

# perfs--NAR

## Some Details

Network Startup Resource Center



**ESnet**

ENERGY SCIENCES NETWORK



These materials are licensed under the Creative Commons Attribution-NonCommercial 4.0 International license (<http://creativecommons.org/licenses/by-nc/4.0/>)

# Bandwidth Testing vs. Latency

## perfSONAR Toolkit version 3.3.2

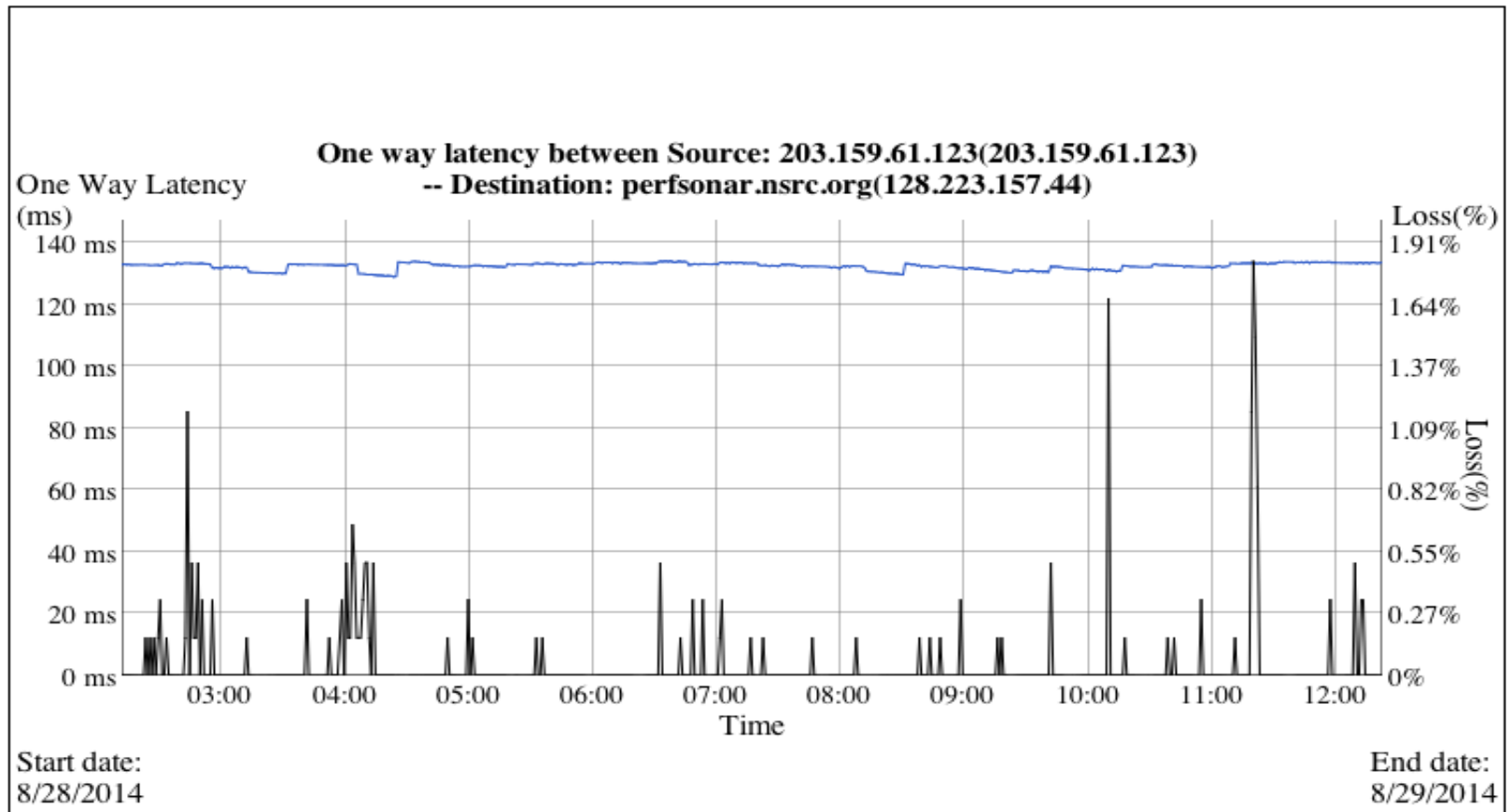
- Select either bandwidth testing or latency – Not both
- Bandwidth testing will affect latency results

perfSONAR

Scale Y axis from 0  Show Reverse Direction Data

### Graph Key (Src-Dst)

- Max delay
- Min delay
- Loss
- Third Quartile
- Median
- First Quartile



# PerfSONAR-PS Where/How to Run

- Does not work well to run perfSONAR in a VM
- Be sure that NTP services are properly configured before collecting data.
- Select the machine's timezone at install time and do not change this.
- For 1Gbps or lower connections you can do some bandwidth testing on smaller machines. Otherwise see: <http://fasterdata.es.net/>
- Run perfSONAR-PS where you will be doing work.

# PerfSONAR-PS Troubleshooting

Solving performance issues is ***completely*** non-trivial in most cases...

You have some tools:

- Historical perfSONAR data
- npad
- ndt
- traceroute, tracepath, mtr, reverse traceroute, ping, netstat, lsof, kernel tuning, tcpdump, etc.

# Securing the perfSONAR-PS Host

Runs CentOS Linux (version 6.5)

- Are the running services reasonable?
- What ports are open and in use?
- Is software up-to-date?
- Do you allow root access?
- Are you using ssh keys or passwords?

# Securing the perfSONAR-PS Host

Are the running services reasonable?

Try this on your perfSONAR-PS machine now:

```
$ ps auxwww | less
```

```
root      1525  0.0  0.0  10676  4400 ?        Ss   Aug26   0:00 NetworkManager --pid-
file=/var/run/NetworkManager/NetworkManager.pid
root      1529  0.0  0.0   4792  2284 ?        S    Aug26   0:00 /usr/sbin/modem-manager
root      1537  0.0  0.0   7596   884 ?        Ss   Aug26   0:00 /usr/sbin/wpa_supplicant -c
/etc/wpa_supplicant/wpa_supplicant.conf -B -u -f /var/log/wpa_supplicant.log -P /var/run/wpa_supplicant.pid
root      1573  0.0  0.0   3404  1052 ?        S    Aug26   0:00 /usr/sbin/web100srv -a -s -l /var/log/ndt/web100srv.log
499      1613  0.0  0.2  28684 20904 ?        Ss   Aug26   0:00 SimpleLSBootStrapClientDaemon.pl
499      1636  0.0  0.3  36164 28764 ?        Ss   Aug26   0:01 ls_cache_daemon.pl
499      1690  0.0  0.2  26096 17932 ?        Ss   Aug26   0:00 perfsonar-daemon.pl (1690)
499      1691  0.0  0.2  26096 17816 ?        S    Aug26   0:06 perfsonar-daemon.pl (1690) - Listener (8087)
499      1874  0.0  0.1  12956  8556 ?        Ss   Aug26   0:00 /opt/perfsonar_ps/perfsonarbuoy_ma/bin/bwmaster.pl:master
499      1910  0.0  0.1  13108  8624 ?        Ss   Aug26   0:02 /opt/perfsonar_ps/perfsonarbuoy_ma/bin/powmaster.pl:master
499      1945  0.0  0.2  34388 20192 ?        Ss   Aug26   0:00 perfsonar-daemon.pl (1945)
499      1946  0.0  0.2  34388 20920 ?        S    Aug26   0:02 perfsonar-daemon.pl (1945) - Listener (9990)
root      2093  0.0  0.0  48008  6796 ?        Sl   Aug26   0:19 /usr/bin/python /usr/bin/fail2ban-server -b -s
/var/run/fail2ban/fail2ban.sock -p /var/run/fail2ban/fail2ban.pid -x
root      3804  0.0  0.0   5280  1264 ?        S    01:05   0:00 /bin/sh /usr/bin/mysqld_safe --datadir=/var/lib/mysql
--socket=/var/lib/mysql/mysql.sock --pid-file=/var/run/mysqld/mysqld.pid --basedir=/usr --user=mysql
mysql    3915  0.0  0.2 143892 24252 ?        Sl   01:05   0:25 /usr/libexec/mysqld --basedir=/usr --datadir=/var/lib/mysql
--user=mysql --log-error=/var/log/mysqld.log --pid-file=/var/run/mysqld/mysqld.pid --socket=/var/lib/mysql/mysql.sock
bwctl   3960  0.0  0.0   3032   700 ?        Ss   01:05   0:00 /usr/bin/bwctld -c /etc/bwctld -R /var/run
499     3989  0.0  0.3  39160 26856 ?        Ss   01:05   0:00 perfsonar-daemon.pl (3989)
499     3990  0.0  0.3  39160 27664 ?        S    01:05   0:02 perfsonar-daemon.pl (3989) - Listener (8085)
499     3991  0.0  0.3  39160 27784 ?        S    01:05   0:00 perfsonar-daemon.pl (3989) - Service Maintenance
```

# Securing the perfSONAR-PS Host

```
$ ps auxwww | less
```

Does it seem reasonable?

```
root      1525  0.0  0.0  10676  4400 ?        Ss   Aug26   0:00 NetworkManager --pid-  
file=/var/run/NetworkManager/NetworkManager.pid  
root      1529  0.0  0.0   4792  2284 ?        S    Aug26   0:00 /usr/sbin/modem-manager  
root      1537  0.0  0.0   7596   884 ?        Ss   Aug26   0:00 /usr/sbin/wpa_supplicant -c  
/etc/wpa_supplicant/wpa_supplicant.conf -B -u -f /var/log/wpa_supplicant.log -P /var/run/wpa_supplicant.pid  
root      1573  0.0  0.0   3404  1052 ?        S    Aug26   0:00 /usr/sbin/web100srv -a -s -l /var/log/ndt/web100srv.log  
499      1613  0.0  0.2  28684 20904 ?        Ss   Aug26   0:00 SimpleLSBootStrapClientDaemon.pl  
499      1636  0.0  0.3  36164 28764 ?        Ss   Aug26   0:01 ls_cache_daemon.pl  
499      1690  0.0  0.2  26096 17932 ?        Ss   Aug26   0:00 perfsonar-daemon.pl (1690)  
499      1691  0.0  0.2  26096 17816 ?        S    Aug26   0:06 perfsonar-daemon.pl (1690) - Listener (8087)  
499      1874  0.0  0.1  12956  8556 ?        Ss   Aug26   0:00 /opt/perfsonar_ps/perfsonarbuoy_ma/bin/bwmaster.pl:master  
499      1910  0.0  0.1  13108  8624 ?        Ss   Aug26   0:02 /opt/perfsonar_ps/perfsonarbuoy_ma/bin/powmaster.pl:master  
499      1945  0.0  0.2  34388 20192 ?        Ss   Aug26   0:00 perfsonar-daemon.pl (1945)  
499      1946  0.0  0.2  34388 20920 ?        S    Aug26   0:02 perfsonar-daemon.pl (1945) - Listener (9990)  
root      2093  0.0  0.0  48008  6796 ?        Sl   Aug26   0:19 /usr/bin/python /usr/bin/fail2ban-server -b -s  
/var/run/fail2ban/fail2ban.sock -p /var/run/fail2ban/fail2ban.pid -x  
root      3804  0.0  0.0   5280  1264 ?        S    01:05   0:00 /bin/sh /usr/bin/mysqld_safe --datadir=/var/lib/mysql  
--socket=/var/lib/mysql/mysql.sock --pid-file=/var/run/mysqld/mysqld.pid --basedir=/usr --user=mysql  
mysql     3915  0.0  0.2 143892 24252 ?        Sl   01:05   0:25 /usr/libexec/mysqld --basedir=/usr --datadir=/var/lib/mysql  
--user=mysql --log-error=/var/log/mysqld.log --pid-file=/var/run/mysqld/mysqld.pid --socket=/var/lib/mysql/mysql.sock  
bwctl    3960  0.0  0.0   3032   700 ?        Ss   01:05   0:00 /usr/bin/bwctld -c /etc/bwctld -R /var/run  
499      3989  0.0  0.3  39160 26856 ?        Ss   01:05   0:00 perfsonar-daemon.pl (3989)  
499      3990  0.0  0.3  39160 27664 ?        S    01:05   0:02 perfsonar-daemon.pl (3989) - Listener (8085)  
499      3991  0.0  0.3  39160 27784 ?        S    01:05   0:00 perfsonar-daemon.pl (3989) - Service Maintenance
```

# Securing the perfSONAR-PS Host

**What ports are open?**

**Does this seem reasonable?**

Let's check by doing:

```
$ sudo lsof -i
```

Review the output from lsof. Now try with netstat:

```
$ sudo netstat -ltup
```

Review the output.



# Securing the perfSONAR-PS Host

## Sample output:

```
owampd 22857 owamp 7u IPv6 472972 0t0 TCP *:owamp-control (LISTEN)
owampd 22857 owamp 8u IPv6 619127 0t0 TCP 203.159.61.100:owamp-control->203.159.61.100:56379
(ESTABLISHED)
owampd 22858 owamp 7u IPv6 472972 0t0 TCP *:owamp-control (LISTEN)
owampd 22858 owamp 8u IPv6 619127 0t0 TCP 203.159.61.100:owamp-control->203.159.61.100:56379
(ESTABLISHED)
powstream 22862 perfsonar 16u IPv4 619143 0t0 TCP 203.159.61.100:37059->203.159.61.100:owamp-control
(ESTABLISHED)
powstream 22862 perfsonar 19u IPv4 619155 0t0 UDP 203.159.61.100:8931
powstream 22870 perfsonar 13u IPv4 575951 0t0 TCP 203.159.61.100:46538->perfsonar.nsrc.org:owamp-
control (ESTABLISHED)
powstream 22870 perfsonar 16u IPv4 619213 0t0 TCP 203.159.61.100:60518->perfsonar.nsrc.org:owamp-
control (ESTABLISHED)
powstream 22870 perfsonar 19u IPv4 619219 0t0 UDP 203.159.61.100:8859
bwctld 24490 bwctl 10u IPv6 630746 0t0 TCP 203.159.61.100:4823->perfsonar.nsrc.org:39359
(ESTABLISHED)
bwctld 24491 bwctl 10u IPv6 630746 0t0 TCP 203.159.61.100:4823->perfsonar.nsrc.org:39359
(ESTABLISHED)
bwctld 24491 bwctl 14u IPv4 630755 0t0 TCP 203.159.61.100:6158->perfsonar.nsrc.org:50357
(ESTABLISHED)
iperf 24492 bwctl 8u IPv4 631745 0t0 TCP 203.159.61.100:5221 (LISTEN)
iperf 24492 bwctl 10u IPv6 630746 0t0 TCP 203.159.61.100:4823->perfsonar.nsrc.org:39359
(ESTABLISHED)
iperf 24492 bwctl 14u IPv4 630755 0t0 TCP 203.159.61.100:6158->perfsonar.nsrc.org:50357
(ESTABLISHED)
httpd 24570 apache 4u IPv6 14159 0t0 TCP *:http (LISTEN)
httpd 24570 apache 6u IPv6 14163 0t0 TCP *:https (LISTEN)
```

# Securing the perfSONAR-PS Host

Is software up-to-date?

Let's do:

```
$ sudo yum update
```

If there are packages to be updated respond “Yes”

# Securing the perfSONAR-PS Host

## Turning on automatic updates

We'll do this now. On your perfSONAR-PS Toolkit machine run the following commands:

```
$ sudo yum install yum-cron  
$ sudo chkconfig yum-cron on  
$ sudo service yum-cron start
```

And you should see:

```
Enabling nightly yum update:
```

```
[OK]
```

# Securing the perfSONAR-PS Host

Do you allow root access via ssh?

Are you using passwords via ssh?

If you answer “Yes” to either... strongly consider:

- Disabling root access
- Allow ssh public/private key access only

Let's do this now...

# Securing the perfSONAR-PS Host

On your virtual machine (pc1-pc36) log in as user “sysadm and do the following:

```
$ ssh-keygen -t rsa -b 2048
```

Press <ENTER> for where to save file. Use a passphrase to protect your private key.

```
Generating public/private rsa key pair.
```

```
Enter file in which to save the key (/home/sysadm/.ssh/id_rsa):
```

```
Enter passphrase (empty for no passphrase):
```

```
Enter same passphrase again:
```

```
Your identification has been saved in /home/sysadm/.ssh/id_rsa.
```

```
Your public key has been saved in /home/sysadm/.ssh/id_rsa.pub.
```

# Securing the perfSONAR-PS Host

On your perfsonar-PS Toolkit node do:

```
$ useradd -G wheel sysadm
$ passwd sysadm
$ sudo -s
# su - sysadm
$ mkdir /home/sysadm/.ssh
$ chmod 700 /home/sysadm/.ssh
```

# Securing the perfSONAR-PS Host

On your virtual machine as user *sysadm* do:

```
$ cd /home/sysadm/.ssh
```

```
$ scp id_rsa.pub sysadm@203.159.61.1NN:~/.ssh/.
```

On your perfSONAR-PS Toolkit node do:

```
$ sudo vi /etc/ssh/sshd_config
```

Next slide for changes to make...

# Securing the perfSONAR-PS Host

In the file `/etc/ss/sshd_config` make the changes:

```
#PermitRootLogin yes
```

to

```
PermitRootLogin no
```

and change

```
PasswordAuthentication yes
```

to

```
PasswordAuthentication no
```

Now save the file and type:

```
$ sudo service sshd reload
```



# Securing the perfSONAR-PS Host

On your virtual machine (pc1-pc36) logged in as user “sysadm” try connecting to your perfSONAR-PS Toolkit node:

```
$ ssh root@203.159.61.1NN
```

This should fail... Now try:

```
$ ssh sysadm@203.159.61.1NN
```

And you should be asked for your private key's passphrase – ***not*** the remote password for sysadm

# Looking at Some Results

Let's review the graphs from a few of our perfSONAR-PS Toolkit installations:

- [perfsonar.nsrc.org](http://perfsonar.nsrc.org)
- [Perfsonar1.itu.dk](http://Perfsonar1.itu.dk)
- 203.159.61.100 (intel i5 NUC)
- 203.159.61.123 (laptop)
- 203.159.61.104
- 203.159.61.109
- 203.159.61.110
- 203.159.61.111
- 203.159.61.112
- 203.159.61.113
- 203.159.61.115

# Looking at Some Results

Command line tests using:

## bwctl from 203.159.61.0/24

```
bwctl --verbose --tool iperf3 -fM -t 30 -s 202.28.229.115
```

```
bwctl --verbose --tool iperf -fM -t 30 -s 202.28.229.115
```

```
bwctl --verbose --tool nuttcp -fM -t 30 -s 202.28.229.115
```

```
bwctl --verbose --tool iperf3 -fM -t 30 -s perfsonar.nsrc.org
```

# Looking at Some Results

Command line tests using ndt (network diagnostic tool):

ndt client (web100c1) from 203.159.61.0/24

```
web100c1t -d -n 202.28.229.115
```

“-d” for (d)ebugging information

End point is UniNet bandwidth testing perfSONAR node in Thailand

# Looking at Some Results

ndt from a browser (port 7123):

**UniNet Web100 based Network Diagnostic Tool (NDT)**

**Located at Bangkok, TH**

This java applet was developed to test the reliability and operational status of your desktop computer and network connection. It does this by sending data between your computer and this remote server.

- The slowest link in the end-to-end path (Dial-up modem to 10 Gbps Ethernet/OC-192)
- The Ethernet duplex setting (full or half);
- If congestion is limiting end-to-end throughput.

It can also identify 2 serious error conditions:

- Duplex Mismatch
- Excessive packet loss due to faulty cables.

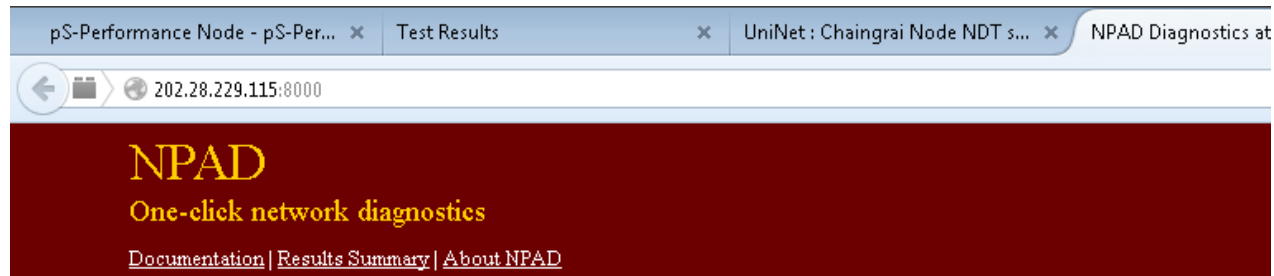
A test takes about 20 seconds. Click on "start" to begin.

TCPWeb100 Network Diagnostic Tool v3.6.5.2  
Click START to start the test

**START**   Statistics   More Details...   Report problem   Options...

# Looking at Some Results

npad (network pathdiag) from a browser (port 8000):



NPAD  
One-click network diagnostics  
[Documentation](#) | [Results Summary](#) | [About NPAD](#)

NPAD (Network Path and Application Diagnosis) is designed to diagnose network performance problems in your end-system (the machine your browser is running on) or the network between it and your nearest NPAD server. For each diagnosed problem, the server prescribes corrective actions with instructions suitable for non-experts.

## Brief instructions

- The test results are most accurate over a short network path. If this NPAD server (located at Bangkok, TH) is not near you, look for a closer server from the list of [Current NPAD Diagnostic Servers](#).
- Have an end-to-end application performance goal ([target round-trip time](#) and [target data rate](#)) in mind. Enter the parameters on the form below and click **Start Test**. Messages will appear in the log window as the test runs, followed by a diagnostic report.
- In the diagnostic report, failed tests (in red) indicate problems that will prevent the application from meeting the end-to-end performance goal. For each message, a question-mark link (?) leads to additional detailed information about the results.
- Every test is fully logged (including your IP address) and test results are [public](#). We use the logs and results to further refine the software.

For more information, see the [NPAD Documentation](#), especially the sections:

- [NPAD Diagnostic Procedure](#) - the full instructions.
- [Theory and Method](#) - why the the tests work.
- [Outcomes](#) - what to do next in the broader debugging context.

Note that tests take 2-5 minutes, depending on the parameters that you provide and the network path. If there is a queue, waiting times might be long.



# Resources

## Axel

<http://freecode.com/projects/axel>

## Esnet Fasterdata Knowledge Base

<http://fasterdata.es.net/>

## GridFTP

<http://toolkit.globus.org/toolkit/docs/latest-stable/gridftp/>

## NDT/NPAD

<https://fasterdata.es.net/performance-testing/network-troubleshooting-tools/ndt-npad/>

<https://twiki.grid.iu.edu/bin/view/Documentation/Release3/NetworkPerformanceToolkit>