

Riot **Direct** Network

Our Network as Code Journey

Jaime Botello

Principal Network Engineer, Riot Games

@fath3r



League of Legends



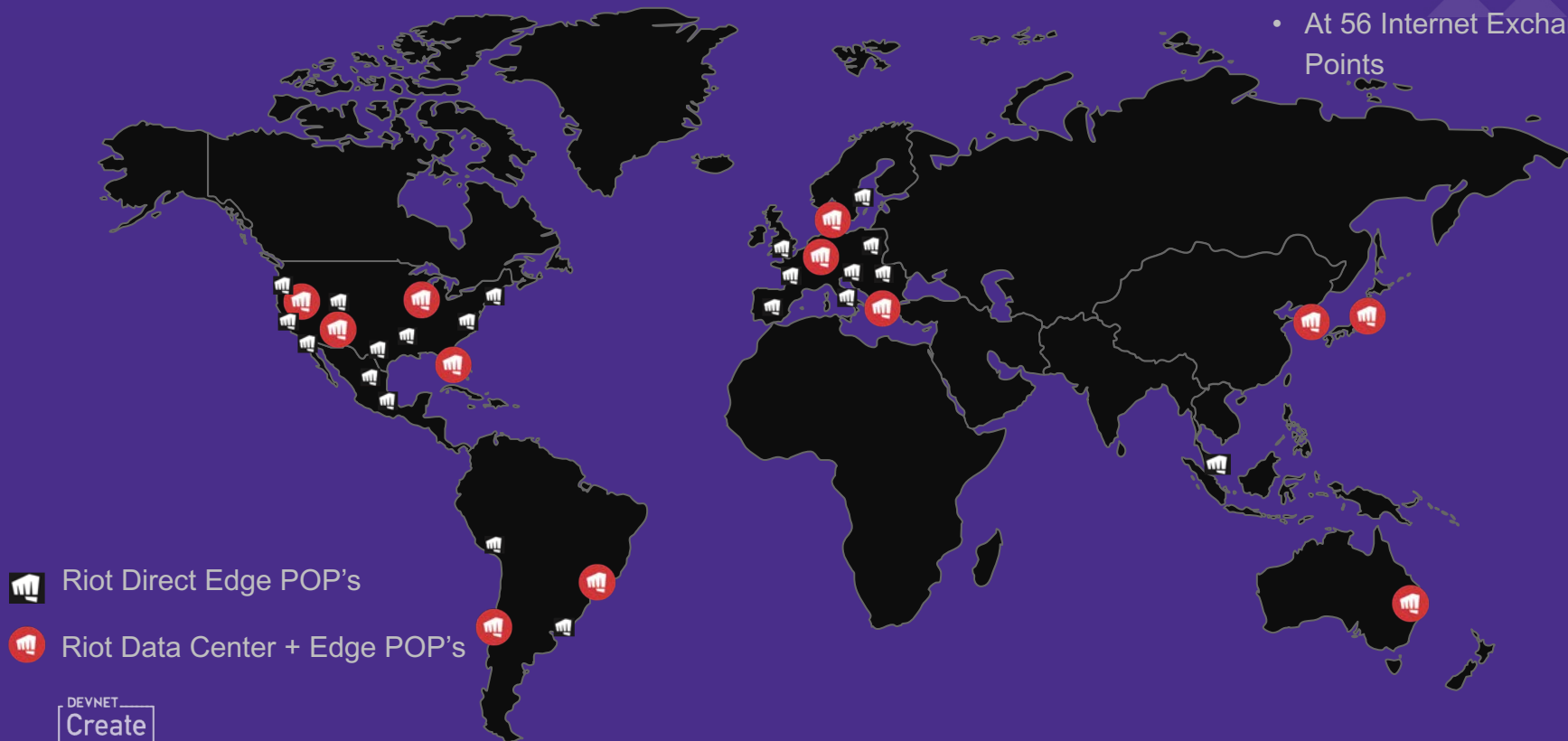
New Releases...



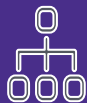
Riot Games Infrastructure

Riot Direct

- Around 2500 eBGP sessions with over 300 ISP's
- At 56 Internet Exchange Points



How we used to **make** changes



Success



Rollback



Intent

Review

Deployment

Validation



The
Engineer



(Hopefully)
a different
Engineer



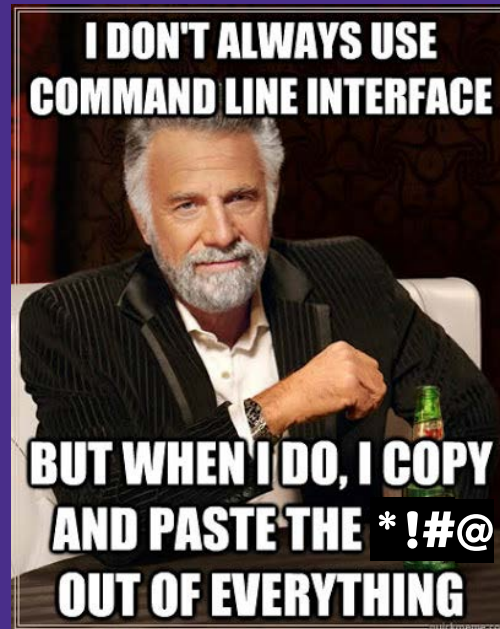
(the same)
The Engineer



We have
this
right?



rollback? Add
no to commands
then copy &
paste



Riot Direct **Network** as Code

*“Is how we define and **manage** riot direct network infrastructure via machine **friendly** declaratives using software **versioning** control and delivering changes via continuous integration and continuous **delivery** pipelines (CI/CD)”*

Riot Direct **Network** as Code

Key Features

- ✓ **Master** branch is the Single Source of Truth (SSoT) and represents the state of the **Production** Network
- ✓ Minimal level of **abstraction**
- ✓ Full configuration **replace** is OP
- ✓ Jenkins CI/CD **tests** and **deploys** configuration to devices with some fault-tolerance
- ✓ Github Source Control allows easy **versioning** and **rollback**
- ✓ Actively maintained **open source** projects (Nornir, napalm, capirca, etc) are utilized to **speed up** development

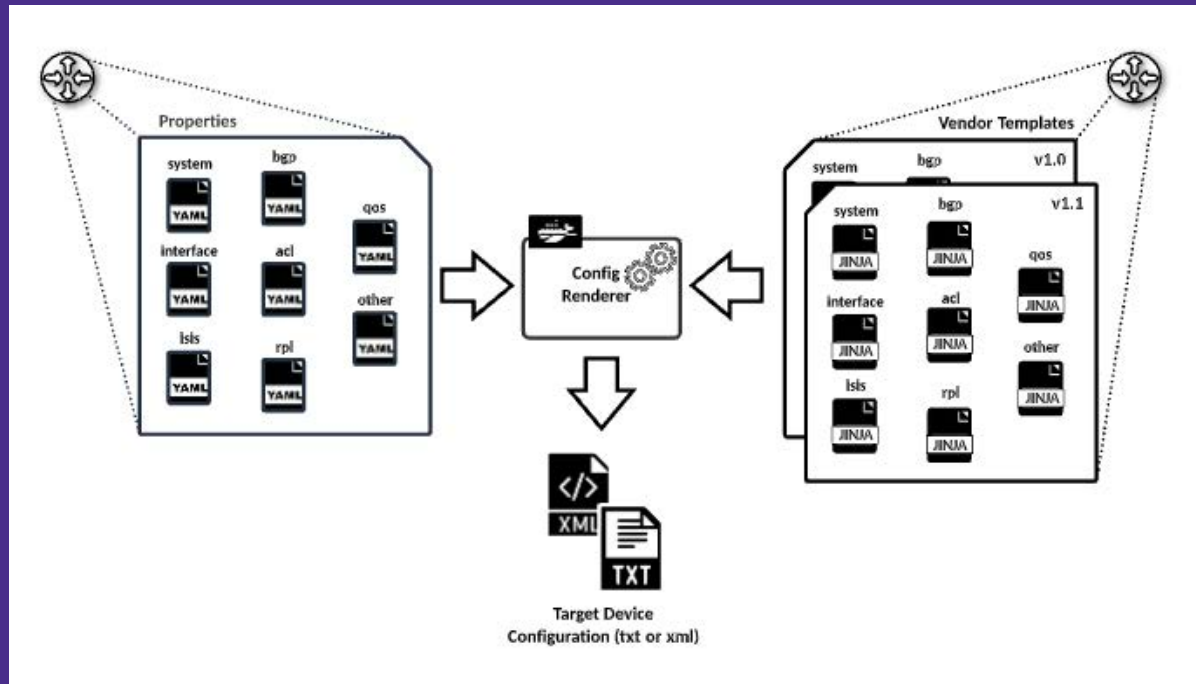
Architecture // Overview

The image displays a GitHub repository interface for `rdnw-as-code-training` on the `master` branch, specifically showing the `er01/bgp.yaml` file. The file content is as follows:

```
1 global:
2   asn: 6507 # Local BGP Autonomous System Number (ASN)
3   router_id: 10.255.255.1 # ipv4 address BGP Router ID
4   ecmp_path: 8 # iBGP/eBGP ECMP Maximum Paths (ex. 2, 4, 8)
5
6 groups:
7   ROUTE_REFLECTORS:
8     type: route_reflector
9     description: iBGP IPV4 and VPNV4 Route Reflectors
10    enabled: true
11    update_source: Loopback0
12    afft:
13      ipv4:
14        import: IMPORT_RR_V4
15        export: EXPORT_RR_V4
16    sessions:
17      10.255.255.10:
18        description: iBGP Session
19        enabled: true
```

Overlaid on the right is a network architecture diagram. It illustrates a multi-region setup with two main regions, EU and APAC, each containing a set of routers. These regions are connected to a central cloud environment (represented by a cloud icon) via Webhook connections. The cloud environment is further connected to a set of routers labeled PALM and NAPALM. The diagram also shows a hierarchy of files and folders, including `tools`, `main`, `inventory`, `pop_build`, `deployer.py`, `hosts.yaml`, `rr.ams02`, `properties (v1)`, `node(n)`, and `properties (v1)`.

Device Model



Tracking change in High Definition

The screenshot displays a network configuration management interface. On the left, a file explorer shows a directory structure: `sites` (expanded) → `apac` (expanded) → `hkg01` (expanded) → `se01` (expanded) → `sin02` (expanded) → `er01` (expanded). Under `er01`, several YAML files are listed: `bgp.yaml` (selected), `exception.yaml`, `interfaces.yaml`, `mpls.yaml`, `routing.yaml`, `rpl.yaml`, `vrf.yaml`, and `er02`.

In the center, a table shows commit history for the selected file:

Line	Date	Author	Count
5	9/9/19	jbotello	1
6	9/9/19	jbotello	1
7	9/16/19	Botello	3
8	3/19/20	Botello	30
9	9/9/19	jbotello	1
10	9/12/19	Botello	2
11	9/12/19	Botello	2
12	9/12/19	Botello	2
13	9/12/19	Botello	2
14	9/12/19	Botello	2
15	9/12/19	Botello	2
16	9/9/19	jbotello	1
17	9/12/19	Botello	2
18	4/6/20	Sheikh	31
19	9/9/19	jbotello	1
20	9/12/19	Botello	2
21	9/12/19	Botello	2
22	4/6/20	Sheikh	31
23	9/12/19	Botello	2
24	9/12/19	Botello	2

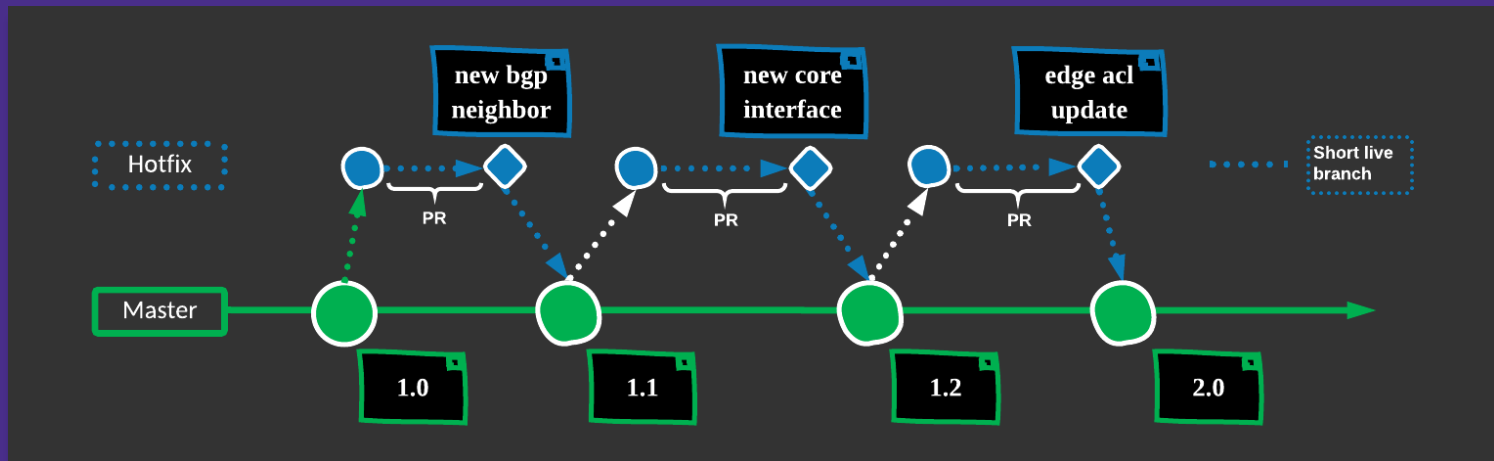
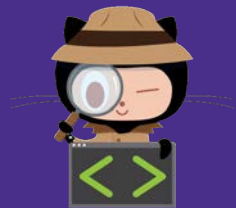
On the right, a configuration snippet for `ROUTE_REFLECTORS` is shown:

```
groups:
  ROUTE_REFLECTORS:
    type: route_reflector # public | transit | customer | internal | vpn
    aft:
      ipv4:
        import: IMPORT_RR_V4
        export: EXPORT_RR_V4
      vpnv4:
        import: PASS
        export: PASS
    update_source: loopback0 # Update source interface
    description: IPv4 and VPNv4 Route Reflectors # Just a description
    enabled: true # True | False
    sessions:
      104.160.157.31:
        description: iBGP session to rr01.sel01
        enabled: true # True | False
      104.160.157.32:
        description: iBGP session to rr01.ty001
```

A commit message is visible in the background: "Botello, 9/12/19, 8:27 PM • Updating both production bgp and rpl configuration for er01.lab01".

How a **network** change looks like?

Github



Change // Network Engineer Workflow

01

Config change

Make device config changes on new branch (ex. hotfix/new_change)

git checkout ..
git add ..
git commit ..

2

1

3

4

02

Pull Request

Open a github pull request (PR)

webhook

5

Spin up docker container with The Deployer tools

Pass parameters

6

Load config to target nodes (dry-run=True)

Config diff Success or Failure

7

Config diff Success or Failure

Add config diff and report to PR comment

7

8

Peer Review and Approval

9

merge to mater

webhook

12

Final Diff

Pass parameters

10

Final Diff

12

Load config to target nodes (dry-run=False)

Final Diff

12

The Network

Network Engineer

GitHub

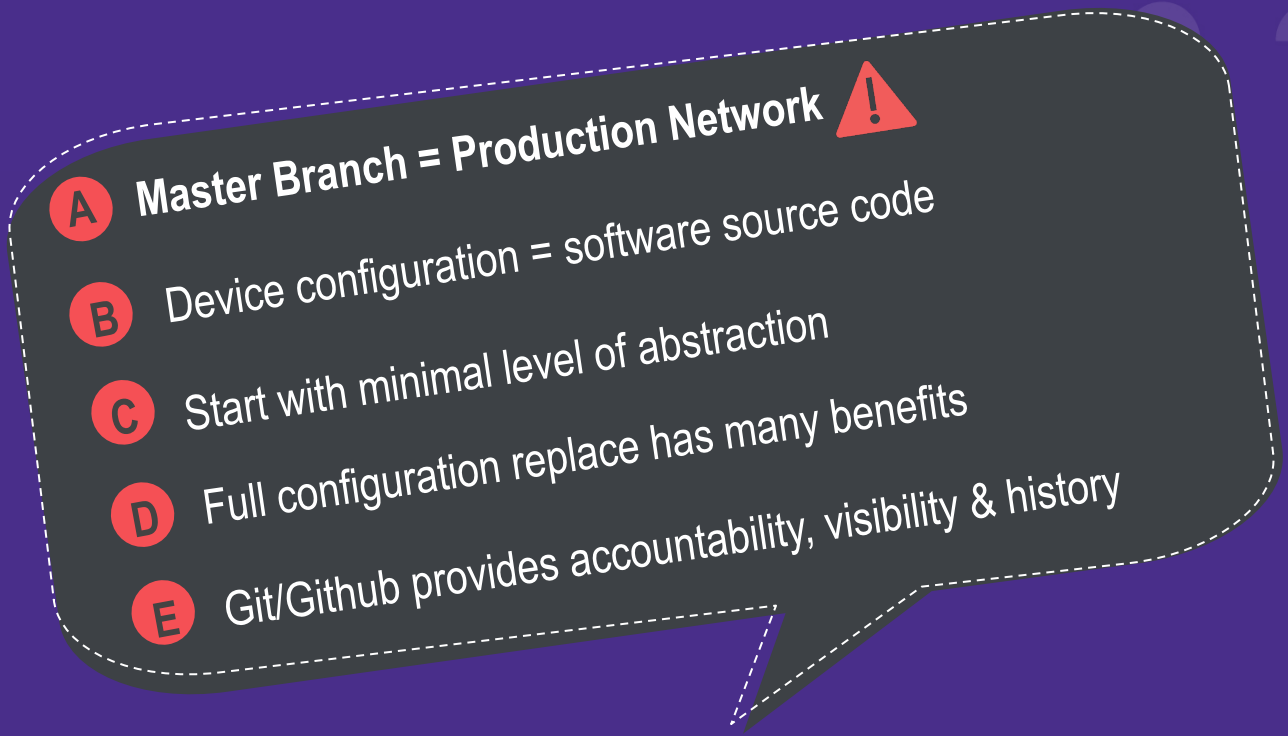

Jenkins

Python Code



Stop!!! Demo Time



- 
- A** Master Branch = Production Network 
 - B** Device configuration = software source code
 - C** Start with minimal level of abstraction
 - D** Full configuration replace has many benefits
 - E** Git/Github provides accountability, visibility & history

Takeaways



DEVNET Create