IRR Lockdown

- or -

How to make the most of existing IRR systems

Job Snijders

job@ntt.net

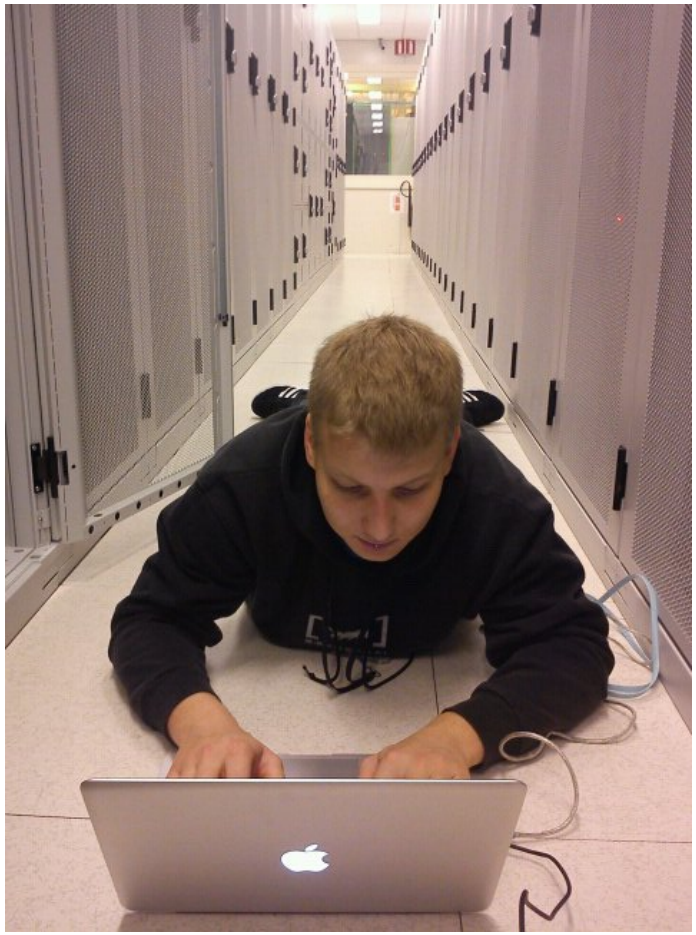
Who am I?

Job Snijders

IP Development @ AS2914 / NTT

Twitter: @JobSnijders
Email: job@ntt.net

Shoe size: 45/EU



Agenda

- What is IRR?
- Issues with IRR
- Protecting RIPE managed space “IRR lockdown”
- New IRR debug tool: “IRR Explorer”
- Q & A

IRR background

- IRR/RPSL is decades old technology
- Register routes which you intend to announce
- Some people classify IRR as garbage:
  - Stale data
  - No incentive to clean up
  - ***Almost*** no verification
  - RPSL is close to non-deterministic shit to parse
- Google keywords: RIPE, RADB, ARIN, “route:”, “route6:”, “aut-num:”, “inetnum:”, “inet6num:

** But there is hope…..

Issues

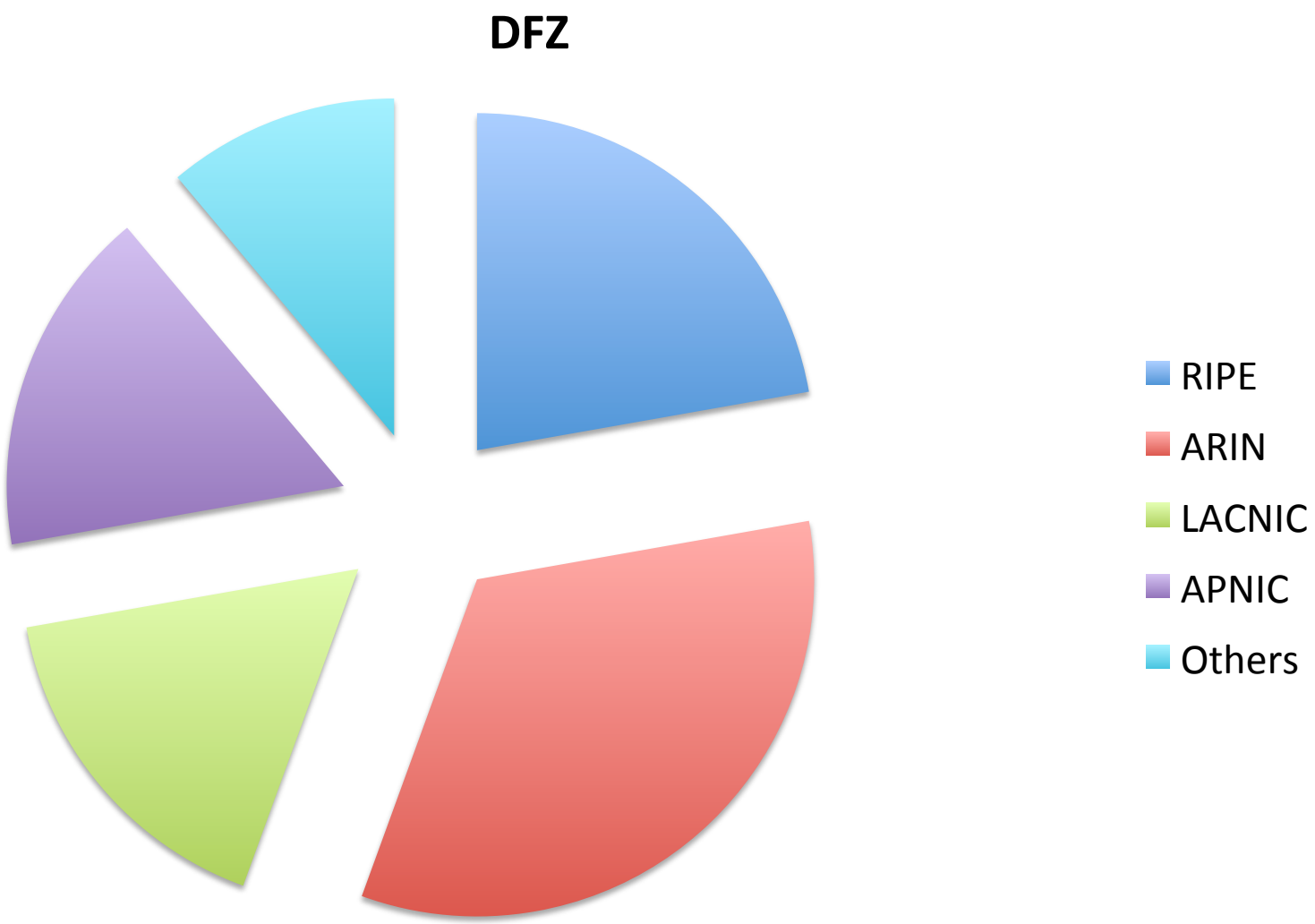
- http://www.bgpmon.net/using-bgp-data-to-find-spammers/
- "Spamming with BGP Spectrum Agility" http://nanog.org/meetings/nanog36/presentations/feamster.pdf
- Mistakes clings to you forever…

What is an IRR lockdown?

- Only honor route objects when they come from the right data source AND have been properly authenticated

- Ignore route objects covering the “locked down” IRR if they come from elsewhere

PIECHART TIME!11 (DFZ)



Quality differences

- Coupling between RIR & IRR functionality?
  - Does the IP owner have to authorize route object creation?
- Verification queuing?
- Yearly payment as keep-alive?
- 24/7 support staff?
- Easily accessible training?
- ….

What makes RIPE's database special?

- NTT can trust the authentication chain from IP block owner (*"inetnum owner")* down to route object creation: both aut-num owner and inetnum owner have to approve
- RIPE arguably has infrastructure in place for good registry service uptime
- RIPE leads 'whois server' software development

The plan

Knowing that: RIPE administrates roughly 35 /8 blocks

NTT considers only to allow route objects, covering RIPE managed space, to influence NTT prefix filters, if the objects come from RIPE's registry itself.

- Ignore certain updates on NRTM streams
- Reject certain route object creation in NTTCOM registry

What are Untrusted NRTM/IRR updates?

- Anything that any IRR sends to **rr.ntt.net** via NRTM, which covers part of the 35 /8s RIPE NCC manages
- Anything with “source: RIPE” from non-RIPE NRTM server
- Any route objects customers create which covers RIPE managed space inside the NTTCOM registry

Benefits

- It will become harder to hijack RIPE space through rogue route objects in lenient registries
- NTT (and other parties using **rr.ntt.net** to generate filters) can trust the filters better
- Clear benefit to register route-objects in same database as the inetnum: (APNIC -> APNIC, RIPE -> RIPE, AFRINIC -> AFRINIC)
- Proper registration will actually mean something, and have impact on global scale

Statistics (28 nov 2014)

1. Total number of RIPE prefixes for which a route object ONLY exists in a foreign IRR AND which were observed in the DFZ: **1004 prefixes** (aggregated 522), spread over 280 ASNs.
2. Total number of prefixes for which a route object exists in both RIPE IRR and a foreign IRR (with mismatching origins), AND where the foreign version is observered in the DFZ: **269 prefixes** spread over 119 ASNs.
3. Combined intersection of #1 and #2 with our customer cone: ~ ***500 prefixes***

Details:
https://www.ripe.net/ripe/mail/archives/routing-wg/2014-November/002887.html

IANA gave to RIPE 193.0.0.0/8:

```
inetnum:        193.0.0.0 - 195.255.255.255
netname:        EU-ZZ-193-194-195
descr:          European Regional Registry
```

Good:

```
route:   193.0.0.0/21
descr:   RIPE-NCC
origin:  AS3333
mnt-by:  RIPE-NCC-MNT
source:  RIPE
```

BAD!

```
route:   193.0.0.0/21
descr:   RIPE-NCC
origin:  AS666
mnt-by:  MAINT-AS237
source:  RADB
```

Why would we ever honor the bad route object?!

Implementation steps:

- Patch IRRd to have shim layer between NRTM receiver and internal radix tree/db, which checks whether the route object comes from RIPE IRR and covers RIPE space
  - In case of failure: insert into DB with **AS0 as origin**
  - Source: https://github.com/irrdnet/irrd/
  - **rr.ntt.net** runs IRRd
- On a daily basis fetch list of RIPE managed prefixes

IRR EXPLORER

Why IRR Explorer?

- Make it easier to test whether an “IRR Lockdown” would affect you
- Get a sense of where people registered route-objects covering your own space
- Motivate people to clean up! ☺
- I grew tired of

```
$ peval | awk | sed | derp | help | echo “get me out of here” | wall –g
```

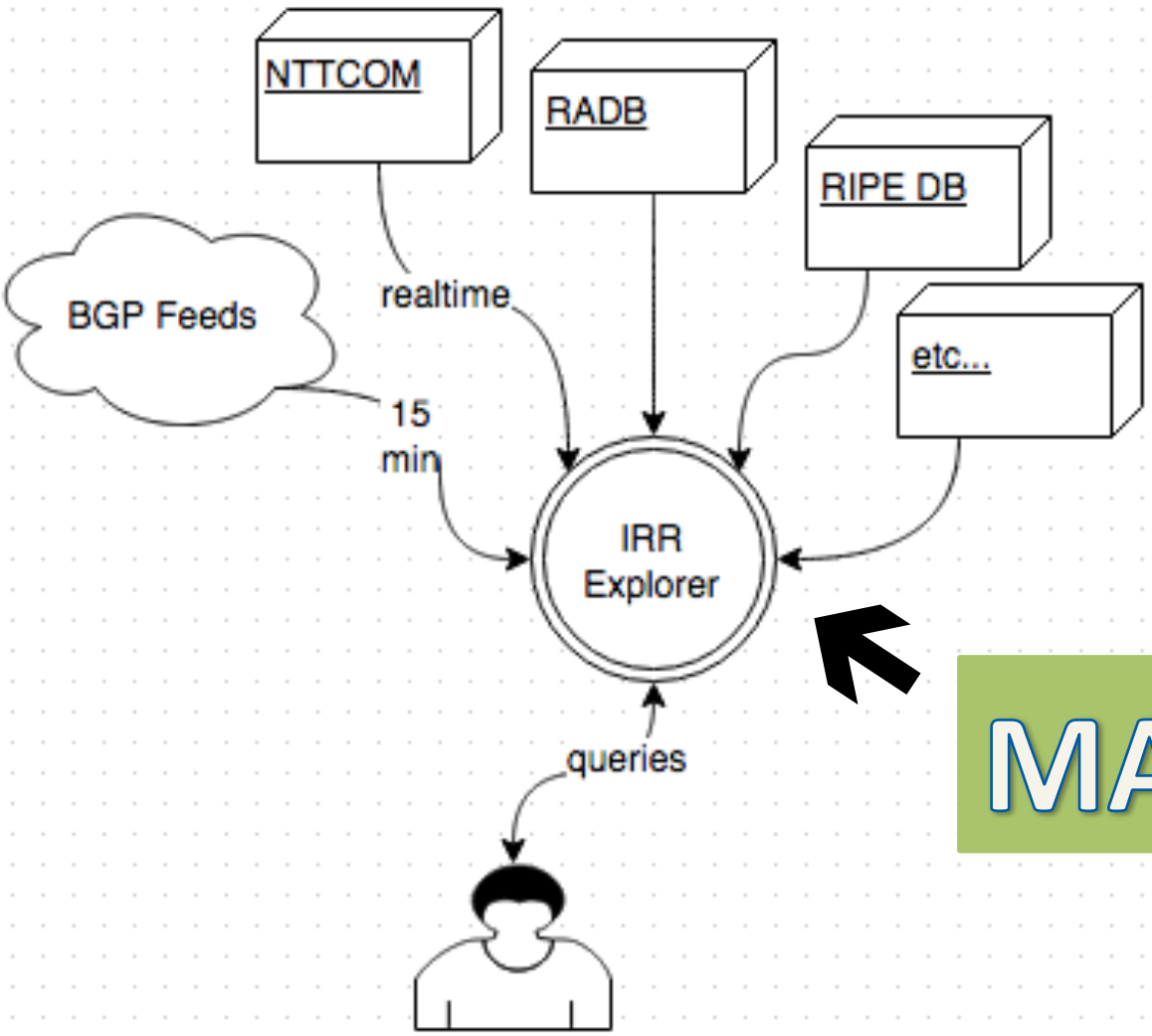
Debugging your IRR data

IRR Explorer

http://irrexplorer.nlnog.net/ is a tool to search where your IRR objects are located and see if they are in the proper database or not

Code: https://github.com/job/irrexplorer

IRR explorer overview

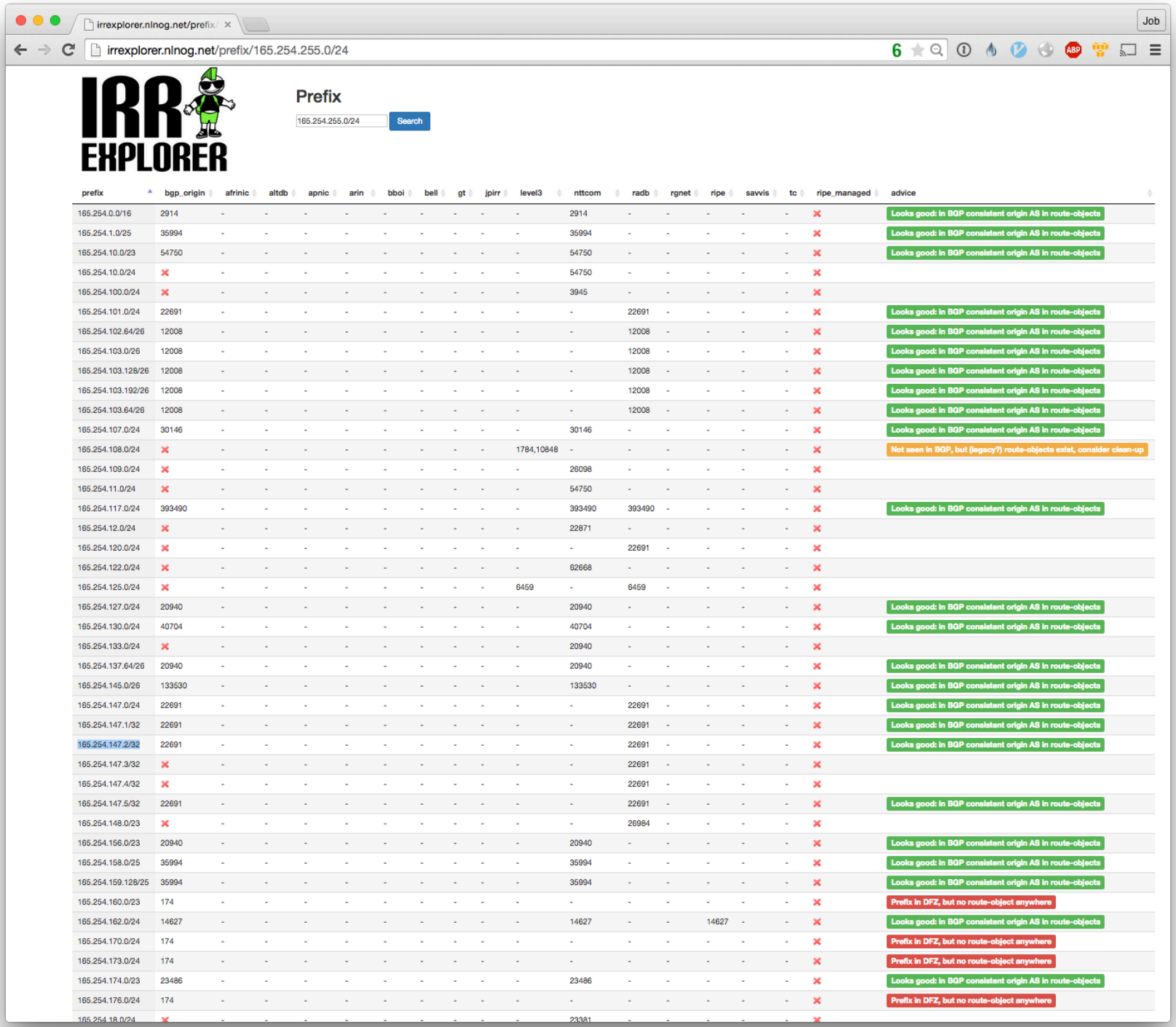


IRR Explorer BETA WARNING

IRR Explorer was hacked together in the last 100 hours with the help of Nat Morris (@natmorris) & Peter van Dijk (@habbie). ☺

IRR Explorer currently is slow and prone to crashes, but that will improve soon!

Ingredients: Python, py-radix, Flask, bootstrap, jquery



prefix	bgp_origin	afrinic	altdb	apnic	arin	bboi	bell	gt	jpirr	level3	nttcom	radb	rgnet	ripe	savvis	tc	ripe_managed	advice
165.254.0.0/16	2914	-	-	-	-	-	-	-	-	-	2914	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.1.0/25	35994	-	-	-	-	-	-	-	-	-	35994	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.10.0/23	54750	-	-	-	-	-	-	-	-	-	54750	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.10.0/24	✗	-	-	-	-	-	-	-	-	-	54750	-	-	-	-	-	✗	
165.254.100.0/24	✗	-	-	-	-	-	-	-	-	-	3945	-	-	-	-	-	✗	
165.254.101.0/24	22691	-	-	-	-	-	-	-	-	-	-	22691	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.102.64/26	12008	-	-	-	-	-	-	-	-	-	-	12008	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.103.0/26	12008	-	-	-	-	-	-	-	-	-	-	12008	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.103.128/26	12008	-	-	-	-	-	-	-	-	-	-	12008	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.103.192/26	12008	-	-	-	-	-	-	-	-	-	-	12008	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.103.64/26	12008	-	-	-	-	-	-	-	-	-	-	12008	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.107.0/24	30146	-	-	-	-	-	-	-	-	-	30146	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.108.0/24	✗	-	-	-	-	-	-	-	-	1784,10848	-	-	-	-	-	-	✗	Not seen in BGP, but (legacy?) route-objects exist, consider clean-up
165.254.109.0/24	✗	-	-	-	-	-	-	-	-	-	26098	-	-	-	-	-	✗	
165.254.11.0/24	✗	-	-	-	-	-	-	-	-	-	54750	-	-	-	-	-	✗	
165.254.117.0/24	393490	-	-	-	-	-	-	-	-	-	393490	393490	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.12.0/24	✗	-	-	-	-	-	-	-	-	-	22871	-	-	-	-	-	✗	
165.254.120.0/24	✗	-	-	-	-	-	-	-	-	-	-	22691	-	-	-	-	✗	
165.254.122.0/24	✗	-	-	-	-	-	-	-	-	-	62668	-	-	-	-	-	✗	
165.254.125.0/24	✗	-	-	-	-	-	-	-	-	6459	-	6459	-	-	-	-	✗	
165.254.127.0/24	20940	-	-	-	-	-	-	-	-	-	20940	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.130.0/24	40704	-	-	-	-	-	-	-	-	-	40704	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.133.0/24	✗	-	-	-	-	-	-	-	-	-	20940	-	-	-	-	-	✗	
165.254.137.64/26	20940	-	-	-	-	-	-	-	-	-	20940	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.145.0/26	133530	-	-	-	-	-	-	-	-	-	133530	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.147.0/24	22691	-	-	-	-	-	-	-	-	-	-	22691	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.147.1/32	22691	-	-	-	-	-	-	-	-	-	-	22691	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.147.2/32	22691	-	-	-	-	-	-	-	-	-	-	22691	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.147.3/32	✗	-	-	-	-	-	-	-	-	-	-	22691	-	-	-	-	✗	
165.254.147.4/32	✗	-	-	-	-	-	-	-	-	-	-	22691	-	-	-	-	✗	
165.254.147.5/32	22691	-	-	-	-	-	-	-	-	-	-	22691	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.148.0/23	✗	-	-	-	-	-	-	-	-	-	-	26984	-	-	-	-	✗	
165.254.156.0/23	20940	-	-	-	-	-	-	-	-	-	20940	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.158.0/25	35994	-	-	-	-	-	-	-	-	-	35994	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.159.128/25	35994	-	-	-	-	-	-	-	-	-	35994	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.160.0/23	174	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✗	Prefix in DFZ, but no route-object anywhere
165.254.162.0/24	14627	-	-	-	-	-	-	-	-	-	14627	-	-	14627	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.170.0/24	174	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✗	Prefix in DFZ, but no route-object anywhere
165.254.173.0/24	174	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✗	Prefix in DFZ, but no route-object anywhere
165.254.174.0/23	23486	-	-	-	-	-	-	-	-	-	23486	-	-	-	-	-	✗	Looks good: in BGP consistent origin AS in route-objects
165.254.176.0/24	174	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✗	Prefix in DFZ, but no route-object anywhere
165.254.18.0/24	✗	-	-	-	-	-	-	-	-	-	23381	-	-	-	-	-	✗	

screenshots

Prefix

193.47.147.0/24 Search

prefix	bgp_origin	afrinic	altdb	apnic	arin	bboi	bell	gt	jpirr
193.47.147.0/24	60564	-	-	-	45671	-	-	-	-

Showing 1 to 1 of 1 entries

level3	nttcom	radb	rgnet	ripe	savvis	tc	ripe_managed	advice
62588	45671	45177,60564	-	60564	-	-	✓	Proper RIPE DB object, but foreign or

Timeline

- Test prefixes @ irrexplorer: now
- Add autnum support to irrexplorer: June
- Add as-set support to irrexplorer: June
- Deploy IRR Lockdown: Q1 2016?

Other parties that (will) IRR Lockdown

- Opteamax
- ECIX route server
- Anybody that uses **rr.ntt.net** in 2016
- You?

Q&A for the routing police