



Route Origin Validation Lab

Part-1: Installing RPKI Validator (Fort)

Login Details

- Username `apnic` and password `training`.

Preinstalled packages

To save time, the following essential package(s) have been preinstalled on the containers:

- `rsync`

Lab Setup

For this lab, we will use [FORT](#) from NIC México as the RPKI validator.

1. Login to your server (SSH from the jumphost to your container using the `username` and `password` given above), where `x` is your VM number:

```
ssh apnic@192.168.30.X
```

2. Update the repository

```
sudo apt update && sudo apt upgrade
```

◦ **NOTE:** Please note that FORT has the following dependencies:

- [jansson](#)
- libcrypto ([LibreSSL](#) or [OpenSSL](#) >=1.1)
- rsync

◦ To install the dependencies:

```
sudo apt install openssl libjansson-dev -y
```

◦ Verify the OpenSSL version

```
openssl version -a
```

```
tashi@fort:~$ openssl version -a
OpenSSL 1.1.1 11 Sep 2018
```

3. Download and install the validator (we will install from the debian package)

```
wget https://github.com/NICMx/FORT-validator/releases/download/v1.2.1/fort_1.2.1-1_amd64.deb
```

and install:

```
sudo apt install ./fort_1.2.1-1_amd64.deb
```

◦ Note:

- there are other install options listed in the vendor [documentation](#)
- the debian package comes with a systemd service, which allows us to run it as a daemon

4. Edit the configuration file to define the server address, RTR port number, etc:

```
sudo nano /etc/fort/config.json
```

```

{
  "tal": "/etc/fort/tal",
  "server": {
    "address": "192.168.30.X",
    "port": "8323",
    "interval": {
      "validation": 900,
      "refresh": 900,
      "retry": 600,
      "expire": 7200
    }
  },
  "output": {
    "roa": "/tmp/fort/fort.csv"
  }
}

```

◦ Note:

- **validation**: time (in seconds) the Validator should wait after updating and validating the ROA cache before updating again from the global repo.
- **refresh**: time (in seconds) the RTR client (router) has to wait before trying to poll the Validator cache ([RFC8210](#) default 3600 seconds).
- **retry**: time (in seconds) the RTR client should wait before retrying after a failed refresh of the cache (RFC8210 default 600 seconds).
- **expire**: time (in seconds) the RTR client can use its validated ROA cache if cannot refresh the data, after which it should discard (RFC8210 default 7200 seconds).
- **output**: print validated ROAs to a CSV file

5. Fort validator ships with all RIR TALs except ARIN's. You need to agree to be bound by [ARIN's Relying Party Agreement \(RPA\)](#) before using it:

- Download ARIN's TAL into the `/etc/fort/tal` directory (used `sudo` where needed)

```

cd /etc/fort/tal
sudo wget https://www.arin.net/resources/manage/rpki/arin-rfc7730.tal -O arin.tal
cd

```

- verify your TALs

```
ls /etc/fort/tal
```

```
tashi@fort:~$ ls /etc/fort/tal
afrinic.tal  apnic.tal  arin.tal  lacnic.tal  ripe.tal
```

6. Start the validator (RTR server):

```
sudo service fort start
```

- will start in server mode by default using the parameters defined in the configuration file (`/etc/fort/config.json`)
- you can check the status with `sudo service fort status`

```
• fort.service - Fort RPKI Validator/Server
  Loaded: loaded (/lib/systemd/system/fort.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2019-11-06 17:09:26 AEST; 4s ago
    Docs: https://nicmx.github.io/FORT-validator/doc/index.html
  Main PID: 1939 (fort)
    Tasks: 17 (limit: 6143)
  CGroup: /system.slice/fort.service
          └─1939 /usr/bin/fort --configuration-file /etc/fort/config.json
             └─1947 rsync --recursive --delete --times --contimeout=20 rsync://rpki.arin.net/repository /tmp/fort/repository/rpki.arin.net/repository
                └─1949 rsync --recursive --delete --times --contimeout=20 rsync://rpki.afnic.net/repository /tmp/fort/repository/rpki.afnic.net/repository
                   └─1950 rsync --recursive --delete --times --contimeout=20 rsync://repository.lacnic.net/rpki /tmp/fort/repository/repository.lacnic.net/rpki
                      └─1954 rsync --recursive --delete --times --contimeout=20 rsync://rpki.arin.net/repository /tmp/fort/repository/rpki.arin.net/repository
                         └─1955 rsync --recursive --delete --times --contimeout=20 rsync://rpki.afrinic.net/repository /tmp/fort/repository/rpki.afrinic.net/repository
                            └─1957 rsync --recursive --delete --times --contimeout=20 rsync://rpki.ripe.net/repository /tmp/fort/repository/rpki.ripe.net/repository
                               └─1958 rsync --recursive --delete --times --contimeout=20 rsync://rpki.afnic.net/repository /tmp/fort/repository/rpki.afnic.net/repository
                                  └─1959 rsync --recursive --delete --times --contimeout=20 rsync://repository.lacnic.net/rpki /tmp/fort/repository/repository.lacnic.net/rpki
                                     └─1960 rsync --recursive --delete --times --contimeout=20 rsync://rpki.afrinic.net/repository /tmp/fort/repository/rpki.afrinic.net/repository
                                        └─1961 rsync --recursive --delete --times --contimeout=20 rsync://rpki.ripe.net/repository /tmp/fort/repository/rpki.ripe.net/repository

Nov 06 17:09:26 fort fort[1939]: INF: output.bgpssec: (null)
Nov 06 17:09:26 fort fort[1939]: INF: }
Nov 06 17:09:26 fort fort[1939]: INF: Server mode configured; disabling logging on standard streams.
Nov 06 17:09:26 fort fort[1939]: INF: (Logs will be sent to syslog only.)
Nov 06 17:09:26 fort fort[1939]: INF: Attempting to bind socket to address ' [REDACTED] ', port '8323'.
Nov 06 17:09:26 fort fort[1939]: INF: Success; bound to address ' [REDACTED] ', port '8323'.
```

- Have a look at the validated ROA payload (*Origin ASN, Prefix, Max prefix length*):

```
more /tmp/fort/fort.csv
```

```
apnic@group13:~$ more /tmp/fort/roas.csv
ASN,Prefix,Max prefix length
AS132238,103.96.76.0/22,24
AS135419,103.120.112.0/22,22
AS135419,103.120.112.0/24,24
AS135419,103.120.113.0/24,24
AS135419,103.120.114.0/24,24
AS135419,103.120.115.0/24,24
AS38719,103.67.234.0/23,24
AS38719,103.67.248.0/24,24
AS38719,2405:df80::/32,48
AS135132,103.52.62.0/24,24
AS64073,103.139.184.0/23,24
AS64073,103.250.88.0/22,24
AS64073,163.47.128.0/22,24
AS64073,202.179.140.0/22,24
AS132730,103.24.32.0/22,24
AS132730,103.224.28.0/22,24
AS132730,163.53.28.0/22,24
AS132730,2406:b100::/32,48
```

- If there is no output yet, it means Fort is still working through the initial process of fetching ROAs from the repos and validating, which generally takes a while.
- **NOTE:** If you still do not have any output, stop and restart Fort by explicitly specifying the file where ROAs will be printed `output.roa=FILE`

```
sudo service fort stop
sudo service fort start --output.roa="/tmp/fort/fort.csv"
```

7. **[Optional]** If you have two separate Validators installed (for redundancy/code diversity), compare the validated ROA payload outputs for consistency:

- But before you compare the VRPs, you need to sort the output. Example below using `sort` to sort alphanumerically:

```
sort /tmp/fort/fort.csv > fort_sorted.csv
```

- Now you can compare the validated ROA outputs, for example Routinator and Fort:

```
diff -u rout_sorted.csv fort_sorted.csv
```

- Discuss any differences with your group mates and instructor.

Now your validator is ready to feed the validated cache to BGP speaking routers through the RTR (RPKI-to-Router) protocol.

Part-2: RTR session

Validator side

Fort can act as an RTR server, to allow RPKI enabled routers to connect to it and fetch the validated cache (ROA cache).

- Based on the configuration file (`/etc/fort/config.json`), the RTR server is listening on `192.168.30.X` (where X is your VM number) and port `8323`
 - The timers can be tweaked to suit your need ([RFC8210](#) has recommendations).
-