Monitoring BGP and Generating IRR Filters on Mikrotik Routers

Ben Ryall | Lee Hetherington

EPF 2023 | Prague, Czech Republic

What we’re talking about

- BGP session monitoring on MikroTik routers
  - You want to know sessions are down, right?
- Generating and deployment of strict IRR filters
  - Mis-configured sessions can cause pain

Why did we do this?

- Everyone needs a hobby
- Lee deployed as35008, using MikroTik routers
- Wanted to keep operations lightweight, but be MANRS compliant
  - Strict IRR Filters, RPKI, Anti-Spoofing etc
  - Prove how easy it is todo the ‘right thing’
- Where features didn’t exist, coded them!

MANRS

What do I need?

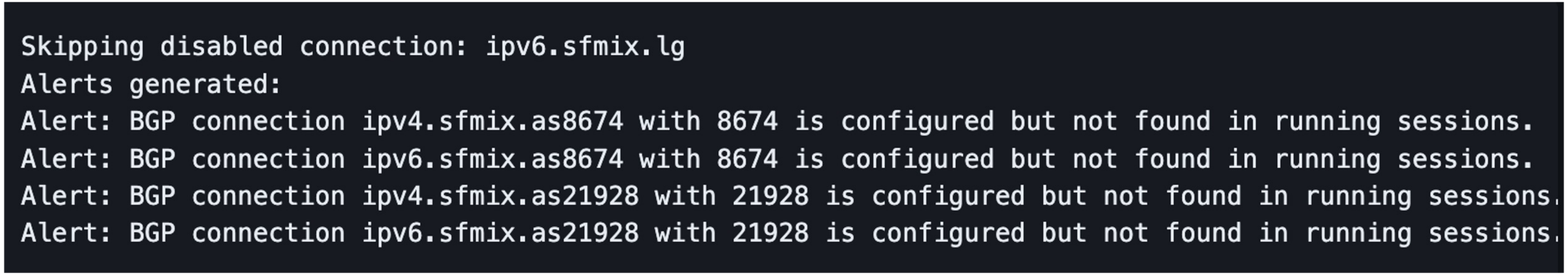
- Mikrotik router(s) running ROS 7+ (We've tested against 7.7 through 7.11) and speaking BGP
- API configured and accessible on your router(s)
- Mikrotik RouterOS API Python Packages
- Something to generate IRR filters

BGP Monitoring

- There are two scripts in this collection
  - Which will help you monitor BGP sessions on your Mikrotik Routers
  - Mikrotik is **STILL** lacking BGP support in their SNMP implementation
- mikrotik_bgpmon.py
- mikrotik_bgpmon_print.py

BGP Monitoring…part 2

- mikrotik_bgpmon.py
  - This script will look at configured (but not disabled) peers under /routing/bgp/connection and compare them with the items under /routing/bgp/session to see if they match
  - It'll also look at the status of sessions under /routing/bgp/session and alert you of any status that isn't established
  - It'll also send you an email with the output each time you run the script if you specify an email address



```
Skipping disabled connection: ipv6.sfmix.lg
Alerts generated:
Alert: BGP connection ipv4.sfmix.as8674 with 8674 is configured but not found in running sessions.
Alert: BGP connection ipv6.sfmix.as8674 with 8674 is configured but not found in running sessions.
Alert: BGP connection ipv4.sfmix.as21928 with 21928 is configured but not found in running sessions.
Alert: BGP connection ipv6.sfmix.as21928 with 21928 is configured but not found in running sessions.
```

BGP Monitoring…part 3

- mikrotik_bgpmon_print.py
  - This script will display the sessions currently running on the router
  - It doesn't look at things which are in /routing/bgp/connection that are not also in /routing/bgp/session - so it shouldn't be used to monitor the health
  - If you supply the routerIP, then up or down to the script at the command line, it'll show you the status

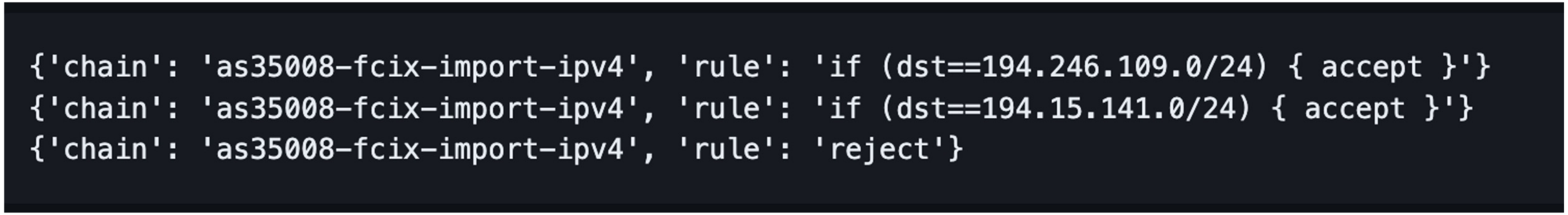
```
Example Output:

Session: ipv6.sfmix.rs1-1, AS: 63055, Peer IP: 2001:504:30::ba06:3055:1, Status: true,
Uptime: 3h19m56s310ms, Prefixes: 55721

Session: ipv4.sfmix.rs1-1, AS: 63055, Peer IP: 206.197.187.253, Status: true, Uptime:
3h19m56s310ms, Prefixes: 111222
```

Deploying Strict IRR Filters

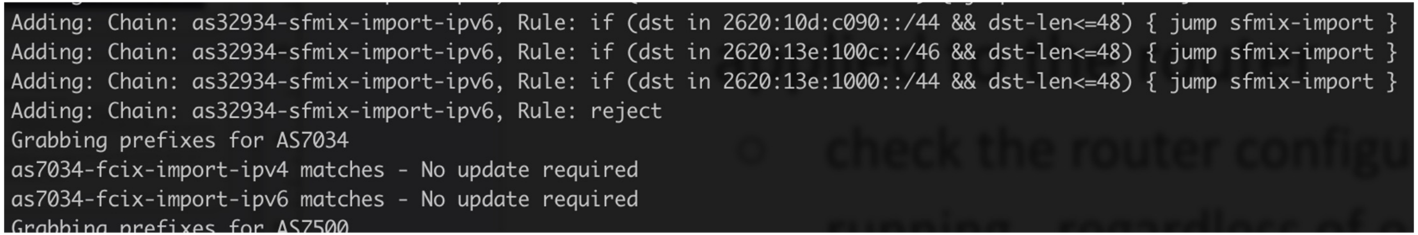
- This script will help you take a desired configuration for your IRR filter from a text file in a specified format
- Examples of IRRPT generating filters, then parsing them through a bash script to output in the required json like format
- Example bash wrappers/scripts in the github repo



```
{'chain': 'as35008-fcix-import-ipv4', 'rule': 'if (dst==194.246.109.0/24) { accept }'}
{'chain': 'as35008-fcix-import-ipv4', 'rule': 'if (dst==194.15.141.0/24) { accept }'}
{'chain': 'as35008-fcix-import-ipv4', 'rule': 'reject'}
```

Deploying Strict IRR Filters… part 2

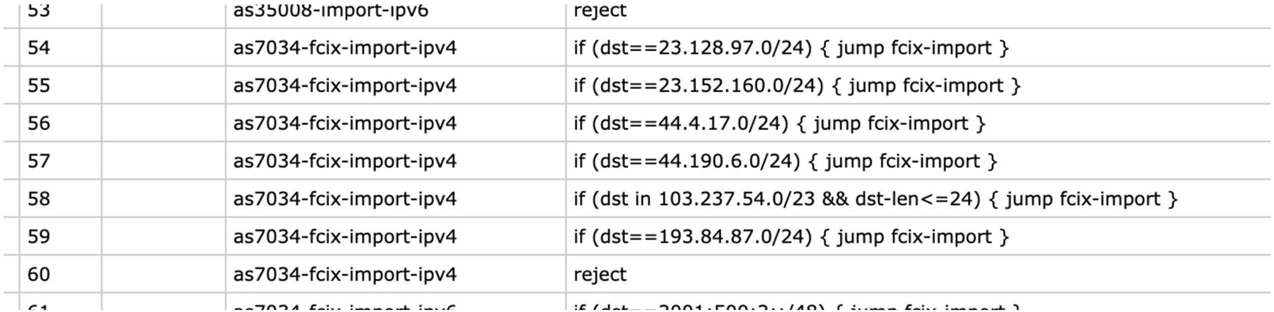
- Script takes the desired output, as an expression of the rules to be applied to the router
  - check the router configuration to see if this matches what's currently running - regardless of ordering
  - and if not, update it
  - wrap it all in bash to loop through your peers!



```
Adding: Chain: as32934-sfmix-import-ipv6, Rule: if (dst in 2620:10d:c090::/44 && dst-len<=48) { jump sfmix-import }
Adding: Chain: as32934-sfmix-import-ipv6, Rule: if (dst in 2620:13e:100c::/46 && dst-len<=48) { jump sfmix-import }
Adding: Chain: as32934-sfmix-import-ipv6, Rule: if (dst in 2620:13e:1000::/44 && dst-len<=48) { jump sfmix-import }
Adding: Chain: as32934-sfmix-import-ipv6, Rule: reject
Grabbing prefixes for AS7034
as7034-fcix-import-ipv4 matches - No update required
as7034-fcix-import-ipv6 matches - No update required
Grabbing prefixes for AS7500
```

Deploying Strict IRR Filters… part 3

- Some examples in github, to show chaining policies together
  - We're using a "slug"/name for each IX – example here shows fcix
  - Allows chaining of policies, in the example here – sending to a fcix-import policy to apply some communities and other TE policy, then accept the prefixes
  - Specific import filter then applied to the neighbor config



53		as35008-import-ipv6	reject
54		as7034-fcix-import-ipv4	if (dst==23.128.97.0/24) { jump fcix-import }
55		as7034-fcix-import-ipv4	if (dst==23.152.160.0/24) { jump fcix-import }
56		as7034-fcix-import-ipv4	if (dst==44.4.17.0/24) { jump fcix-import }
57		as7034-fcix-import-ipv4	if (dst==44.190.6.0/24) { jump fcix-import }
58		as7034-fcix-import-ipv4	if (dst in 103.237.54.0/23 && dst-len<=24) { jump fcix-import }
59		as7034-fcix-import-ipv4	if (dst==193.84.87.0/24) { jump fcix-import }
60		as7034-fcix-import-ipv4	reject
61		as7034-fcix-import-ipv6	if (dst==2001:500:2::/48) { jump fcix-import }

Questions?

Find the code on Github

https://github.com/edgenative