EMnify

How our Cloudy Mindsets Approached Physical Routers

SNMP was not an option

Steffen Gebert

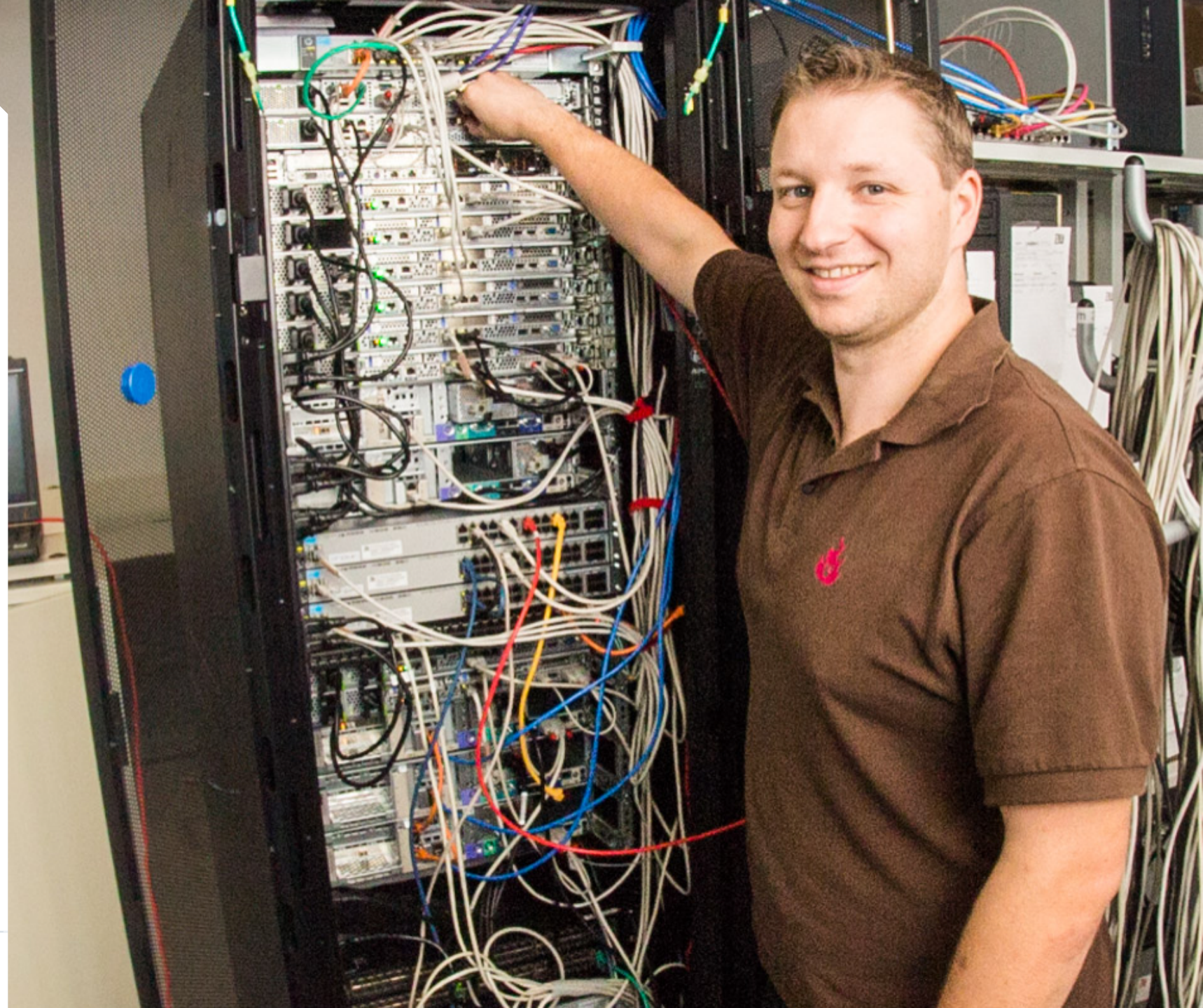
DENOG12, 09.11.2020

Cloudy Mindset?

09/
11/

5 years ago

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG



I 3-10 years ago

I 0-3 years ago

EMnify

11/
9/2

Is This a Better World?

09/
11/

Focus on Business Value

Prefer Managed Services

09/
11/

And Suddenly… Hardware?

Agenda

1. Context
2. More Context
3. Deployment
4. Monitoring
5. Conclusion

Move??
Our scale[throughput] bores you

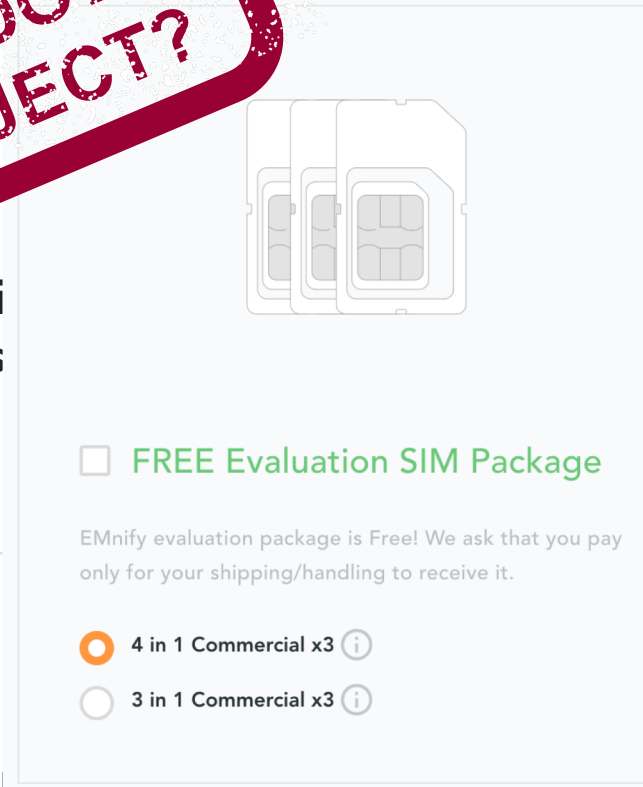
Connectivity Platform for the Internet of Things

EMnify's Connectivity Platform

Cellular connectivity in 580 networks in 190 countries

APIs

Pri
pay-as



SMS/USSD <> REST bridge

VPN / AWS TGW

Impleme
own virtualized mobile core network

Supporting Global IoT Deployments

Traditional Operators

Home-routing of roaming SIM data prevents distributed architecture

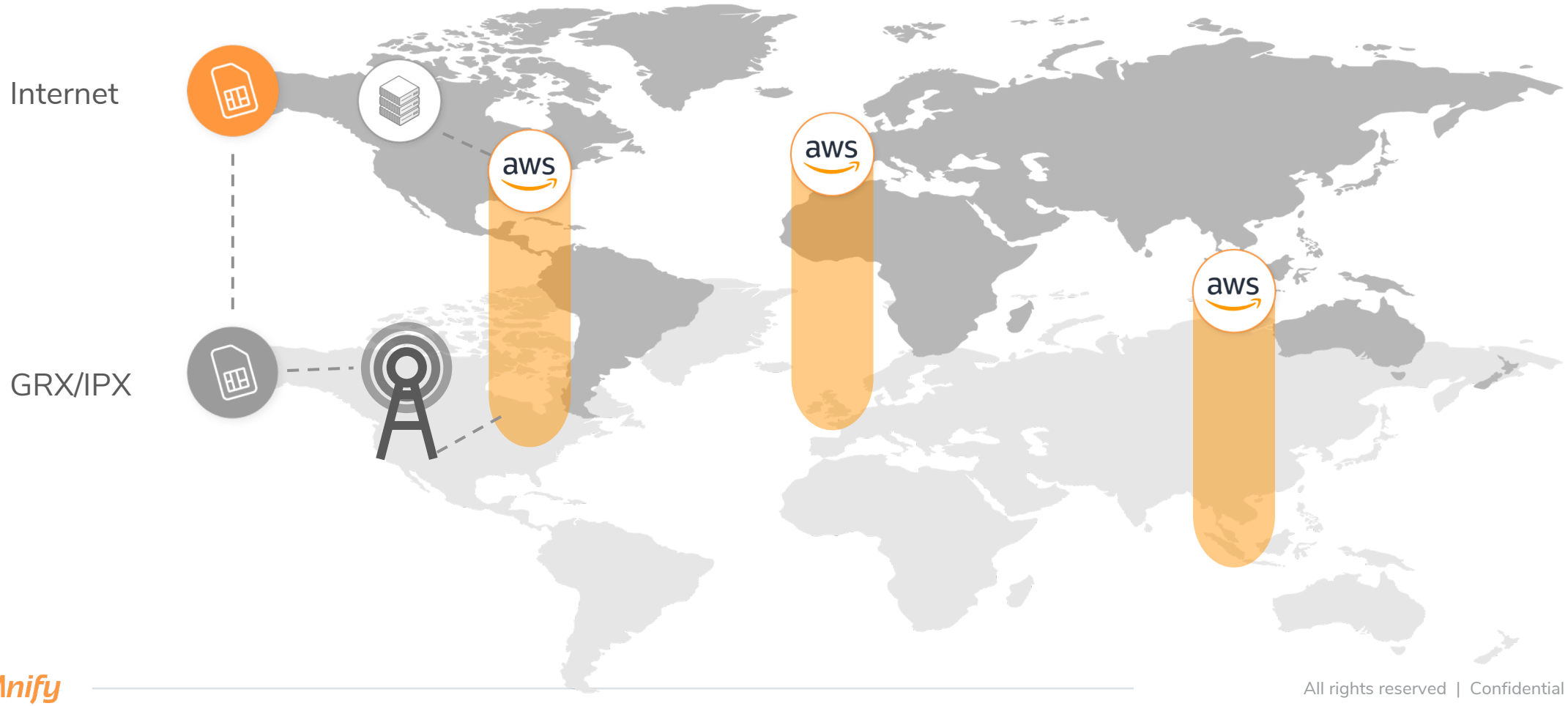
EMnify Connectivity

EMnify's mobile core network is deployed in multiple AWS regions – keeping data local

We’re critical to our customers’ success

Increased demands vs. AWS as “General Purpose Cloud”

GRX/IPX Network (GPRS Roaming Exchange)



09/11/

Running BGP on AWS?

We had to move logic out of AWS

Solution Space

Managed Service

Software Router (TNSR)

Linux Switches (Cumulus)

Router (Juniper)

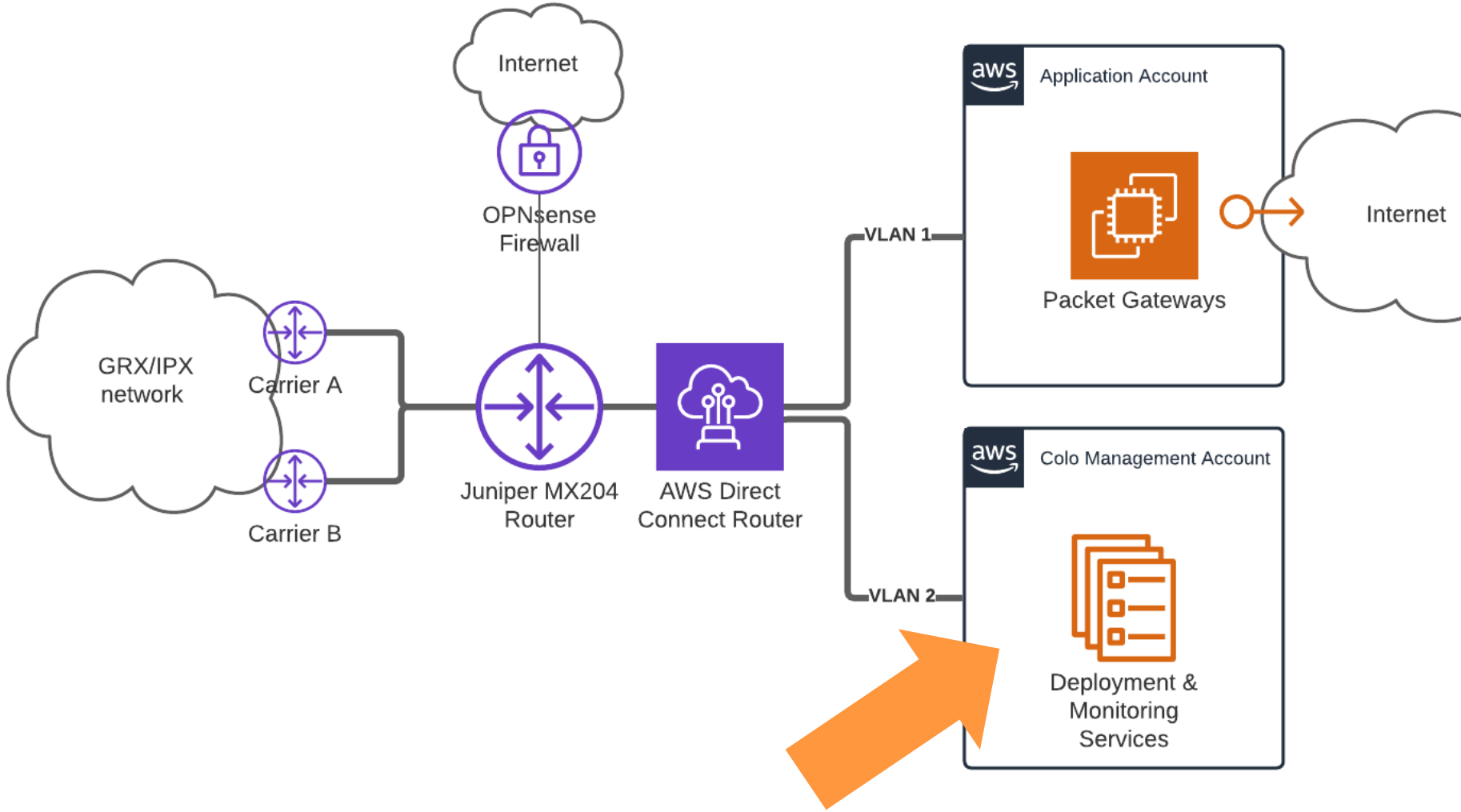
We could not find a fitting managed service

We had to buy hardware

We chose boring technology

Setup

- 2x Juniper MX204
- Colocation rack space
- Fiber links towards
  - 2 carriers
  - AWS
- Out-of-band management access via OPNsense



Why am I here today?

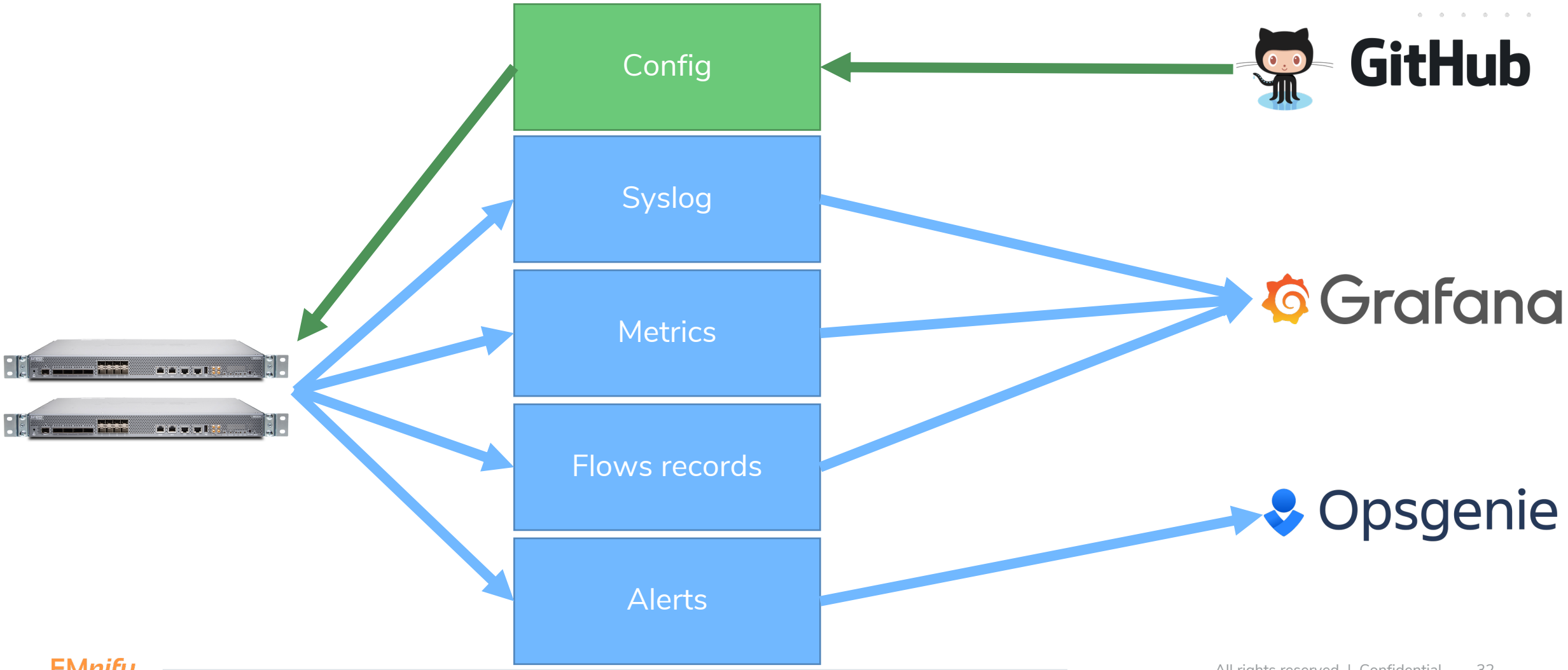
You’re not doing it wrong!

Greenfield project

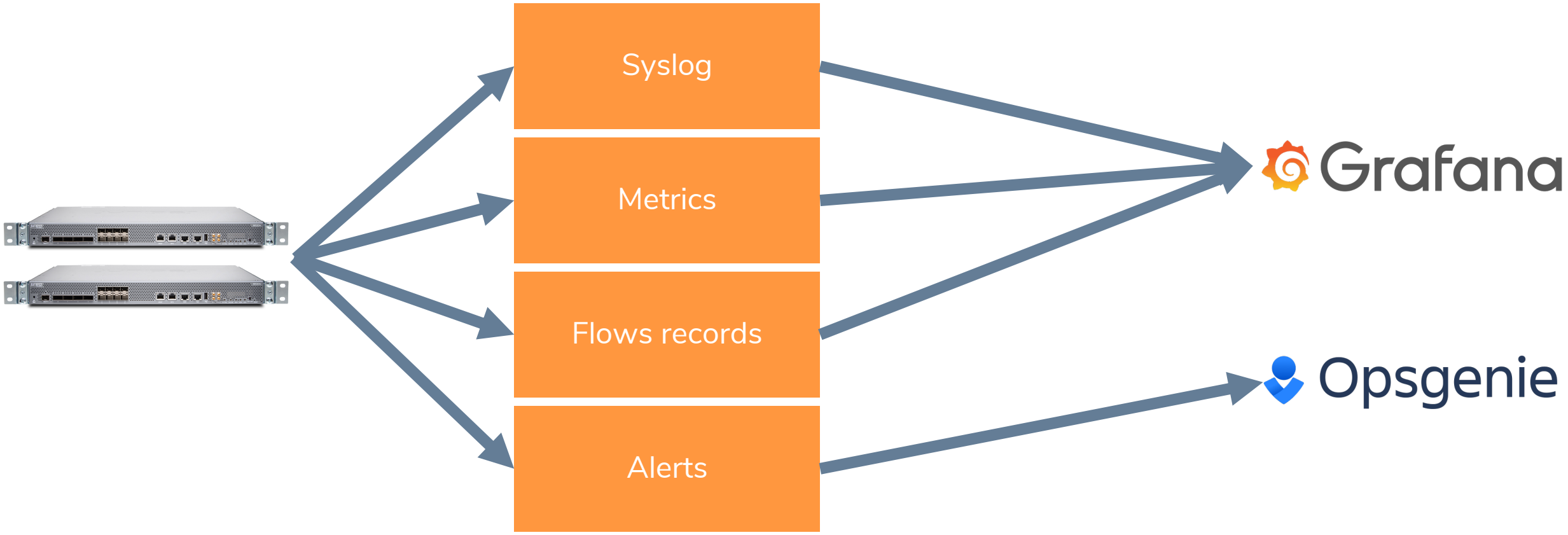
Scratching the surface only

Topics

Integration Points



Monitoring



Design Principles

- Don’t get too much out of our comfort zone
- Prefer managed services, whenever possible
- Don’t run anything that requires lot of handholding
- When this is set up, it will not change frequently
- If those things break, the service (probably) is still fine
- Uncool: start a VM

Design Principles

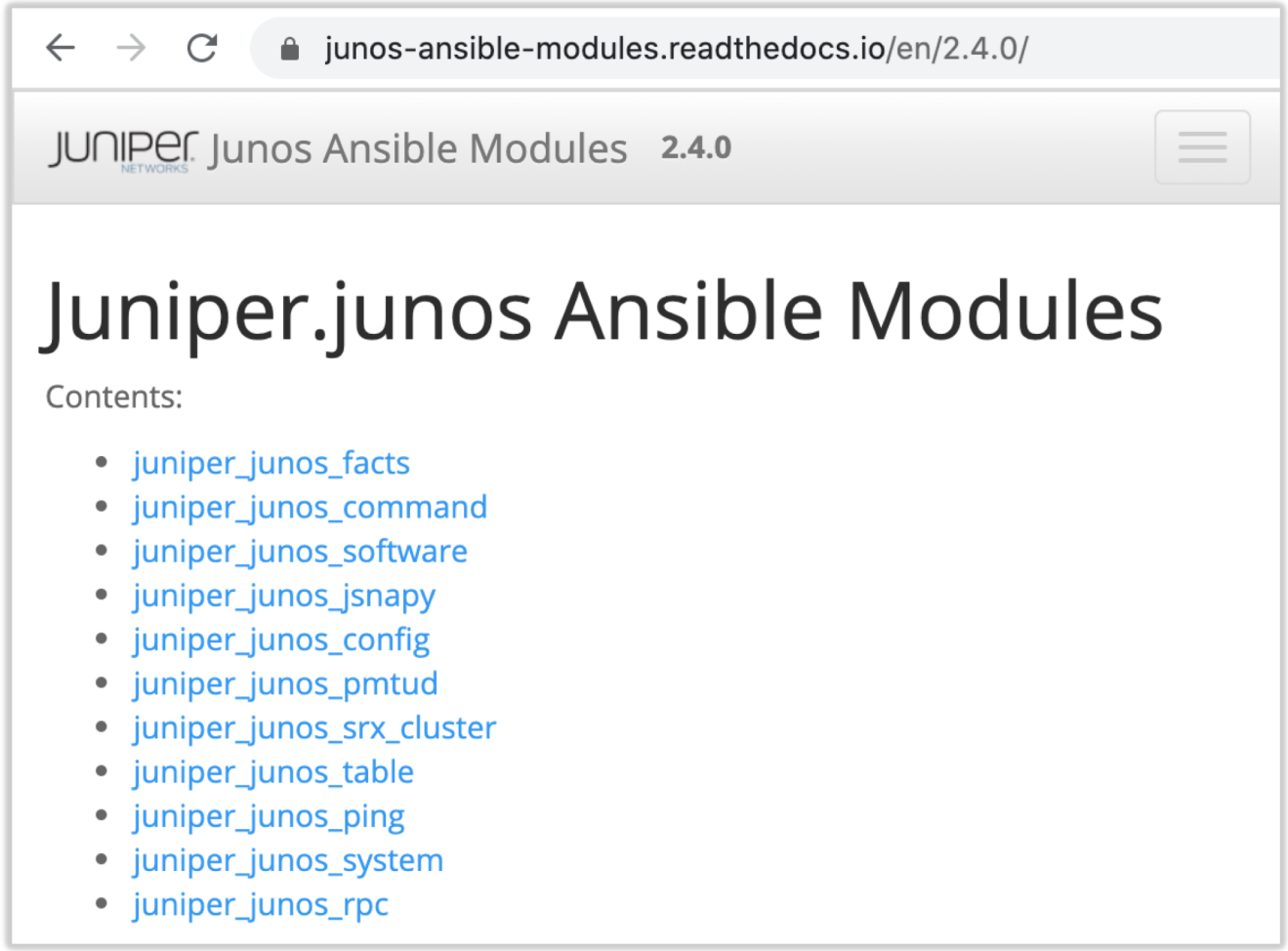
- Everything is active (no failover [no heart attacks])
- Simplicty (single routing engine)

EMnify

Deployment

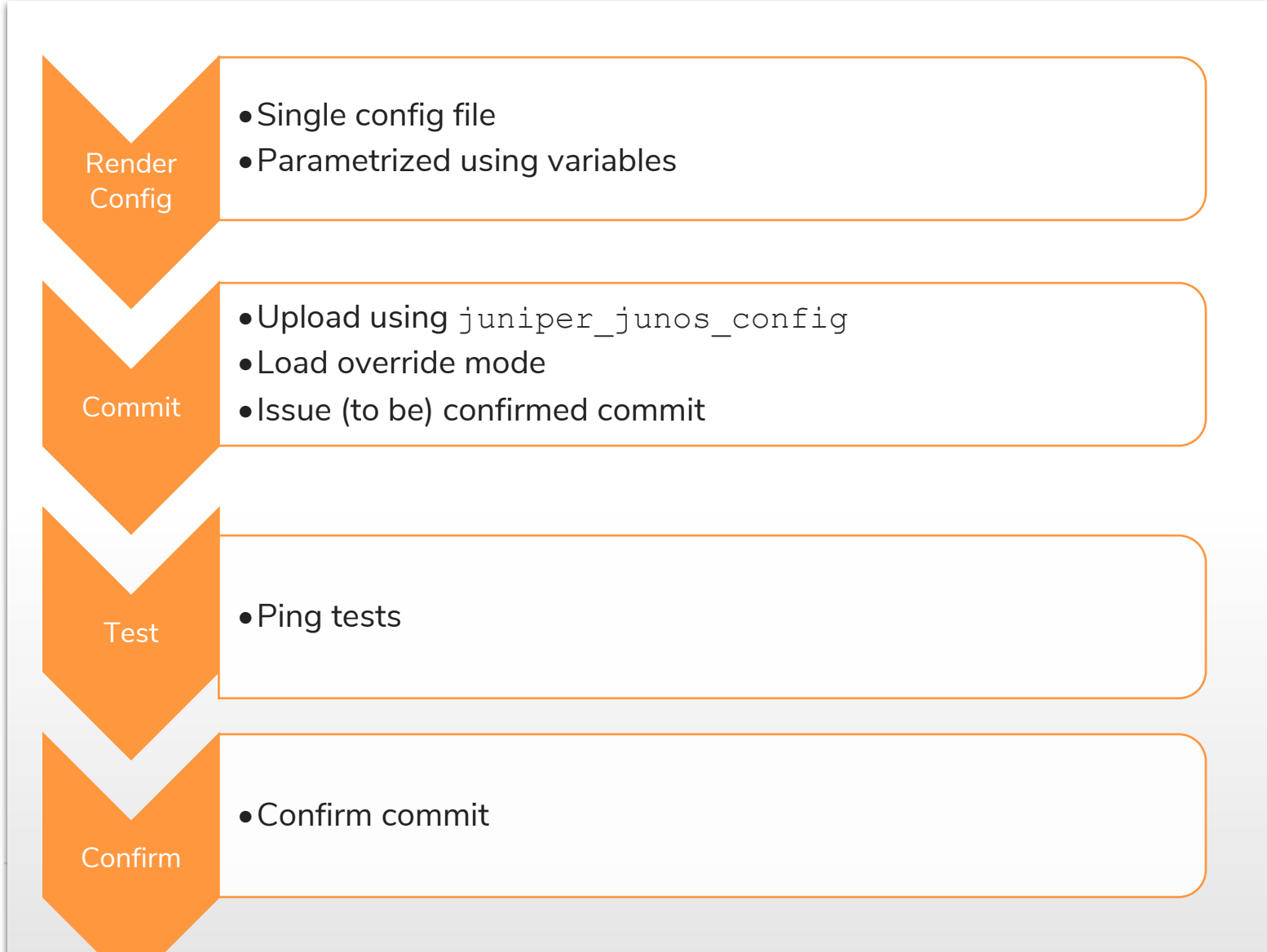
A human shall not SSH into something

juniper_junos Ansible Modules



ANSIBLE

Configuration Deployment



Code Example

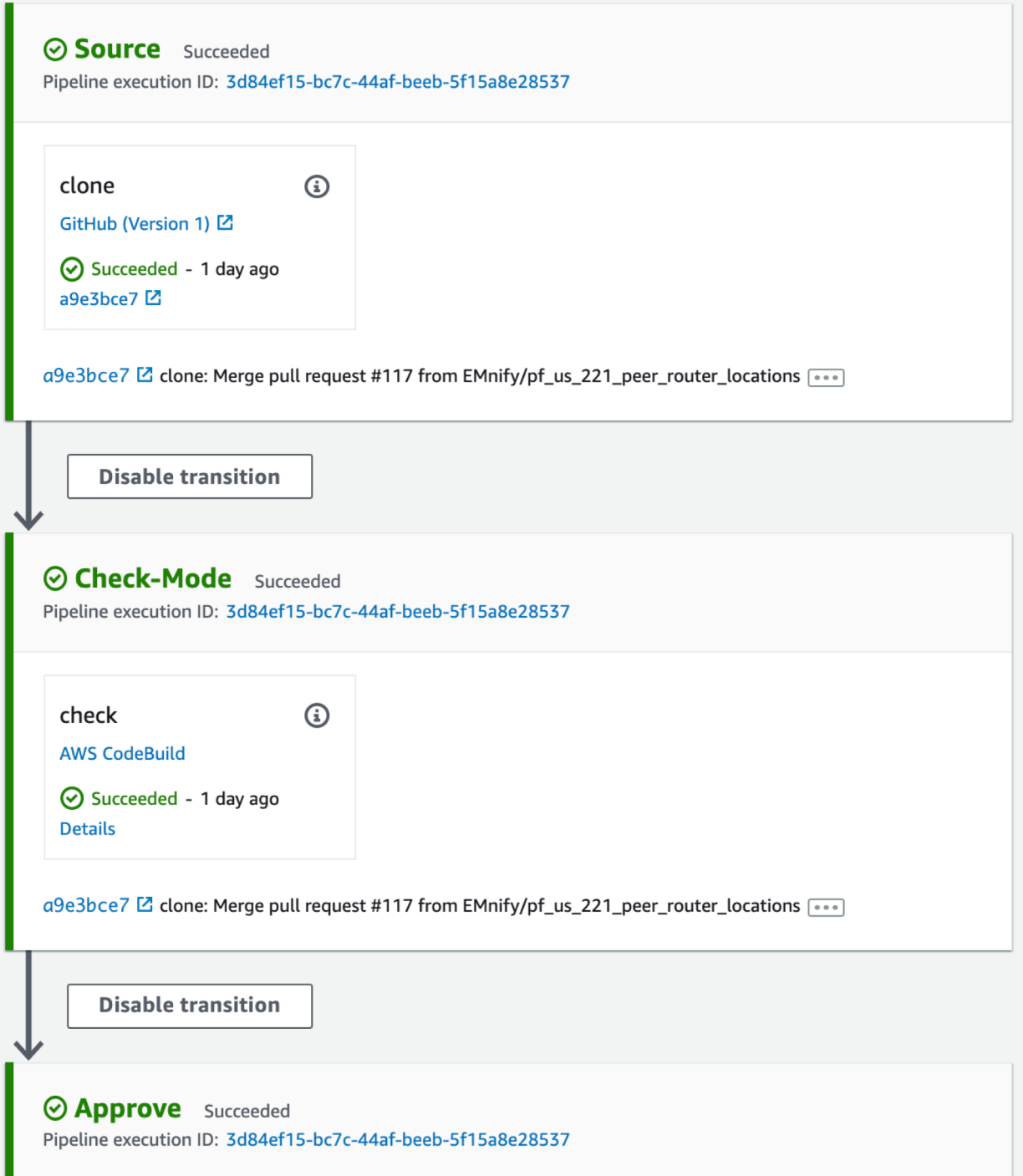
```
- name: install generated configuration file onto device
  juniper_junos_config:
    provider: "{{ juniper_connection_settings }}"
    src: "{{ conf_file }}"
    load: override
    comment: "playbook execution, commit confirmed"
    confirmed: 3 # wait X minutes until rollback
    diff: yes
    ignore_warning: yes
  register: config_results
  notify: confirm previous commit
```

Config Pipeline

- Separate AWS account
- Connectivit

AWS CodePipeline

AWS CodeBuild

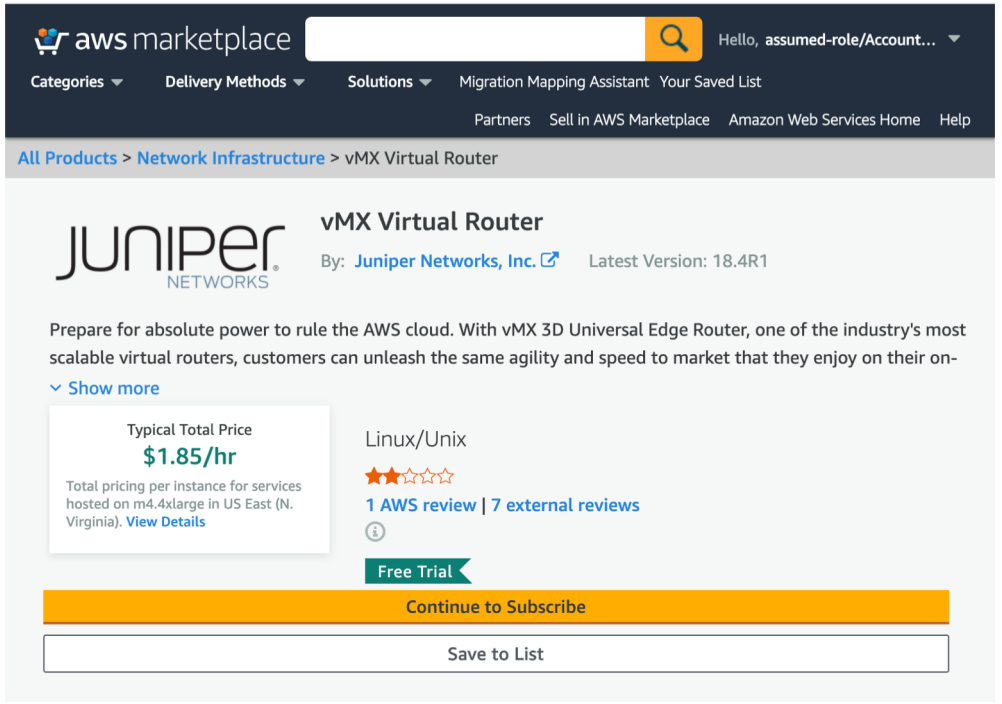


And where are the tests?

In a Perfect World..

- 2-star review could have been mine :D
- Latest version: 18.4R1
- Takes ~30min to be ready
- AWS does not support VLANs!

- Only for manual testing



09/
11/

I On My Bucket List

- Launch an AWS bare metal host (b/c vMX)
- Start virtualized topology in network emulator
- Apply configuration pipeline
- Emulate BGP peers
- Execute end-to-end connectivity tests
- Emulate link failures
- Verify connectivity

11/
9/2

Routine Operations (Runbooks)

Firmware Update - Checks

```
/workspace/prod # ansible-playbook upgrade_check.yaml -u steffen.gebert
...
TASK [Validate result] *****************
[ mx204-am3 ] Chassis Alarms
---------------------------
Expect:
No alarms currently active
Actual:
No alarms currently active
...
[ mx204-am3 ] Core Dumps
---------------------------
Expect:
/var/crash/*core*: No such file or directory
Actual:
/var/crash/*core*: No such file or directory


[ mx204-am3 ] ⚠ Proceed? ⚠
:
Press 'C' to continue the play or 'A' to abort
```

Firmware Update - Draining

```
- name: Drain traffic
  juniper_junos_config:
    provider: "{{ juniper_connection_settings }}"
    load: 'set'
    lines:
      - 'activate policy-options policy-statement OUT-OF-SERVICE-SWITCH term as-path-p
    comment: 'Drain traffic to router for upgrade'


- name: Traffic drained
  pause:
    prompt: |
      [ {{item}} ] Traffic is draining.
      Verify that traffic is completely drained on the following dashboard before proc
      [ {{item}} ] ⚠ Proceed with the JunOS upgrade ⚠?
  loop: "{{ ansible_play_hosts }}"
```

Firmware Update – Execute!

```
- name: Install Junos OS package
  juniper_junos_software:
    provider:
      host: "{{ ansible_host }}"
      timeout: 3600
    remote_package: "{{ junos_vm_file }}"
    validate: True
    cleanfs: False
    vmhost: True
    reboot: True
  ignore_errors: yes # rpc times out when upgrading, despite the provider timeout sett
  register: output
```

Challenges

Deploy a file to the router (!!!)

Max length of `file copy` URLs

Feedback for invalid confg

Lots of boilerplate code

EMnify

Monitoring

Logs

Logs

- Syslog to DNS host name
- Maintained by AWS CloudMap
- Pointing to fluntbit container, running in Fargate
- Forwarding everything into AWS CloudWatch Logs
- Querying in Grafana via AWS CloudWatch Log insights
- fields @timestamp, @message | filter ident = "sshd"

Flow Records

Flow Logs

- IPFIX + BMP
- Pmacct
- Stdout via AWS firelens
- Fluentbit
- AWS CloudWatch Logs
- Grafana

Metrics

- Hardware information (temperature, light levels, etc.)
- Interface counters (throughput, errors, etc.)
- BGP information (received/accepted/installed prefixes)

Setup

- Jtimon

  nileshsimaria/jtimon

  Junos Telemetry Interface (JTI)

  Prometheus exporter

  Prometheus

  Requires PKI

  JTImon – copy of template file for every router, ENUMs

  JTI still lacks many sensors, even in 20.

Why not SNMP

- We store around 400k time series (sampled every 15 seconds)
- High cardinality

EMnify

Alerting

Conclusion

- Please contact me, if you want details (configs etc)