py-cgnat

Python module for generating CGNAT rules using netmap

Brief

Python library and CLI program for generating firewall rules to deploy Carrier-Grade NAT, besides translating a given IP and port to its private address and vice versa. The methodology consists in building netmap rules at 1:32 public-private ratio, mapping a range of 2.000 ports for each client. Works for any netmask, since that follow the 1:32 ratio:

Private prefix	Public prefix	N. of clients
...	...	...
/20	/25	4096
/21	/26	2048
/22	/27	1024
/23	/28	512
/24	/29	256
/25	/30	128
/26	/31	64
/27	/32	32

Supported Platforms

- MikroTik RouterOS

Requirements

- Python 3.7+

How to install it?

Installation can just being done with `pip`:

```
pip install pycgnat
```

How to use it?

1. Command Line Interface

For **generating** the rules, you can print it in console or save it to a file:

```
pycgnat 100.64.0.0/20 203.0.113.0/25 gen routeros filename.rsc
pycgnat 100.64.0.0/20 203.0.113.0/25 gen routeros
```

For **translating** a private IP to its public one, use the `direct` option:

```
pycgnat 100.64.0.0/20 203.0.113.0/25 trans --direct 100.64.2.15
pycgnat 100.64.0.0/20 203.0.113.0/25 trans -d 100.64.2.15
```

For **translatig** a public IP and port to its private IP correspondent, use the `reverse` option:

```
pycgnat 100.64.0.0/20 203.0.113.0/25 trans --reverse 203.0.113.20:13578
pycgnat 100.64.0.0/20 203.0.113.0/25 trans -r 203.0.113.20:13578
```

The CLI includes useful **help** command (supported by `argparse` framework), so just type:

```
pycgnat --help
pycgnat -h
```

2. Python library

You can use the functionalities directly in Python lang. Just **import** the wanted module to your program:

```
from pycgnat.translator.reverse import cgnat_reverse

dic = cgnat_reverse(privnet, pubnet, IPv4Address('203.0.113.20'), 13578)
print(dic['private_ip'])
```

The full `pycgnat`'s documentation is written in the source-code.

Future works

- Add support for other platfoms (I'm using MikroTik for while, so this is the reason for only supporting it at first version).