Tools-lib-python

Toolkit for have some informations about OS in python

For full content and examples with details, see readme in repository [here]

Contributors

- ABOGOUNRIN Aicha
- BKWEDOU-NGAMENI Bechir Delibes
- DIALLO Abdourahmane
- IROUDAYAME Jérémy

Requirements

- Linux server
- At leat 2Go RAM and 1 CPU
- python3 and python3-pip installed

Install application

- Clone the repository

```
bngameni@delbechirclara:$ git clone https://github.com/my-esgi-projects/tp-tools-lib-python.git
Cloning into 'tp-tools-lib-python'...
remote: Enumerating objects: 130, done.
remote: Counting objects: 100% (130/130), done.
remote: Compressing objects: 100% (83/83), done.
remote: Total 130 (delta 50), reused 110 (delta 39), pack-reused 0
Receiving objects: 100% (130/130), 141.92 KiB | 2.25 MiB/s, done.
Resolving deltas: 100% (50/50), done
```

- Install requirements

```
bngameni@delbechirclara:$ cd tp-tools-lib-python/
bngameni@delbechirclara:$ python3 -m pip install -r requirements.txt
```

Install application

- Enter in workspace directory

```
bngameni@delbechirclara:$ cd workspace
```

Client-Server application using socket

“
- For test this, Install application both client side and server side
- Here you have two scripts to launch. First server script and client script after.

”

Client-Server application using socket

Server script

- Display help

```
bngameni@delbechirclara:$ python server.py -h
usage: server.py [-h] [-H HOST] [-P PORT]

Launch without options will use localhost as default host and 8090 as defaut port

options:
  -h, --help            show this help message and exit
  -H HOST, --host HOST  server listen address
  -P PORT, --port PORT  server listen port
```

Client-Server application using socket

Server script

- Launch server script

```
bngameni@delbechirclara:$ python3 server.py --host 0.0.0.0 --port 8989
```

“ • Server is muted when it's launched ”

Client Script

- Display help

```
bngameni@delbechirclara:$ python client.py -h
usage: client.py [-h] [-H HOST] [-P PORT]

Launch without options will use localhost as default host and 8090 as defaut port

options:
  -h, --help            show this help message and exit
  -H HOST, --host HOST  server listen address
  -P PORT, --port PORT  server listen port
```

Client Script

- Show network interface of client server

```
bngameni@ansible-dev:$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:6b:f0:dc brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.45/24 brd 192.168.1.255 scope global dynamic enp0s8
       valid_lft 84917sec preferred_lft 84917sec
    inet6 2001:861:5e49:aa00:a00:27ff:fe6b:f0dc/64 scope global dynamic mngtmpaddr noprefixroute
       valid_lft 86400sec preferred_lft 14400sec
    inet6 fe80::a00:27ff:fe6b:f0dc/64 scope link
       valid_lft forever preferred_lft forever
```

Client Script

- Launch client script

```
bngameni@delbechirclara:$ python3 client.py --host 192.168.1.94 --port 8989
1. Display OS Informations
2. Display CPU Informations
3. Display Informations About Sensors
4. Display Informations About Memory
0. Quit application
Enter your choice: 1
         1. Show os name
         2. Show os architecture
         3. Show os release
         4. Show os hostname
         0. Back to main menu
Enter your choice: 1
 The os name is : Linux
```

Client Script

“ • Only first menu is showed on this readme. ”

Filter automation using iptables on linux server from client script in python

- Display help

```
bngameni@delbechirclara:$ python iptables_ssh.py -h
usage: iptables_ssh [options]
Execute commands througth ssh
options:
  -h, --help            show this help message and exit
  -H HOST, --host HOST  Server for ssh connexion
  -U USERNAME, --username USERNAME Username for login to the server
  -P PASSWORD, --password PASSWORD Password for login to the server
  -F FILE, --file FILE  Path to file which contains commands
  -L LOG, --log LOG     Path to file for logging all informations
```

Filter automation using iptables on linux server from client script in python

- Launch iptables to remote server
  - Content of iptables rules files

```
bngameni@delbechirclara:$ cat files/rules.txt
# Autoriser les connexions entrantes sur le port 80
iptables -A OUTPUT -p tcp --sport 80 -m state --state ESTABLISHED -j ACCEPT

# Autoriser les connexions entrantes sur le port 443
iptables -A INPUT -p tcp --dport 443  -j ACCEPT

# Autoriser les connexions sortantes sur le port 443
iptables -A OUTPUT -p tcp --sport 443  -j ACCEPT
```

Filter automation using iptables on linux server from client script in python

- Launch iptables to remote server
  - Run script

```
bngameni@delbechirclara:$ python iptables_ssh.py --host 192.168.1.45 --username=xxxxxx --password=xxxxxxx --file=files/rules.txt --log=./ssh_log.txt
Sending command: iptables -A OUTPUT -p tcp --sport 80 -m state --state ESTABLISHED -j ACCEPT
Output: []
Sending command: iptables -A INPUT -p tcp --dport 443  -j ACCEPT
Output: []
Sending command: iptables -A OUTPUT -p tcp --sport 443  -j ACCEPT
Output: []
Sending command: iptables -A INPUT -p tcp --dport 22 -j ACCEPT
Output: []
Sending command: iptables -A OUTPUT -p tcp --dport 22 -j ACCEPT
Output: []
Sending command: iptables -A OUTPUT -p icmp  -j DROP
Output: []
Sending command: iptables -A INPUT -p icmp  -j DROP
Output: []
Sending command: iptables -A INPUT -p udp --dport 123  -j ACCEPT
Output: []
Sending command: iptables -A OUTPUT -p udp --dport 123 -j ACCEPT
Output: []
```

Filter automation using iptables on linux server from client script in python

- Launch iptables to remote server
  - Show log files

```
bngameni@delbechirclara:$ tail -n10 ./ssh_log.txt
INFO 2023-05-14 10:04:06,529 Authentication (password) successful!
DEBUG 2023-05-14 10:04:06,529 [chan 0] Max packet in: 32768 bytes
DEBUG 2023-05-14 10:04:07,033 Received global request "hostkeys-00@openssh.com"
DEBUG 2023-05-14 10:04:07,033 Rejecting "hostkeys-00@openssh.com" global request from server.
DEBUG 2023-05-14 10:04:07,076 [chan 0] Max packet out: 32768 bytes
DEBUG 2023-05-14 10:04:07,076 Secsh channel 0 opened.
DEBUG 2023-05-14 10:04:07,077 [chan 0] Sesch channel 0 request ok
DEBUG 2023-05-14 10:04:07,103 [chan 0] EOF received (0)
DEBUG 2023-05-14 10:04:07,103 [chan 0] EOF sent (0)
INFO 2023-05-14 10:04:08,081 command: iptables -A OUTPUT -p udp --dport 123 -j ACCEPT -> stdout: []
```

Filter automation using iptables on linux server from client script in python

- Launch iptables to remote server
  - Show iptables of remote server

```
ubuntu@ansible-dev:~$ sudo iptables-save
# Generated by iptables-save v1.8.4 on Sun May 14 08:24:02 2023
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -p icmp -j DROP
-A OUTPUT -p tcp -m tcp --sport 80 -m state --state ESTABLISHED -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 443 -j ACCEPT
-A OUTPUT -p tcp -m tcp --dport 22 -j ACCEPT
# Completed on Sun May 14 08:24:02 2023
```

Filter automation using iptables on linux server from client script in python

- Launch iptables to remote server
  - Try ping to remote server and see effects of ping drop rules

```
bngameni@delbechirclara:$ ping -c4 192.168.1.45
PING 192.168.1.45 (192.168.1.45) 56(84) bytes of data.
^C
--- 192.168.1.45 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3068ms
```

Mini wireshark

- Launch application

```
bngameni@delbechirclara:$ sudo python3 wireshark.py
Mini WireShark - Select protocol to filter:
1. http filtering
2. dns filtering
3. tcp filtering
4. udp filtering
5. icmp filtering
0. Quit
Entrez le numéro du protocole : 3
1. Scan avec scappy
2. Scan with pyshark
Entrez l'outils de filtrage: 1
IP / TCP 10.0.16.13:51372 > 34.107.221.82:http PA / Raw
IP / TCP 34.107.221.82:http > 10.0.16.13:51372 PA / Raw
```