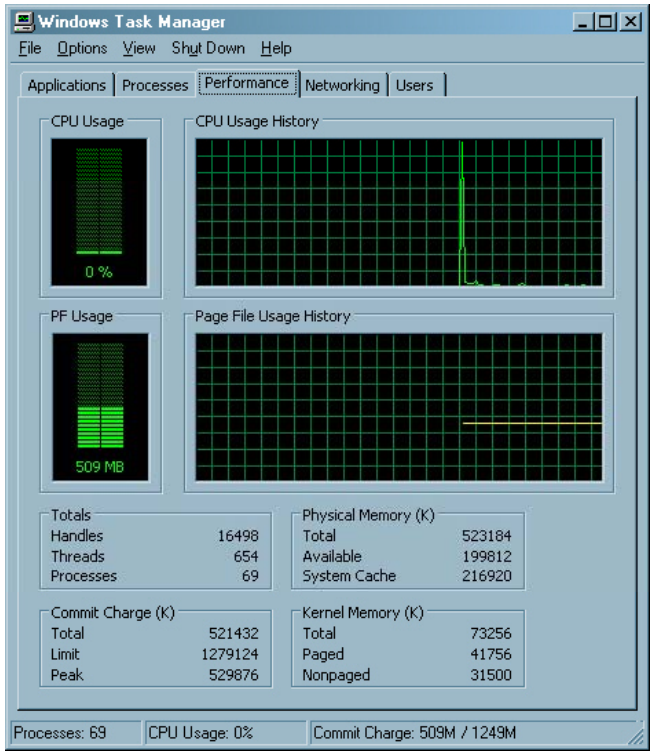
$$waktu\_download\_typical = \frac{ukuran\_file}{throughput}$$

Network Availability

- is the metric used to determine uptime and downtime
- Availability = (uptime)/(total time) = 1-(downtime)/(total time)
- Network availability is the IP layer reachability
- Better > 99.9%
- 99.9%
  - 30x24x60x0.1%=43.3 (Minutes), means the down time should be less than 45 minutes in one month
- 99.99%
  - 30x24x60x0.01%=4.3 (Minutes), means the down time should be less than 5 minutes in one month!
- 99.9% is acceptable for R&E networks (Even 99.0% is acceptable), some commercial ISPs can reach 99.99%

CPU and Memory Utilization

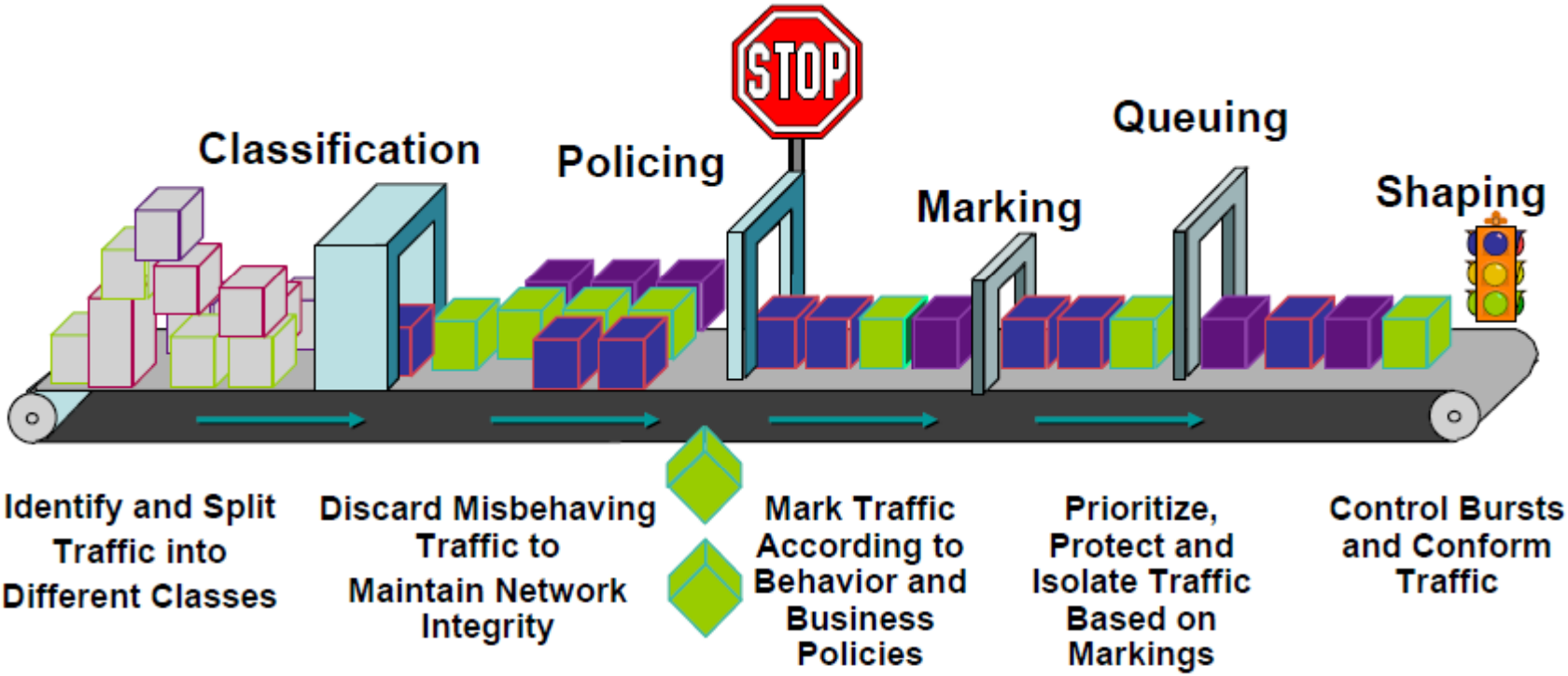
- **We focus on routers**
- **CPU utilization better less than 30%**
- **For global routing routers, at least 512M memory is needed**



QoS

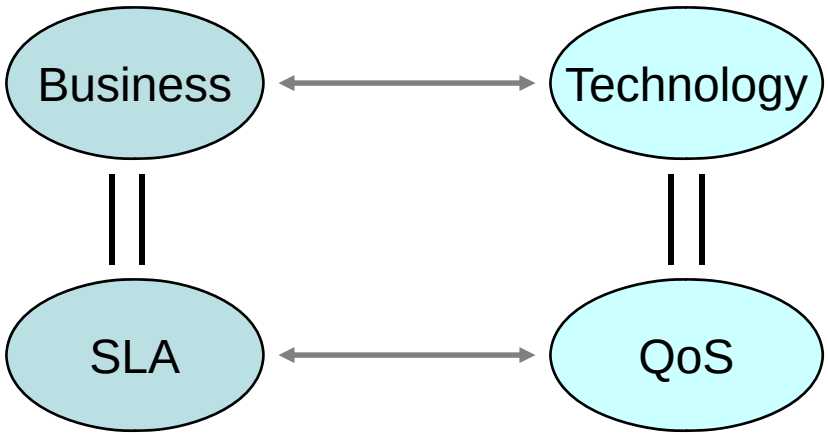
- **QoS: Quality Of Service**
- **QoS is technology to manage network performance**
- **QoS is a set of performance measurements**
  - **Delay, Jitter, packet loss, availability, bandwidth utilization (throughput) etc.**
- **IP QoS: QoS for IP service**

QoS Architecture



SLA and QoS

- **SLA: Service Level Agreement**
- **SLA is the agreement between service provider and customer, SLA defines the quality of the service the service provider delivered, such as delay, jitter, packet loss etc.**
- **SLA is a very important part of the business contract, and also can be used to distinguish the service level of different ISPs**



SLA example: Level 3

Delay

Packet Loss

Availability

Jitter

Bandwidth

	Premium	*Optimized*	*Enhanced*
Latency (one way)			
0 - 250 miles	10ms	10ms	10ms
251 - 500 miles	15ms	15ms	15ms
501 - 1000 miles	20ms	20ms	20ms
1001 - 1500 miles	25ms	25ms	25ms
1501 - 2000 miles	30ms	30ms	30ms
2001 - 2500 miles	35ms	35ms	35ms
2501+ miles	45ms	45ms	45ms
Trans-Atlantic	45ms	45ms	45ms
Packet Delivery	100%	100% of CIR*	99.95%
Availability, Single Port	99.98%	99.98%	99.98%
Availability, Dual Port	99.998%	99.998%	99.998%
Jitter	2ms	10ms	NA
Bandwidth	100%	100% of CIR*	NA

SLA example: Sprintlink

	Delay	Packet loss	Availability	Jitter
North America	55 ms	0.30%	99.90%	2 ms
Europe	44 ms	0.30%	99.90%	2 ms
Asia	105 ms	0.30%	99.90%	2 ms
South pacific	70 ms	0.30%	99.90%	2 ms
Continental US (Peerless IP)	55ms	0.1%	n/a	2 ms

Measurement Technology

- **We’ve known what metrics used to describe network performance, but how to measure them?**
- **Technologies and tools**
  - **ping, traceroute,iperf, jperf.**
  - **SNMP**
  - **Netflow (Cisco), Sflow (Juniper), NetStream (Huawei)**
  - **IP SLA (Cisco)**
  - **Etc.**

Active Measurement Tools

- **Tools that inject packets into the network to measure some value**
  - **Available Bandwidth**
  - **Delay/Jitter**
  - **Loss**
- **Requires bi-directional traffic or synchronized hosts**

Passive Measurement Tools

- Tools that monitor existing traffic on the network and extract some information
  - Bandwidth used
  - Jitter
  - Loss rate
- May generate some privacy and/or security concerns

ping

- **Normally used as a troubleshooting tool**
- **Uses ICMP Echo messages to determine:**
  - **Whether a remote device is active (for trouble shooting)**
  - **round trip time delay (RTT), but not one-way delay**
  - **Packet loss**
- **Sometime we need to specify the source and length of packet using extended ping in router or host**
  - ***Why using large packet when ping?***

    (to test the link quality and throughput.)
  - ***Large packet ping is prohibited in Windows, but Linux is ok***

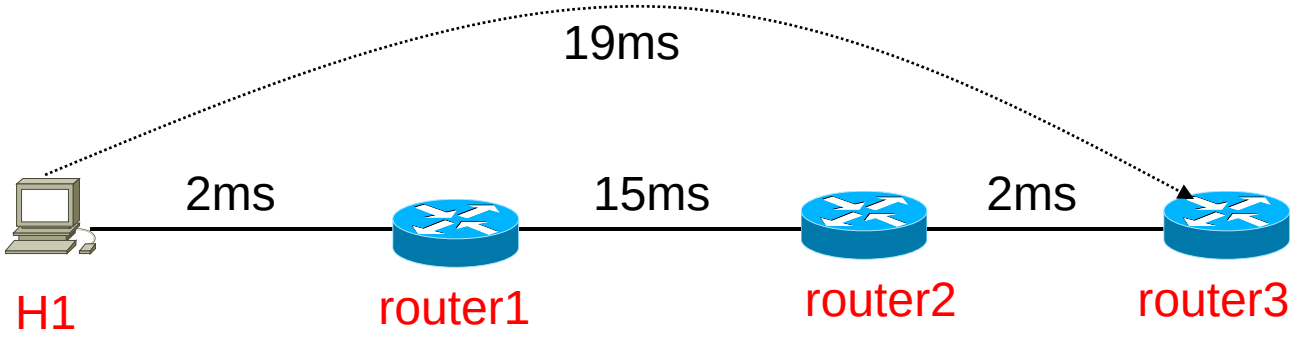
Sample Ping

Freebsd>% ping 202.112.60.31
PING 202.112.60.31 (202.112.60.31) 56(84) bytes of data.
64 bytes from 202.112.60.31: icmp_seq=1 ttl=253 time=0.326 ms
……
64 bytes from 202.112.60.31: icmp_seq=6 ttl=253 time=0.288 ms
6 packets transmitted, 6 received, 0% packet loss, time 4996ms
rtt min/avg/max/mdev = 0.239/0.284/0.326/0.025 ms

router# ping
Protocol [ip]:
Target IP address: 202.112.60.31
Repeat count [5]:
Datagram size [100]: 3000
Timeout in seconds [2]:
Extended commands [n]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 3000-byte ICMP Echos to 202.112.60.31, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms

traceroute

- **Can be used to measure the RTT delay, and also the delay between the routers along the path**
- **Unix/linux traceroute uses UDP datagram with different TTL to discover the route a packet take to the destination, Microsoft Windows tracert uses ICMP protocol, If Windows tracert appears to show continuous timeouts, the router may be filtering ICMP traffic – try a Unix/Linux traceroute**
- **After the Nachi worm, many ISPs filter ICMP traffic. So ping can not work, but traceroute is ok**



Sample Traceroute

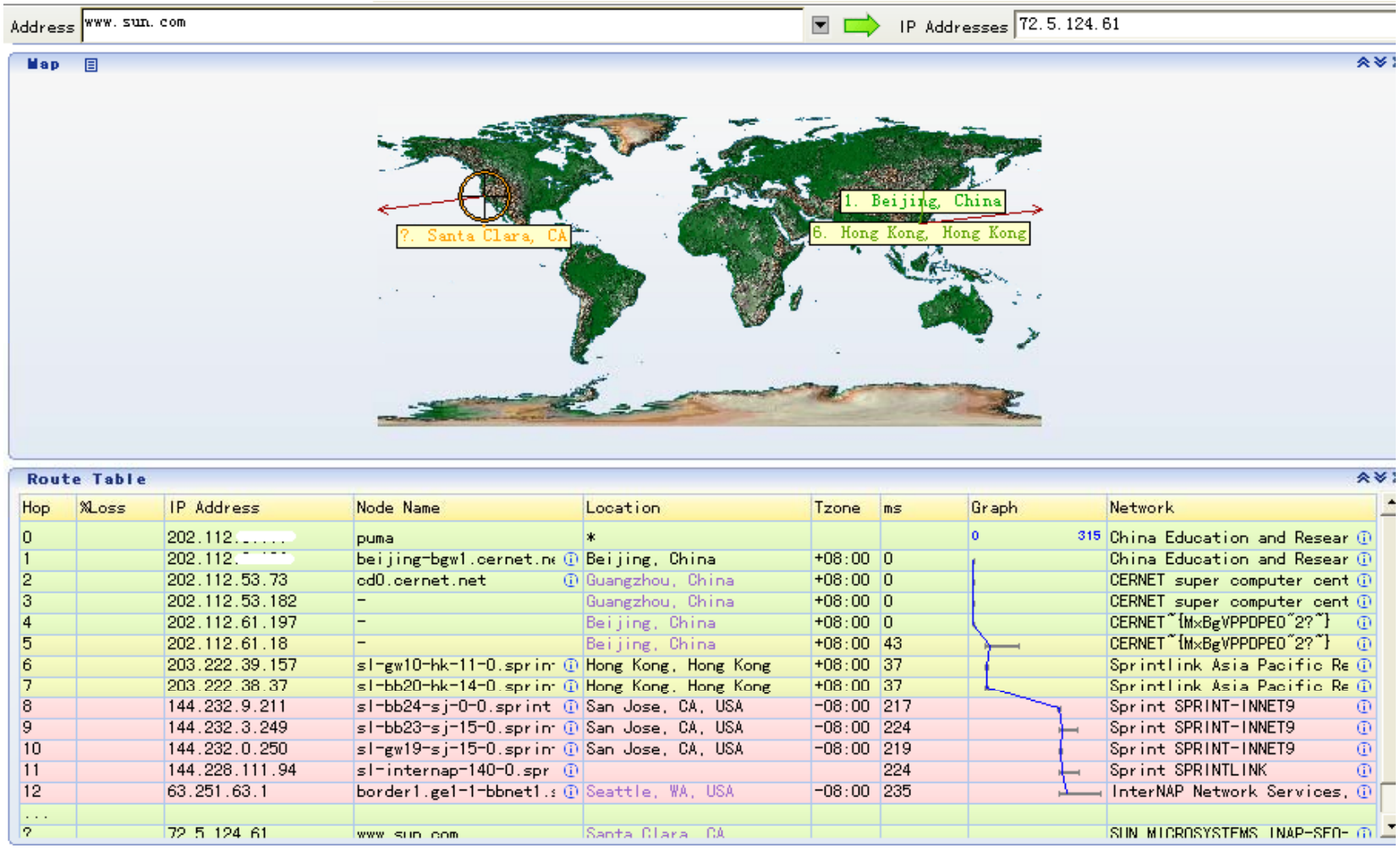
```
C:\WINDOWS>tracert -d www.163.com

Tracing route to www.cache.split.netease.com [202.108.9.16]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  202.112.0.126
  2    24 ms    14 ms    <1 ms  202.112.53.73
  3    <1 ms    <1 ms    <1 ms  202.112.53.178
  4    <1 ms    <1 ms    <1 ms  202.38.123.18
  5    <1 ms    <1 ms    <1 ms  202.38.123.10
  6   125 ms   124 ms   123 ms  219.158.28.73
  7   125 ms   129 ms   130 ms  219.158.11.121
  8   130 ms   132 ms   132 ms  202.96.12.162
  9   125 ms   123 ms   122 ms  202.106.193.38
 10   122 ms   121 ms   126 ms  202.106.193.166
 11   121 ms   123 ms   122 ms  61.148.3.234
 12   125 ms   127 ms   127 ms  202.108.9.16

Trace complete.
```

Router# traceroute 202.112.60.37
Type escape sequence to abort.
Tracing the route to 202.112.60.37

1	202.112.53.169	0 msec	0 msec	0 msec
2	202.112.36.250	20 msec	20 msec	16 msec
3	202.112.36.254	28 msec	28 msec	24 msec
4	202.112.53.202	24 msec	*	24 msec

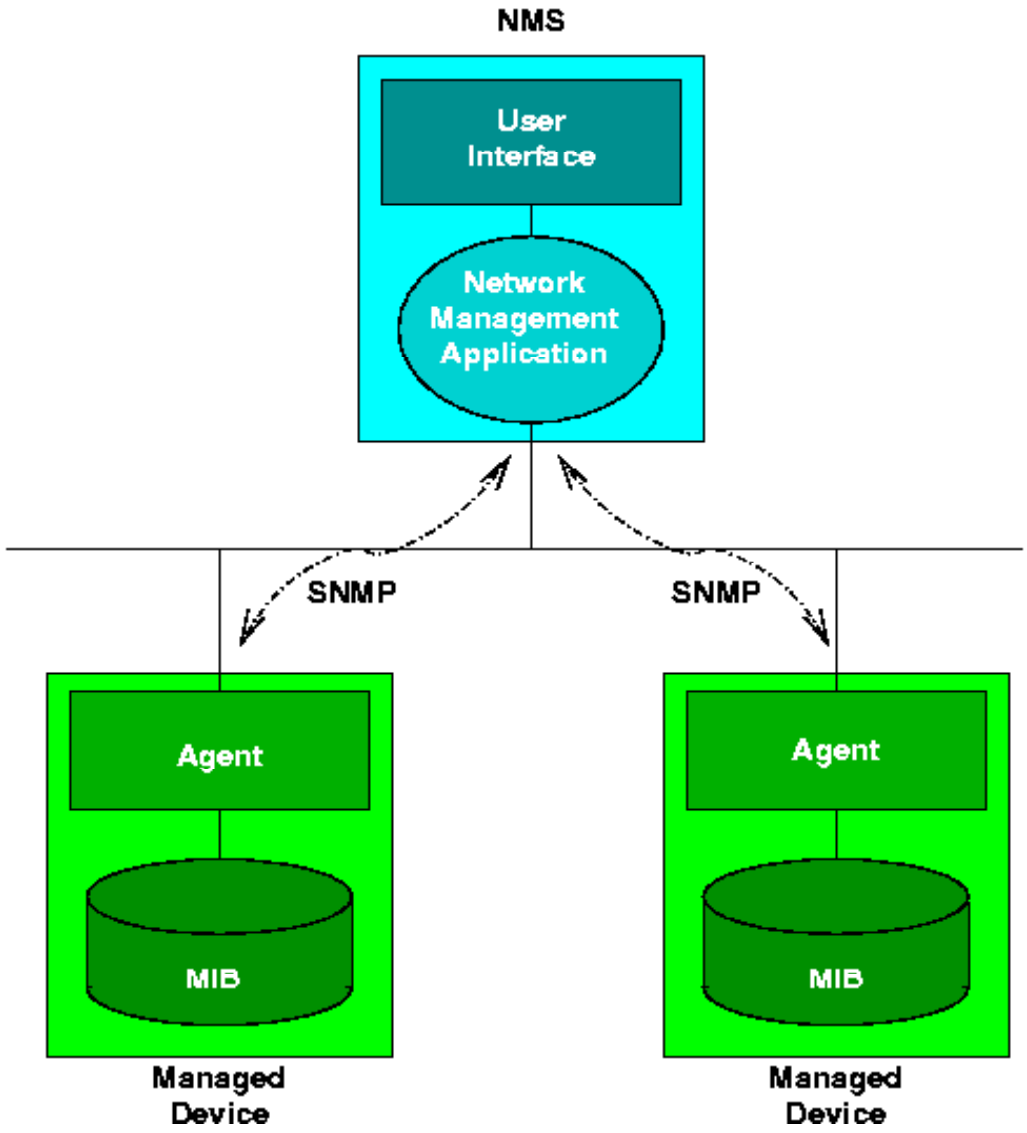
Visual Route

- **Visualization of traceroute information**
- **http://www.visualroute.com**



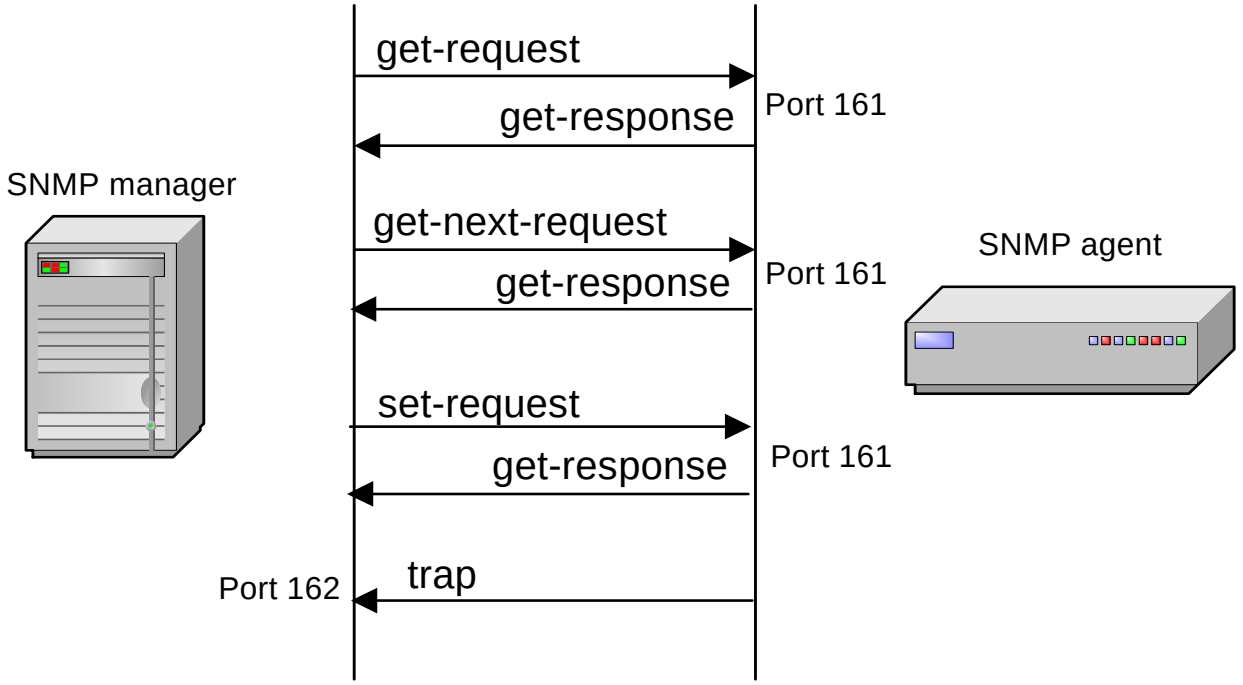
Hop	%Loss	IP Address	Node Name	Location	Tzone	ms	Graph	Network
0		202.112.[redacted]	puma	*			0 315	China Education and Resear
1		202.112.[redacted]	beijing-bgw1.cernet.ne	Beijing, China	+08:00	0		China Education and Resear
2		202.112.53.73	cd0.cernet.net	Guangzhou, China	+08:00	0		CERNET super computer cent
3		202.112.53.182	-	Guangzhou, China	+08:00	0		CERNET super computer cent
4		202.112.61.197	-	Beijing, China	+08:00	0		CERNET~{MxBgVPPDPEO~2?~}
5		202.112.61.18	-	Beijing, China	+08:00	43		CERNET~{MxBgVPPDPEO~2?~}
6		203.222.39.157	sl-gw10-hk-11-0.sprin	Hong Kong, Hong Kong	+08:00	37		Sprintlink Asia Pacific Re
7		203.222.38.37	sl-bb20-hk-14-0.sprin	Hong Kong, Hong Kong	+08:00	37		Sprintlink Asia Pacific Re
8		144.232.9.211	sl-bb24-sj-0-0.sprint	San Jose, CA, USA	-08:00	217		Sprint SPRINT-INNET9
9		144.232.3.249	sl-bb23-sj-15-0.sprin	San Jose, CA, USA	-08:00	224		Sprint SPRINT-INNET9
10		144.232.0.250	sl-gw19-sj-15-0.sprin	San Jose, CA, USA	-08:00	219		Sprint SPRINT-INNET9
11		144.228.111.94	sl-internap-140-0.spr			224		Sprint SPRINTLINK
12		63.251.63.1	border1.ge1-1-bbnet1.s	Seattle, WA, USA	-08:00	235		InterNAP Network Services,
...								
?		72.5.124.61	www.sun.com	Santa Clara, CA				SUN MICROSYSTEMS INAP-SFO-

SNMP Architecture



SNMP Protocol

- **C/S based, Client Pull and Server Push**
- **Ports: UDP 161(snmp messages), UDP 162(trap messages)**
- **SNMP manager and an SNMP agent communicate using the SNMP protocol**
  - **Generally: Manager sends queries and agent responds**
  - **Exception: Traps are initiated by agent.**

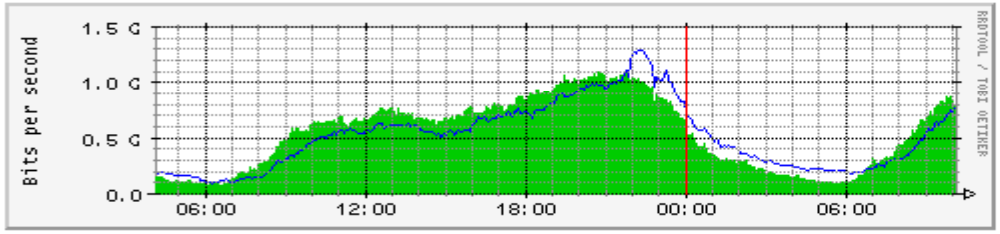


MRTG

- **The Multi Router Traffic Grapher: a freeware written in Perl, works on unix/linux, graph data collected from routers and other devices or applications based on SNMP.**
- **One of most popular network monitoring tools used today: to monitoring the bandwidth utilization of network link**
- **SNMP v2c support, no more counter wrapping**
- **http://oss.oetiker.ch/mrtg/**

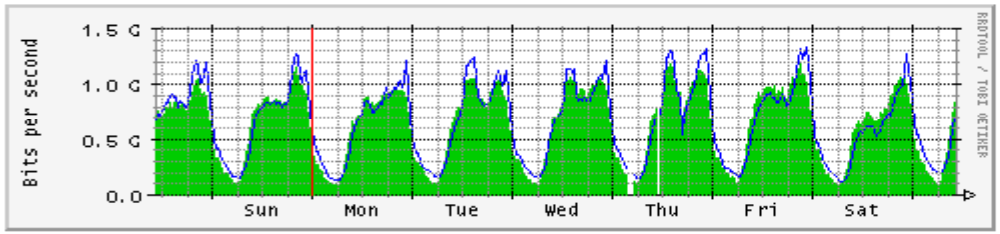
MRTG Example

`Daily' Graph (5 Minute Average)



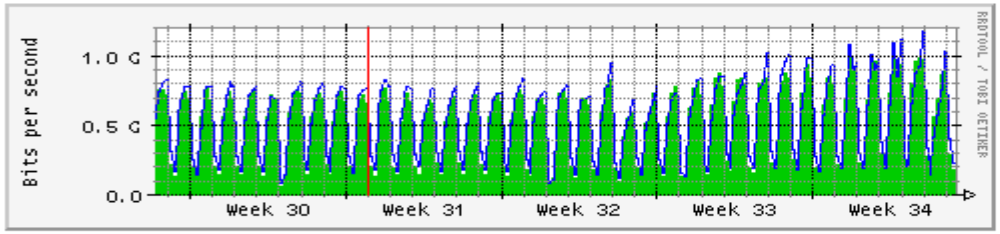
最大 In: 1087.6 Mb/s (43.5%) 平均 In: 537.7 Mb/s (21.5%) 当前 In: 797.4 Mb/s (31.9%)
最大 Out: 1284.4 Mb/s (51.4%) 平均 Out: 512.9 Mb/s (20.5%) 当前 Out: 764.9 Mb/s (30.6%)

`Weekly' Graph (30 Minute Average)



最大 In: 1185.3 Mb/s (47.4%) 平均 In: 638.9 Mb/s (25.6%) 当前 In: 841.1 Mb/s (33.6%)
最大 Out: 1338.4 Mb/s (53.5%) 平均 Out: 656.3 Mb/s (26.3%) 当前 Out: 699.1 Mb/s (28.0%)

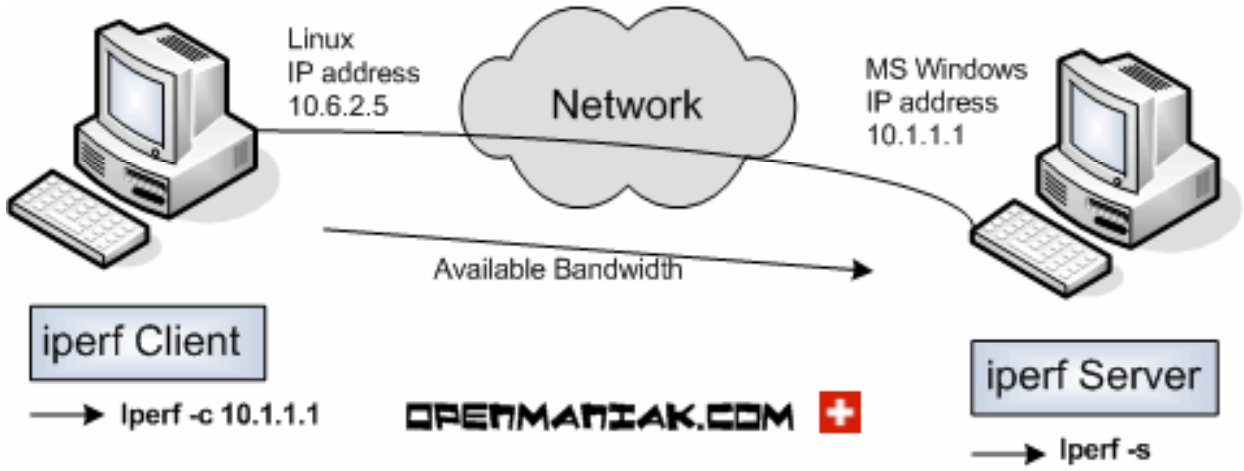
`Monthly' Graph (2 Hour Average)



最大 In: 1036.8 Mb/s (41.5%) 平均 In: 538.1 Mb/s (21.5%) 当前 In: 674.9 Mb/s (27.0%)
最大 Out: 1174.0 Mb/s (47.0%) 平均 Out: 560.6 Mb/s (22.4%) 当前 Out: 529.4 Mb/s (21.2%)

IPerf

- **Client/server application that**
  - **–Measures maximum TCP performance**
  - **–Facilitates tuning of TCP and UDP parameters**
  - **–Reports bandwidth, jitter, and packet loss**
- **http://dast.nlanr.net/Projects/Iperf/**



Contoh iperf

➔ Client side:

#iperf -c 10.1.1.1 -u -b 10m

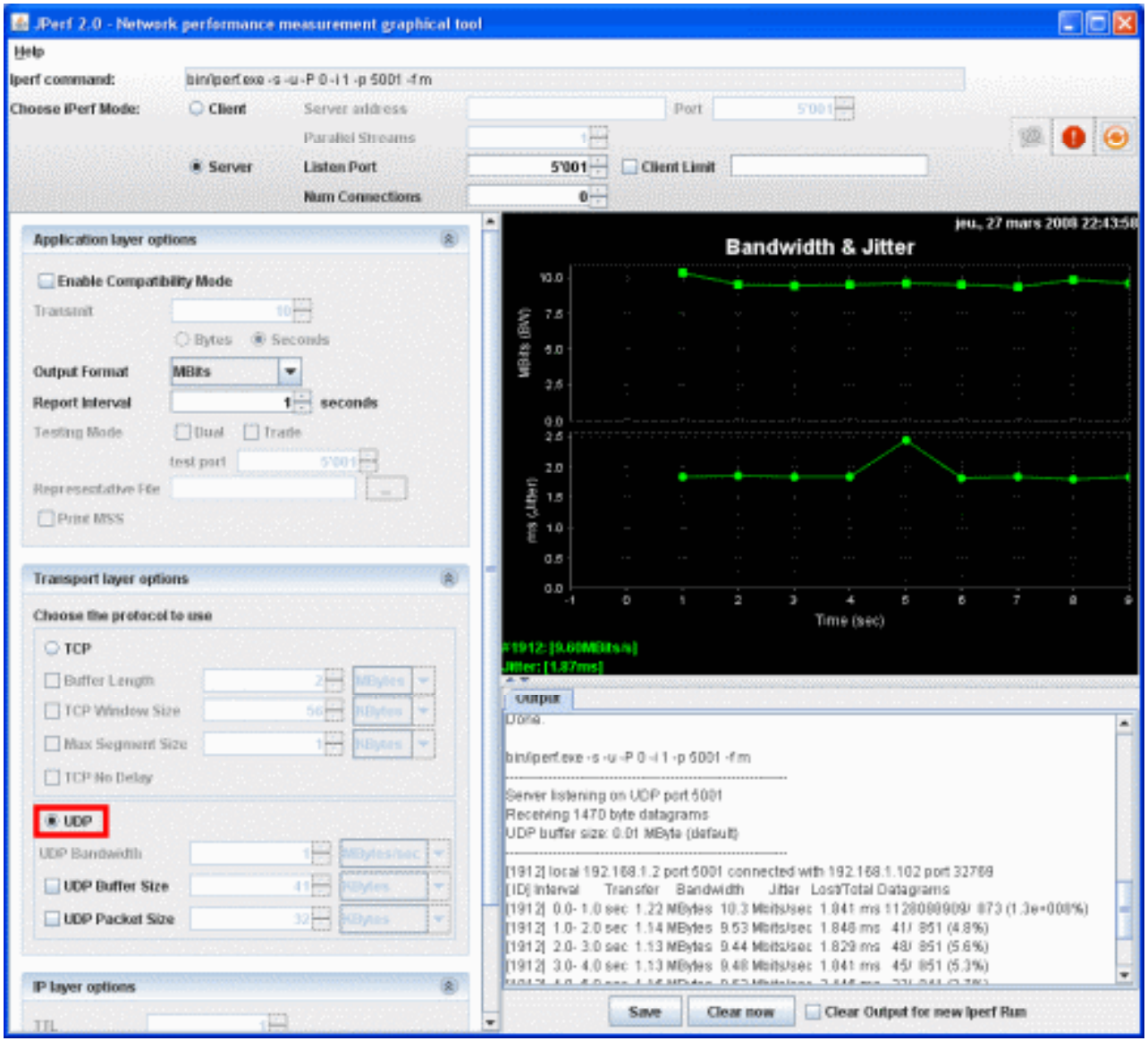
```
------------------------------------------------------------
Client connecting to 10.1.1.1, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size: 108 KByte (default)
------------------------------------------------------------
[ 3] local 10.6.2.5 port 32781 connected with 10.1.1.1 port 5001
[ 3]  0.0-10.0 sec  11.8 MBytes  9.89 Mbits/sec
[ 3] Sent 8409 datagrams
[ 3] Server Report:
[ 3]  0.0-10.0 sec  11.8 MBytes  9.86 Mbits/sec  2.617 ms  9/ 8409  (0.11%)
```

➔ Server side:

#iperf -s -u -i 1

```
------------------------------------------------------------
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 8.00 KByte (default)
------------------------------------------------------------
[904] local 10.1.1.1 port 5001 connected with 10.6.2.5 port 32781
[ ID] Interval       Transfer     Bandwidth        Jitter   Lost/Total Datagrams
[904]  0.0- 1.0 sec  1.17 MBytes  9.84 Mbits/sec  1.830 ms  0/ 837  (0%)
[904]  1.0- 2.0 sec  1.18 MBytes  9.94 Mbits/sec  1.846 ms  5/ 850  (0.59%)
[904]  2.0- 3.0 sec  1.19 MBytes  9.98 Mbits/sec  1.802 ms  2/ 851  (0.24%)
[904]  3.0- 4.0 sec  1.19 MBytes  10.0 Mbits/sec  1.830 ms  0/ 850  (0%)
[904]  4.0- 5.0 sec  1.19 MBytes  9.98 Mbits/sec  1.846 ms  1/ 850  (0.12%)
[904]  5.0- 6.0 sec  1.19 MBytes  10.0 Mbits/sec  1.806 ms  0/ 851  (0%)
[904]  6.0- 7.0 sec  1.06 MBytes  8.87 Mbits/sec  1.803 ms  1/ 755  (0.13%)
[904]  7.0- 8.0 sec  1.19 MBytes  10.0 Mbits/sec  1.831 ms  0/ 850  (0%)
[904]  8.0- 9.0 sec  1.19 MBytes  10.0 Mbits/sec  1.841 ms  0/ 850  (0%)
[904]  9.0-10.0 sec  1.19 MBytes  10.0 Mbits/sec  1.801 ms  0/ 851  (0%)
[904]  0.0-10.0 sec  11.8 MBytes  9.86 Mbits/sec  2.618 ms  9/ 8409  (0.11%)
```

Jperf


JPerf 2.0 - Network performance measurement graphical tool
Help
Iperf command:
bin\iperf.exe -s -u -P 0 -i 1 -p 5001 -f m
Choose iPerf Mode:
Client
Server
Server address
Parallel Streams
Listen Port
5001
Client Limit
Num Connections
0
Application layer options
Enable Compatibility Mode
Transmit
Bytes
Seconds
Output Format
MBits
Report Interval
1
seconds
Testing Mode
Dual
Trade
test port
Representative File
Print MSS
Transport layer options
Choose the protocol to use
TCP
Buffer Length
TCP Window Size
Max Segment Size
TCP No Delay
UDP
UDP Bandwidth
UDP Buffer Size
UDP Packet Size
IP layer options
TTL
Bandwidth & Jitter
Time (sec)
Output
bin\iperf.exe -s -u -P 0 -i 1 -p 5001 -f m
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 0.01 MByte (default)
[1912] local 192.168.1.2 port 5001 connected with 192.168.1.102 port 32769
[ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[1912] 0.0- 1.0 sec 1.22 MBytes 10.3 MBits/sec 1.041 ms 1128089809/ 873 (1.3e+008%)
[1912] 1.0- 2.0 sec 1.14 MBytes 9.53 MBits/sec 1.848 ms 41/ 851 (4.8%)
[1912] 2.0- 3.0 sec 1.13 MBytes 9.44 MBits/sec 1.829 ms 48/ 851 (5.6%)
[1912] 3.0- 4.0 sec 1.13 MBytes 9.48 MBits/sec 1.041 ms 45/ 851 (5.3%)
Save
Clear now
Clear Output for new Iperf Run

Performance Management Process