Nginx Tool User Guide	Public
1.0	Total 7 pages

Nginx Tool User Guide

Shyju PV (shyju.pv@gmail.com)

Table of Contents

1 Introduction

This document gives details of Nginx tool configuration and execution with caliper framework.

2 Lab Setup for Nginx tool

To execute Nginx tool with caliper framework, below setup is required:

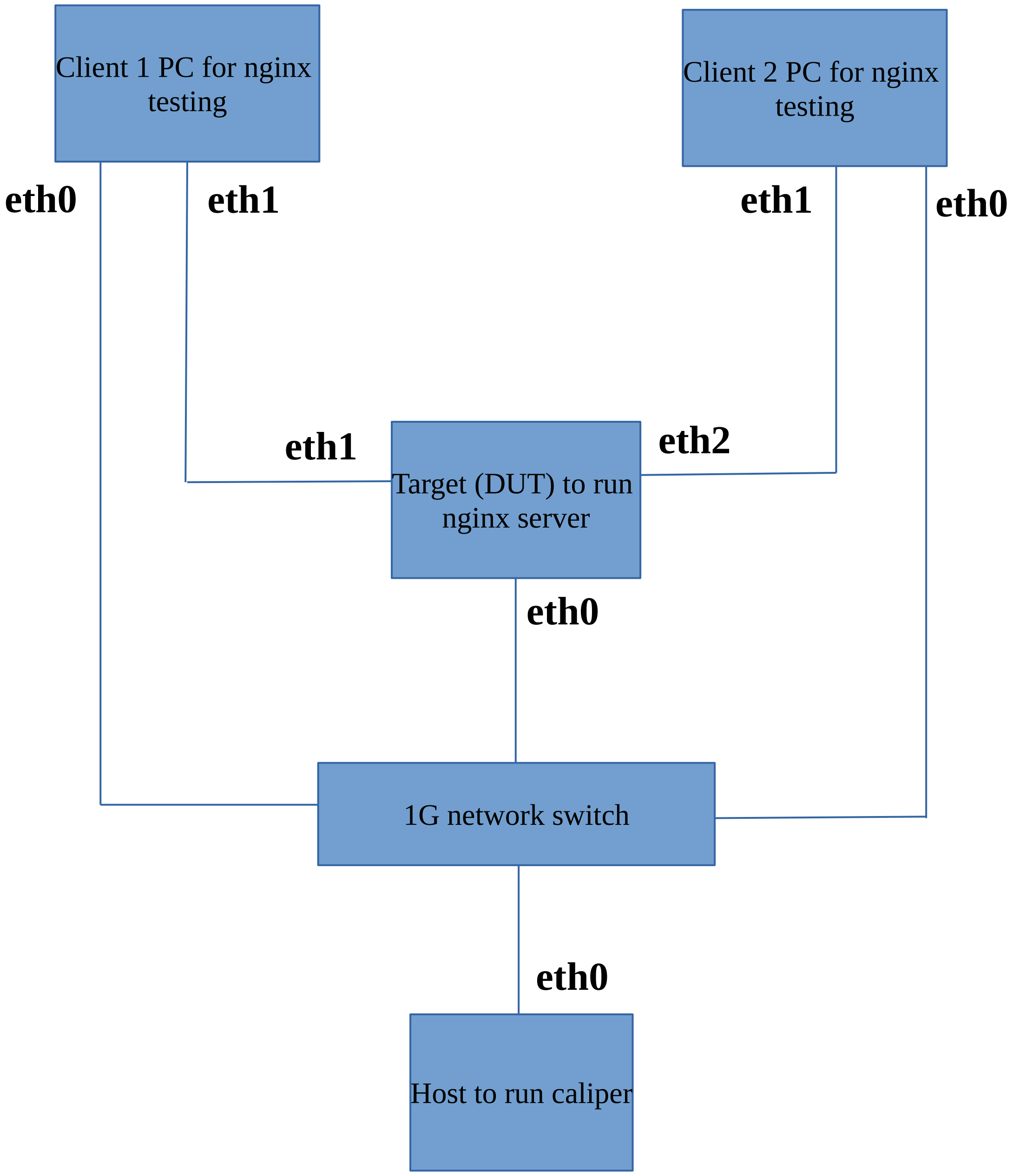

Client 1 PC for nginx testing
Client 2 PC for nginx testing
eth0
eth1
eth1
eth0
eth1
eth2
Target (DUT) to run nginx server
eth0
1G network switch
eth0
Host to run caliper

Figure 1: Lab Setup for Nginx Tool

Install caliper in the host PC (Refer caliper user manual).
Auto login is mandatory for below platforms:

- All client PCs to Target (DUT) platform.

- Host PC to target and host to all client PC.

Auto-login can be done by using below commands:

- Use this command to generate public key (if key is not generated): ***ssh-keygen -t rsa***
- Use this command to copy the key to platform: ***ssh-copy-id -i ~/.ssh/id_rsa.pub <user_name>@<ip_address>***

Note: To copy the publickey from clients to Target (DUT), use respective ethernet interface of the target.
Example: For client 1, use eth1 ip address of Traget (DUT) in above command**.**

3 Weighttp client installation

Weighttp is a lightweight and small benchmarking tool for webservers (like Nginx).
weighttp supports multithreading to make good use of modern CPUs with multiple cores as well as asynchronous i/o.

To install weighttp tool in client machine follow below steps:

For Ubuntu OS client PC:

```
sudo apt-get install libev4 libev-dev
git clone https://github.com/lighttpd/weighttp.git
cd weighttp
./autogen.sh
./configure && make && make install
```

For CentOS client PC:

```
yum install -y epel-release
yum install libev libev-devel -y
cd /usr/local/src
git clone https://github.com/lighttpd/weighttp.git
cd weighttp
./waf configure
./waf build
./waf install
```

Copy run_weighttp_script.sh from caliper/utils/automation_scripts/Scripts directory to client PC home directory.
For example: scp caliper/utils/automation_scripts/Scripts/run_weighttp_script.sh root@192.168.40.10:~/run_weighttp_script.sh

Note: For more details of weighttp tool, refer link - https://redmine.lighttpd.net/projects/weighttp/wiki

4 client_config.cfg file description

Client PC information and target platform ethernet interfaces information:

```
[nginx]
no_of_clients: <specify number of clients. In above diagram, number of clients are 2 >
client_1_ip: <eth0 ip address of client 1>
client_1_user: <client 1 user name>
client_1_password: <password of client 1 user name>
client_2_ip: <eth0 ip address of client 2>
client_2_user: <client 2 user name>
client_2_password: <password of client 2 user name>
target_ip_1_10g: <eth1 ip address of target platform>
target_ip_2_10g: <eth2 ip address of target platform>
target_port_1: <give uniq port number of traget platform which is available>
target_port_2: <give uniq port number of traget platform which is available>
```

client 1 login information:

```
[TestNode]
```

```
ip: <eth0 ip address of client 1>
port: <unique port number which is avalilable>
user: <client 1 user name>
password: <password of client 1 user name>
```

This section will be used to protect nginx clients to be used by single target.

Example:

```
[TestNode]
ip:192.168.40.56
port:5000
user:root
password:root
TestNode_ip_10g:192.168.60.110
# SSH implementation used by server (ssh or paramiko)
ssh_engine: raw_ssh
# enable OpenSSH connection sharing. Only useful if engine_ssh is 'raw_ssh'
enable_master_ssh: True
```

Example:

```
[nginx]
no_of_clients:2
client_1_ip:192.168.40.56
client_1_user:root
client_1_password:root
client_2_ip:192.168.40.9
client_2_user:root
client_2_password:root123
target_ip_1_10g:192.168.60.111
target_ip_2_10g:192.168.50.100
target_port_1:8000
target_port_2:8001
```

5 nginx_run.cfg file test case description

This file contains the configuration of test case to be executed on target platform. The sample test case is given as below:

```
[nginx_64_core_wrps]
category = Performance application nginx_64_core_wrps
scores_way = compute_speed_score 1
no_of_clients = 2
command = "if [ -f nginx_tar.gz ]; then tar -xvf nginx_tar.gz; rm nginx_tar.gz; fi; pushd nginx/nginx_scripts; ./nginx_conf.sh; ~/irq_0_7.sh;  ~/irq_32_39.sh; ./run_nginx_dstat.sh 2 $TARGET_IP_1_10G $TARGET_PORT_1 24 8 $TARGET_IP_2_10G $TARGET_PORT_2 24 40; popd;"
parser = nginx_parser
```

Main fields are "**no_of_clients**" and "**command**".

- **no_of_clients :** If there is only one client which communicates to nginx server, please give no_of_clients = 1.
- irq_0_7.sh and irq_32_39.sh files are configuration files to set IRQs of ethernet ineterfaces on the respective CPU cores (CPU affinity) .

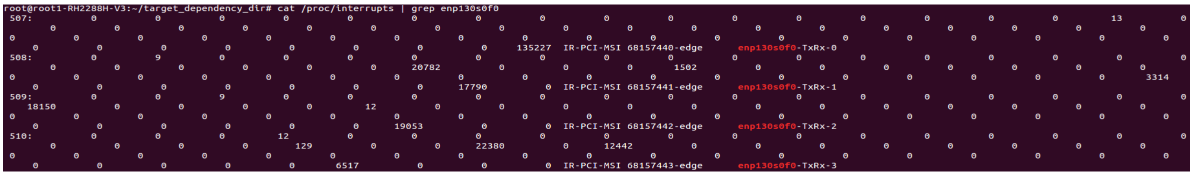
User has to create this file on the target platform and give the file name in the "**command**" field as a parameter.

For example, see the contents of irq_0_7.sh file given below. (eth1 interface interrupts will be handled by CPU 0 to 7)

```
#!/bin/bash
# eth1 assigned to 0-7
echo 00000000,00000001 > /proc/irq/84/smp_affinity
echo 00000000,00000001 > /proc/irq/85/smp_affinity
```

```
 5 echo 00000000,00000002 > /proc/irq/87/smp_affinity
 6 echo 00000000,00000002 > /proc/irq/88/smp_affinity
 7 echo 00000000,00000004 > /proc/irq/90/smp_affinity
 8 echo 00000000,00000004 > /proc/irq/91/smp_affinity
 9 echo 00000000,00000008 > /proc/irq/93/smp_affinity
10 echo 00000000,00000008 > /proc/irq/94/smp_affinity
11 echo 00000000,00000010 > /proc/irq/96/smp_affinity
12 echo 00000000,00000010 > /proc/irq/97/smp_affinity
13 echo 00000000,00000020 > /proc/irq/99/smp_affinity
14 echo 00000000,00000020 > /proc/irq/100/smp_affinity
15 echo 00000000,00000040 > /proc/irq/102/smp_affinity
16 echo 00000000,00000040 > /proc/irq/103/smp_affinity
17 echo 00000000,00000080 > /proc/irq/105/smp_affinity
18 echo 00000000,00000080 > /proc/irq/106/smp_affinity
19 echo 00000000,00000001 > /proc/irq/108/smp_affinity
20 echo 00000000,00000001 > /proc/irq/109/smp_affinity
21 echo 00000000,00000002 > /proc/irq/111/smp_affinity
22 echo 00000000,00000002 > /proc/irq/112/smp_affinity
23 echo 00000000,00000004 > /proc/irq/114/smp_affinity
24 echo 00000000,00000004 > /proc/irq/115/smp_affinity
25 echo 00000000,00000008 > /proc/irq/117/smp_affinity
26 echo 00000000,00000008 > /proc/irq/118/smp_affinity
27 echo 00000000,00000010 > /proc/irq/120/smp_affinity
28 echo 00000000,00000010 > /proc/irq/121/smp_affinity
29 echo 00000000,00000020 > /proc/irq/123/smp_affinity
30 echo 00000000,00000020 > /proc/irq/124/smp_affinity
31 echo 00000000,00000040 > /proc/irq/126/smp_affinity
32 echo 00000000,00000040 > /proc/irq/127/smp_affinity
33 echo 00000000,00000080 > /proc/irq/129/smp_affinity
34 echo 00000000,00000080 > /proc/irq/130/smp_affinity
```

Use “***cat /proc/interrupt | grep <Interface name>*** command to find the IRQ number assigned to eth1 interface. The sample output is as follows:



```
root@root1-RH2288H-V3:~/target_dependency_dir# cat /proc/interrupts | grep enp130s0f0
 507:          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0         13          0
       0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0
0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0
   0          0          0          0          0          0          0          0     135227  IR-PCI-MSI 68157440-edge      enp130s0f0-TxRx-0
 508:          0          9          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0
       0          0          0          0          0          0      20782          0          0          0          0       1502          0          0          0          0          0          0          0
0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0       3314
   0          0          0          0          0          0          0      17790          0  IR-PCI-MSI 68157441-edge      enp130s0f0-TxRx-1
 509:          0          0          9          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0
   18150          0          0          0          0         12          0          0          0          0          0          0          0          0          0          0          0          0          0
0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0
   0          0          0          0          0          0      19053          0          0  IR-PCI-MSI 68157442-edge      enp130s0f0-TxRx-2
 510:          0          0          0         12          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0
       0          0          0          0        129          0          0      22380          0      12442          0          0          0          0          0          0          0          0
0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0          0
   0          0          0          0          0       6517          0          0          0  IR-PCI-MSI 68157443-edge      enp130s0f0-TxRx-3
```

The description of line number 3 *[echo 00000000,00000001 > /proc/irq/84/smp_affinity]* is as follows:

- echo 00000000,00000001 indicates CPU number 0.
- '84' indicates that IRQ number 84 is assigned to eth1 Tx/Rx channel. The IRQ number is depends on the platform.
- So eth1 Tx/Rx interrupt will be handled by CPU0
- Same way user can assign the IRQ number of other eth1 Tx/Rx channels to different CPUs.
- User has to assign IRQs equally all the CPUs (CPU range 0 to 7 has been used in above example)

Note:

1. The intention of assigning IRQs to CPUs is to get maximum throughput of nginx server by limiting the context switches and balancing the load.

run_nginx_dstat.sh script parameters description is as follows:

- ./run_nginx_dstat.sh
  <number of clients>
  $target_ip_1_10g
  $target_port_1
  <number of CPU cores uses for first instance of nginx server>
  <starting number of CPU core for first instance of nginx server>
  $target_ip_2_10g
  $target_port_2

<number of CPU cores uses for second instance of nginx server>
<starting number of CPU core for second instance of nginx server>

$target_ip_1_10g, $target_port_1, $target_ip_2_10g and $target_port_2: The ip address and port number will be taken from client_config.cfg file of the caliper workspace that should have been already configured.

6 nginx_application_run.cfg file test case description

This file contains the configuration of test case to be executed on Client PC. The sample test case is given as below:

```
[nginx_32_core_local_static]
command1 = "~/irq_0_7.sh; ~/run_weighttp_script.sh $target_user_name $target_ip_1_10g $target_port_1 24 8 25000 index0.html"
command2 = "~/irq_0_7.sh; ~/run_weighttp_script.sh $target_user_name $target_ip_2_10g $target_port_2 24 8 25000 index1.html"
```

- command1 and command2 indicates commnads to be execute on client 1 and client 2 respectively. If user wants to add 3rd client, then add command3 in this file and so on.
- irq_0_7.sh: configuration file to bind IRQs of ethernet ineterfaces to the range of CPUs.
  The explanation is same as above.
- run_weighttp_script.sh will take below paramters:
  $target_user_name
  $target_ip_1_10g
  $target_port_1
  <number of CPU cores uses for weighttp process>
  <starting number of CPU core for weighttp process>
  25000
  index0.html
- $target_user_name, $target_ip_1_10g and $target_port_1: the target platform information will be taken from client_config.cfg file.
- 25000 is the number of requests to be send to nginx server.
- index0.html file is the default html file for the client 1, index1.html file for client 2 and so on.