




Tutorial: Running BGP in 2019

Eugene Bogomazov
Qrator Labs

Why we?



- Qrator Labs
 - DDoS mitigation
 - Own Anycast network
-  Radar
 - Where to place nodes?
 - BGP monitoring

What it's all about

- BGP BCP
- Analytics/Toolset
- Modern trends/security
- ROA party
 - But without booze :(

What it's not about

- Inner ISP structure
- Command examples
- Configuration automation
 - Templates
 - Validation
 - Deployment
- ***Must to follow*** recommendations

Ideal world

- Read the standards
- Get ASN and prefixes
- Set a configuration
- And that's all...

But...



Cruel world

- Implementation doesn't follow standards
- Cheap/old hardware
- Incorrect/outdated configurations
- Curved arms
- Attackers

We have a situation

- Nobody likes problems
 - At least their own
- Guiding stone
 - Prevention
 - Investigation
 - Mitigation

What it's all about

- **BGP BCP**
- Analytics/Toolset
- Modern trends/security
- ROA party

Where to start?

- We need an operational BGP BCP!
 - But we already have one ([BCP194](#))
- Snijders presentations ([here](#) or [here](#))
- Example of filters - <http://bgpfilterguide.nlnog.net/>

TCP security

- Basic ACL
- TCP MD5/TCP AO
- GTSM (aka TTL hack)
 - Even in case of multihop sessions

Prefix filter. Denial

- Special use
- Too specific
- Default
- From unwanted directions
 - IXP prefixes
 - Local prefixes

Prefix filter. Permission

- Static customer prefixes whitelist
 - Just no
- IANA allocated space
 - Only for v6
- Dynamic filter based on IRR information
 - It's having its own problems
 - *Later...*

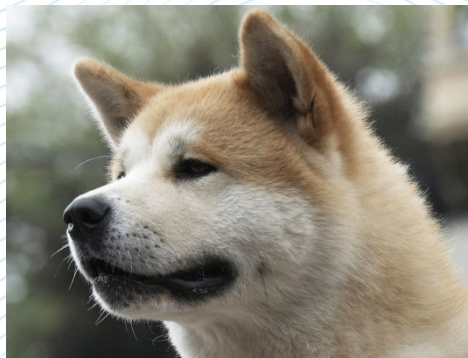
AS_PATH filter

- Not covered by BCP
- Neighbor check
- Bogon ASN
- TIER_1 filtering
- *Other types?*

Leakage prevention

- Route Leak — violation of policy
- Prevent with custom communities
- Prefix rate limiting
- Automation with **BGP Roles**

- One problem:



Community

- A way of marking routes
 - Carry information
 - Policy implementation
- Large Communities - [how to?](#)
- Study your upstream capabilities
 - Geo/blacklist/"you name it" communities

Aggregation (rfd)

- Pretty often appeared ([example](#))
- Decrease prefix max limit?
- BGPSec and ROA?
 - In case of updating routes
- Customers PI blocks?
 - In case of announcing own prefixes
- Do we really need it?

**rfd: request for discuss*

AS_PATH length (rfd)

- No limitation in RFC
- Someone was not ready (case, another)
- Real distance is not so big
- Possible filtration can be applied
- But why?
 - And where is a border?

What it's all about

- BGP BCP
- **Analytics/Toolset**
- Modern trends/security
- ROA party

Is something wrong?



- Business metrics
 - Traffic drop
 - Traffic raise
 - Decreased number of users
 - Increased number of tickets
-
- Or just intuition

Yeah, investigation!

- Nature of a problem
 - Is it a BGP problem?
- What happened?
- Who is responsible?
- What to do next?

Toolset

- Ripe Atlas
 - Ripe Widgets
 - Public API
 - Looking glasses
 - Whois
-
- One possible example (not free)

Route analytics



- BGP raw data
 - Ripe RIS, RV, PCH
- From MRT to human
 - [bgpdump](#), [bgpstream](#)
- History player
 - [BGPlay](#) (but with 6 hour delay)

In moment



- Ripe Atlas ([framework](#))
 - But credits...
- Looking glasses
 - But pain (customized, without API, etc)
- Alternatives?
 - [Ripe... API!](#) (yeah, they have one)
 - [Radar API](#) (only paths)

Why we need it?

- Find an attacker
- AS_PATH manipulation?
 - Collect many routes
 - And neighbor check will save a day!
 - Except not always...
- In case of bad IP — find an owner

Todo:

- Find abuse contact
 - Ripe [method](#) (or [our](#))
- Write a letter
- Wait

- Or write a letter to special mailing list
 - And wait

Active?

- Create special filter
- Your address space under attack?
 - Hijack — more specific announce
 - or delegate
 - Route leak — AS_PATH manipulation

What it's all about

- BGP BCP
- Analytics/Toolset
- **Modern trends/security**
- ROA party

Main trends



- Prevent accidental errors
 - Make explicit policies
 - Fail-safe strategy
 - Bring some automation in process
- Making life a little bit easier
 - Large Communities
 - Shutdown communication
 - Graceful shutdown

uRPF (rfd)

- Anti spoofing technique ([BCP38](#), [BCP84](#))
- Based on BGP
 - Which is asymmetric
- Main enemy — TE
- [Modern draft](#) is under development
 - Problem remains the same
- Someone use it?

Security

- Anything in route can be changed
 - Anything
- Security by claiming
 - Route objects/IRR
 - ROA/RPKI
- Secure AS_PATH

IRR

- Create a Customer Cone with AS_SET
 - We all know problems
- Choose route objects
- Create a prefix filter

IRR (rfd)

- Some IRR are not trusted
- No maxLength at all
 - Exact/covered filter type
- No any formal policy
 - Are we ready to define one?
 - What to do with delegated prefixes?

Hello from the past

How to Sign It?

It's simple:

1. <https://my.ripe.net/#/rpki>
2. Sign only aggregates;
3. Set max_length to 32 in IPv4 (128 in IPv6);

** From last ROA signing party*

O RLY?

How to Sign It?

It's simple:

1. <https://my.ripe.net/#/rpki>
2. Sign **only aggregates**
3. Set max_length to **32** in IPv4 (**128** in IPv6);

** From last ROA signing party*

ROA (rfd)

- Which maxLength to use?
 - Valid cases vs hijacks
 - Don't use at all?
 - Don't need to be max for blackhole?
- Less specific is «Not found»
 - Great with uRPF

AS_PATH manipulation

- Based on loop detection
- Route Leak prevention
- Link load balancing
- Link overloading
- Pilosov-Kapela (real example)

AS_PATH verification

	BGPSec	ASPA
Main goal	Stop crafted routes	Stop global propagation
AS_PATH + NLRI	Yes	Only AS_PATH
AS_PATH validation	Is real?	Is valid?
Cryptographic load	For each route in each direction	Only during filter creation
Partial deployment	For «connected islands»	For independent deployment
Prevent route leaks	With draft extension	As a side effect
Status	RFC; not spreaded	Draft; waiting

What it's all about

- BGP BCP
- Analytics/Toolset
- Modern trends/security
- **ROA party**

ROA regional status



- Around 21% of prefixes are signed
 - Really good coverage
- More than 100 ISP are fully signed!
- Good big boys: AS198605, AS43451

How to check



Via any BGP prefix monitor

- https://bgp.he.net/AS197068#_prefixes
- <https://radar.qrator.net/as197068/prefixes>

- Via API

- https://stat.ripe.net/docs/data_api#rpki-validation
- <https://api.radar.qrator.net/#/Connectivity/prefixes>

- Via dashboard

- <https://my.ripe.net/#/rpki>

What to do next?

- Own repository — great
 - But we are lazy
- Choose wisely your prefixes and maxLength
- Go to RPKI dashboard <https://my.ripe.net/#/rpki>
- Follow intuition
 - Or just [read/watch](#)



Questions?

Contacts: eb@qrator.net

PS: My questions (rfd)

- Do we need to aggregate routes?
- Do we need to standardize AS_PATH max length
- Does someone use uRPF? Not the loose one?
- What to sign in ROA?
- Is «Not Found» for less specific in ROA good?
- Do we need to create a standard policy for filtration based on IRR?