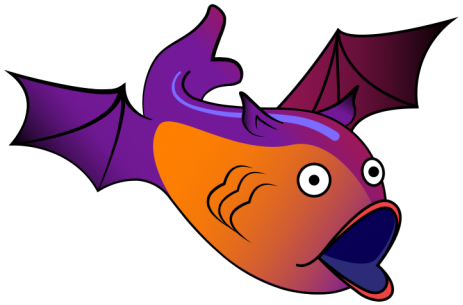
Batfish and Pybatfish: Using open source tools to validate network configuration



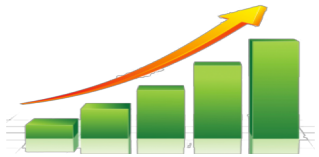
Dan Halperin

INTENTIONET

NANOG 75

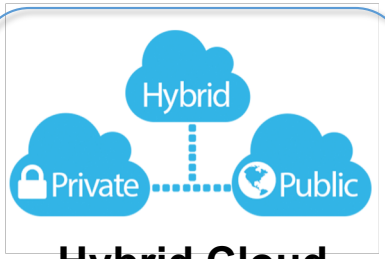
@dhalperi

Engineers managing increasingly diverse complex networks and tasks



Scale

Heterogeneity



Hybrid Cloud

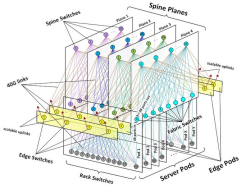
servicenow

Jira Service Desk

Tickets

SOX PCI

Audits



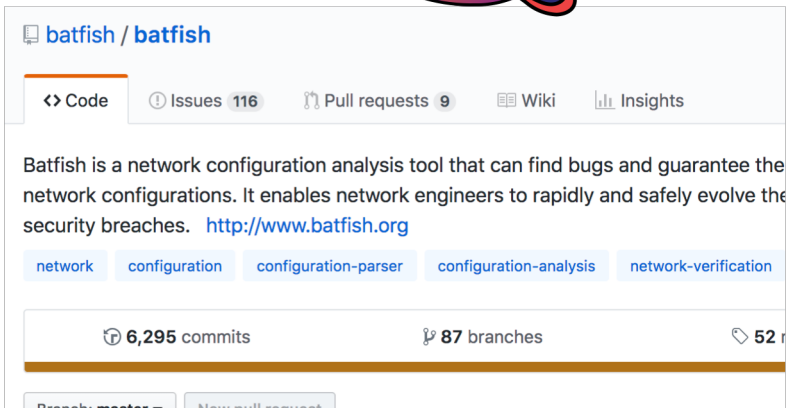
New Builds

Existing tools don't suffice

- **Monitoring**: find a small class of specific bugs after they're already in production. *Incomplete* and *reactive*.
- **Operational state validation**: collect and verify runtime router state (e.g., with Ansible). *Incomplete* and *reactive*.
- **Emulation**: verify partial impact of change on a partial subset of the network. Manual, resource-intensive, *incomplete* – but ***proactive***!

We need better tools:
Proactive and Comprehensive!

Batfish: tools for proactive, comprehensive, & automated network analysis



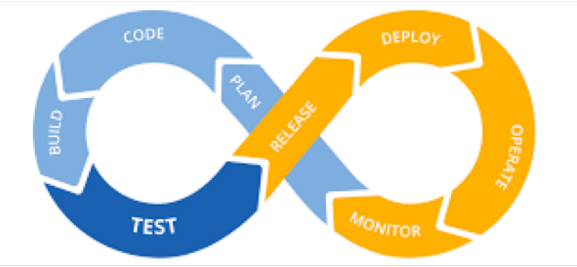
- Used in production by multiple Fortune 500 companies
- Open source since 2014 under Apache 2.0 License
- Contributors from Intentionet, BBN, Colgate University, Microsoft, Princeton, UCLA, USC, and others

What can Batfish do?

- Audit Configuration Settings
  - Are all devices compliant with site standards?
  - Are all cross-device sessions (BGP, IPsec, MLAG, …) compatible?
- Data Plane Analysis
  - How does traffic flow from A to B, and why?
  - Will all tunnels come up?
- Comprehensive Reachability Analysis
  - Can ANY flows violate cross-site isolation?
  - Can ALL clients reach the DNS server?
- Comprehensive ACL/Firewall Analysis
  - Is every ACL rule in effect?
  - What flows will traverse the firewall?

How can you use Batfish proactively?

- Build a CI/CD pipeline
  - Proactive / pre-deployment validation
  - Continuous / post-deployment validation



- Test specific network scenarios
  - Test DR (Disaster Recovery) plan
  - Test network maintenance MOP

How Batfish works: Input

1. All Batfish needs is **raw network configuration**

Batfish can use more data (e.g., BGP advertisements, LLDP/CDP data, Link status), but these are not required.

How Batfish works: Parsing & unification

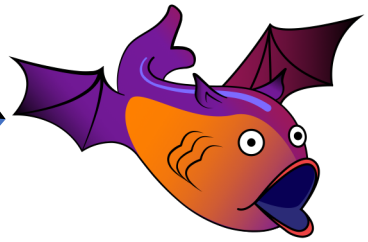
2. **Parsed** into unified **vendor-independent data model**

IOS

template peers
route-maps
access-lists

JunOS

bgp groups
policy-statements
firewall-filters



Batfish unified model

- BGP sessions
- Routing policies
- ACLs

Example: IOS or JunOS BGP config

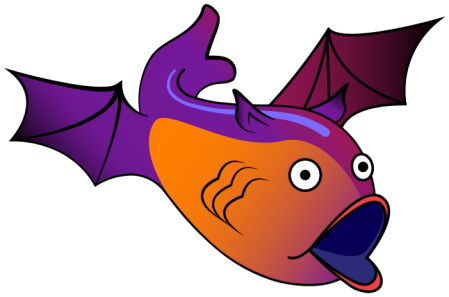
```
router bgp 1
 bgp router-id 1.2.2.2
 neighbor as1 peer-group
 neighbor as1 remote-as 1
 neighbor as3 peer-group
 neighbor as3 remote-as 3
 neighbor as4 peer-group
 neighbor as4 remote-as 4
 neighbor 1.10.1.1 peer-group as1
 neighbor 10.13.22.3 peer-group as3
 neighbor 10.14.22.4 peer-group as4
 neighbor 1.14.22.4 update-source Loopback0
 address-family ipv4
  <snip>
  neighbor as4 route-map as4_to_as1 in
  neighbor as4 route-map as1_to_as4 out
  neighbor 1.10.1.1 activate
  neighbor 10.13.22.3 activate
  neighbor 10.14.22.4 activate
 exit-address-family
```

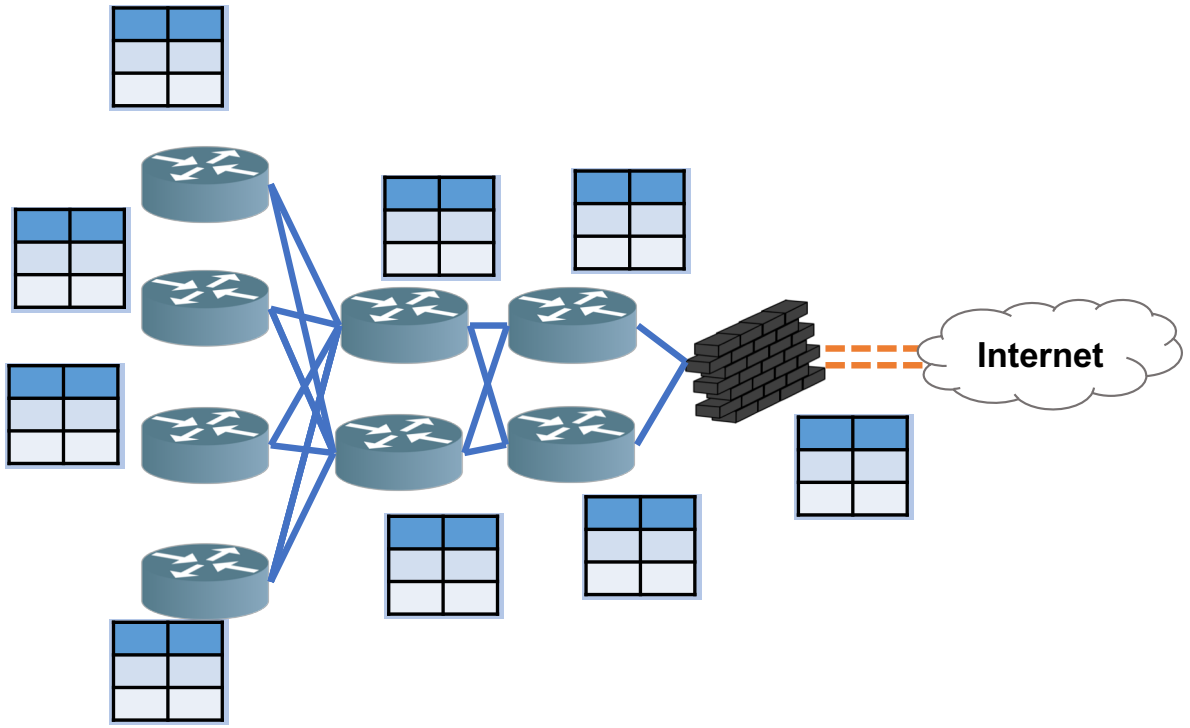
```
set protocols bgp local-as 1
set protocols bgp group as1 type internal
set protocols bgp group as1 local-address 1.2.2.2
set protocols bgp group as1 peer-as 1
set protocols bgp group as1 export as1_to_as1
set protocols bgp group as1 neighbor 1.10.1.1
set protocols bgp group as3 type external
set protocols bgp group as3 peer-as 3
set protocols bgp group as3 export as1_to_as3
set protocols bgp group as3 import as3_to_as1
set protocols bgp group as3 neighbor 10.13.22.3
set protocols bgp group as4 type external
set protocols bgp group as4 peer-as 4
set protocols bgp group as4 export as1_to_as4
set protocols bgp group as4 import as4_to_as1
set protocols bgp group as4 neighbor 10.14.22.4
<snip>
```

Batfish vendor-independent model

How Batfish works: Routing simulation

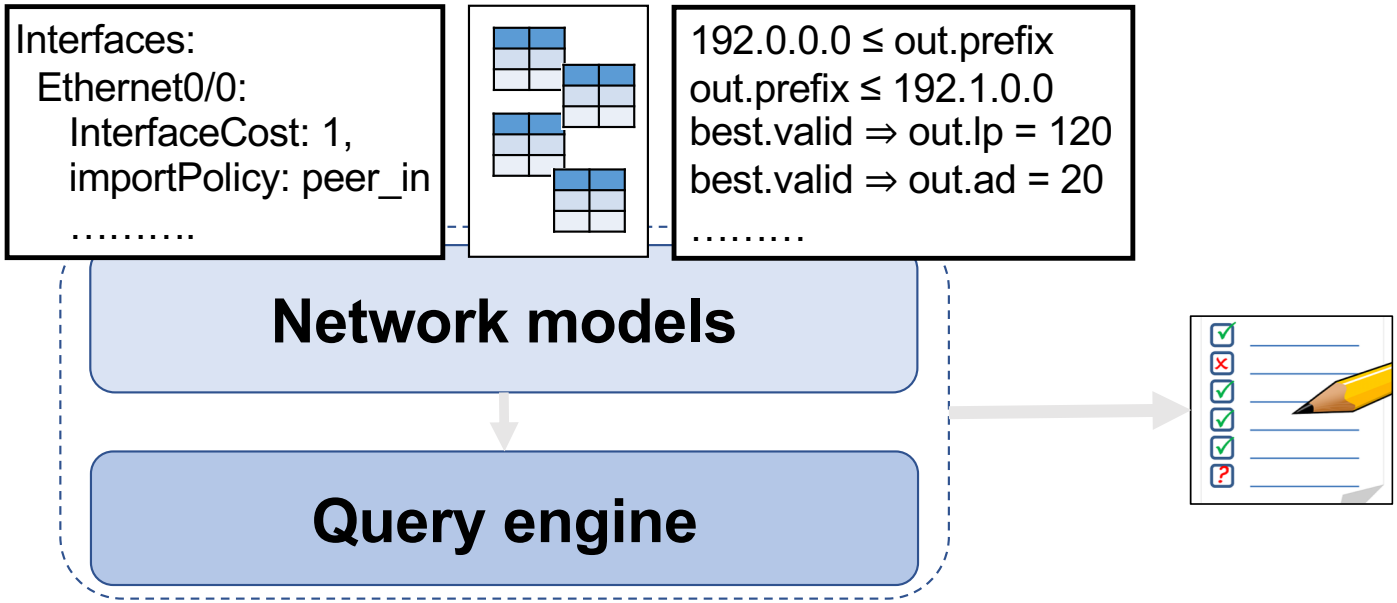
3. **Routing simulation**
and computation of FIBs,
RIBs for ALL devices





How Batfish works: Comprehensive mathematical guarantees

4. **Network verification and query engine** reasons about *all possible* sources, destinations, flows, failures, routing advertisements, and more.



Batfish finds real bugs

- **Automatically detected** bugs found in real networks
  - Misconfigured NTP servers
  - Misconfigured ACLs
  - BGP sessions could never come up because of template errors
  - BGP sessions could never come up because of ACLs
  - ECMP treated flows differently on different paths

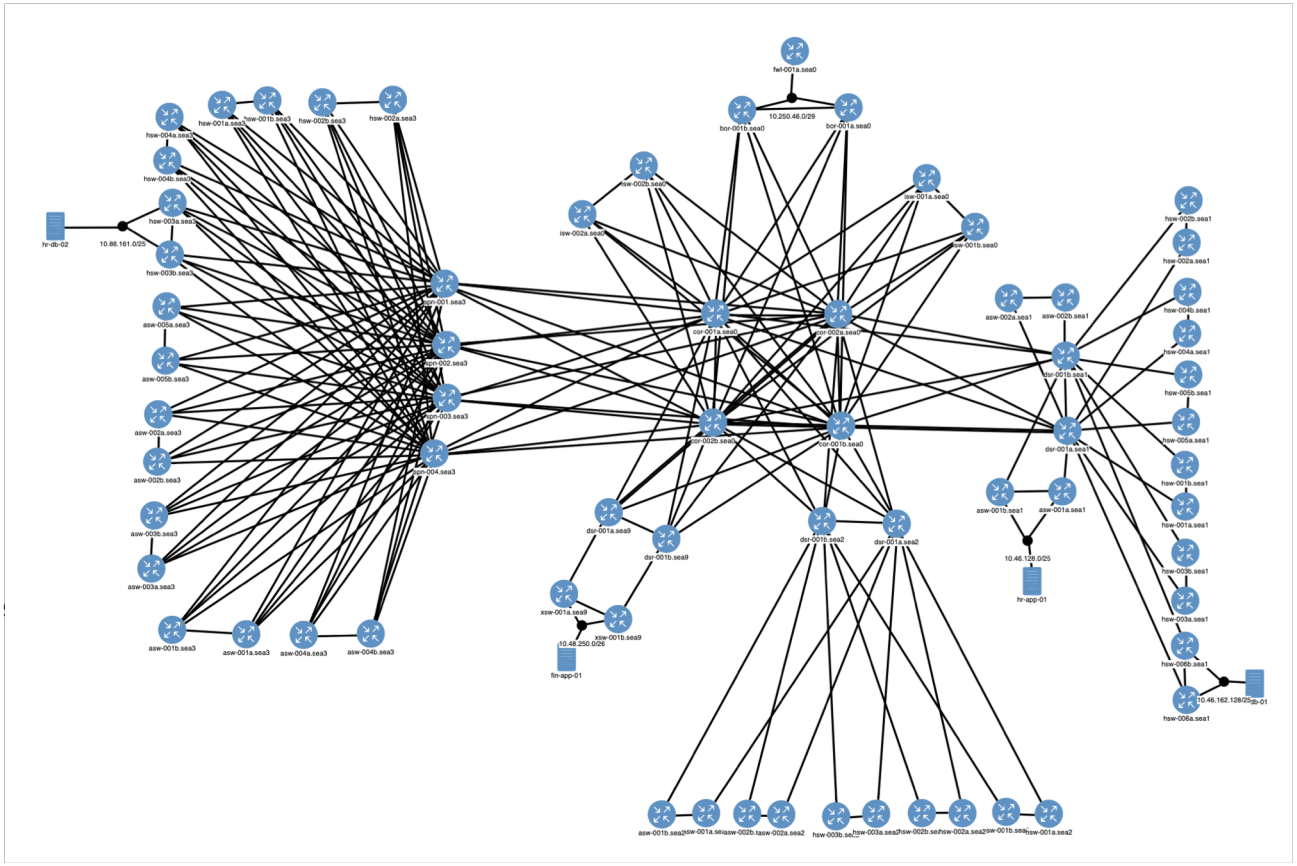
Inconsistency, increased latency, reduced capacity, network fragility, security holes

Demo #1

Batfish in Action

Datacenter network

- 67-node DC with firewall, border routers, distribution layer, and 2 pods (left and right)
- 4 Linux servers in different parts of the network
- BGP and OSPF
- Juniper and Cisco devices, server protection using iptables



What can Batfish do?

- Audit Configuration Settings
  - Are all devices compliant with site standards?
  - Are all cross-device sessions (BGP, IPsec, MLAG, …) compatible?
- Data Plane Analysis
  - How does traffic flow from A to B, and why?
  - Will all tunnels come up?
- Comprehensive Reachability Analysis
  - Can ANY flows violate cross-site isolation?
  - Can ALL clients reach the DNS server?
- Comprehensive ACL/Firewall Analysis
  - Is every ACL rule in effect?
  - What flows will traverse the firewall?

jupyter NANOG75 Last Checkpoint: an hour ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In [ ]:
```
%run /networks/nanog75/startup.py
```

Input - raw configs

In [ ]:
```
!ls /networks/nanog75/configs/
```

In [ ]:
```
!head /networks/nanog75/configs/tor-002a.sea2.test.com
```

In [ ]:
```
!head /networks/nanog75/configs/tor-002a.sea1.test.com
```

In [ ]:
```
bf_set_network('NANOG-75')
snapshot = bf_init_snapshot('/networks/nanog75')
```

Initialization report

In [ ]:
```
bfq.initIssues().answer().frame()
```

Structured, vendor neutral data model of configurations

In [ ]:
```
node_props = bfq.nodeProperties().answer().frame()
node_props.head(5)
```

What can Batfish do?

- Audit Configuration Settings
  - Are all devices compliant with site standards?
  - Are all cross-device sessions (BGP, IPsec, MLAG, …) compatible?
- Data Plane Analysis
  - How does traffic flow from A to B, and why?
  - Will all tunnels come up?
- Comprehensive Reachability Analysis
  - Can ANY flows violate cross-site isolation?
  - Can ALL clients reach the DNS server?
- Comprehensive ACL/Firewall Analysis
  - Is every ACL rule in effect?
  - What flows will traverse the firewall?

jupyter NANOG75 Last Checkpoint: an hour ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Routing simulation and traceroutes ¶

In [ ]:
```
start_node = 'tor-001a.sea3'
destined_outside = HeaderConstraints(dstIps='8.8.8.8')

tr_out = bfq.traceroute(startLocation=start_node, headers=destined_outside, maxTraces=3).answer
show(tr_out.head(1))
```

In [ ]:
```
tr_in = bfq.traceroute(
    startLocation='enter(fwl-001a.sea0[reth0.6])',
    headers=HeaderConstraints(dstIps='10.88.16.7')).answer().frame()
show(tr_in.head(1))
```

Comprehensive reachability checks

In [ ]:
```
anything_coming_in = bfq.reachability(
    pathConstraints=PathConstraints(startLocation='enter(fwl-001a.sea0[reth0.6])'),
    headers=HeaderConstraints(dstIps=f'ofLocation(tor-001a.sea3)'),
    maxTraces=2).answer().frame()
show(anything_coming_in)
```

ACL/Filter Analysis

What can Batfish do?

- Audit Configuration Settings
  - Are all devices compliant with site standards?
  - Are all cross-device sessions (BGP, IPsec, MLAG, …) compatible?
- Data Plane Analysis
  - How does traffic flow from A to B, and why?
  - Will all tunnels come up?
- Comprehensive Reachability Analysis
  - Can ANY flows violate cross-site isolation?
  - Can ALL clients reach the DNS server?
- Comprehensive ACL/Firewall Analysis
  - Is every ACL rule in effect?
  - What flows will traverse the firewall?

jupyter NANOG75 Last Checkpoint: an hour ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Comprehensive reachability checks

In [ ]:
```
anything_coming_in = bfq.reachability(
    pathConstraints=PathConstraints(startLocation='enter(fwl-001a.sea0[reth0.6])'),
    headers=HeaderConstraints(dstIps=f'ofLocation(tor-001a.sea3)'),
    maxTraces=2).answer().frame()
show(anything_coming_in)
```

ACL/Filter Analysis

In [ ]:
```
explanation = bfq.testFilters(
    filters='~SECURITY_POLICIES_TO~reth0.5',
    nodes='fwl-001a.sea0',
    headers=HeaderConstraints(srcIps='135.97.2.89', dstIps='10.88.16.7', ipProtocols='ICMP'),
    startLocation='enter([reth0.6])').answer().frame()
print(explanation.Trace[0])
```

In [ ]:
```
bfq.filterLineReachability(ignoreComposites=True).answer()
```

In [ ]:
```
!grep 'BLOCK-RFC1918' /networks/nanog75/configs/bor-001b.sea0.test.com
```

In [ ]:
```
!grep 'ALLOW-ALL-ICMP-FROM-SEA' /networks/nanog75/configs/bor-001b.sea0.test.com
```

In [ ]:
```
!grep 'policy-options prefix-list SEA-PUBLIC-NETS' /networks/nanog75/configs/bor-001b.sea0.test
```

What can Batfish do?

- Audit Configuration Settings
  - Are all devices compliant with site standards?
  - Are all cross-device sessions (BGP, IPsec, MLAG, …) compatible?
- Data Plane Analysis
  - How does traffic flow from A to B, and why?
  - Will all tunnels come up?
- Comprehensive Reachability Analysis
  - Can ANY flows violate cross-site isolation?
  - Can ALL clients reach the DNS server?
- Comprehensive ACL/Firewall Analysis
  - Is every ACL rule in effect?
  - What flows will traverse the firewall?

jupyter NANOG75 Last Checkpoint: an hour ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

ACL/Filter Analysis

In [ ]:
```
explanation = bfq.testFilters(
    filters='~SECURITY_POLICIES_TO~reth0.5',
    nodes='fw1-001a.sea0',
    headers=HeaderConstraints(srcIps='135.97.2.89', dstIps='10.88.16.7', ipProtocols='ICMP'),
    startLocation='enter([reth0.6])').answer().frame()
print(explanation.Trace[0])
```

In [ ]:
```
bfq.filterLineReachability(ignoreComposites=True).answer()
```

In [ ]:
```
!grep 'BLOCK-RFC1918' /networks/nanog75/configs/bor-001b.sea0.test.com
```

In [ ]:
```
!grep 'ALLOW-ALL-ICMP-FROM-SEA' /networks/nanog75/configs/bor-001b.sea0.test.com
```

In [ ]:
```
!grep 'policy-options prefix-list SEA-PUBLIC-NETS' /networks/nanog75/configs/bor-001b.sea0.test
```

"What if" ... one of my border routers went down?

In [ ]:
```
failed_bor_1a = bf_fork_snapshot(base_name=snapshot, deactivate_nodes=['bor-001a.sea0'])
```

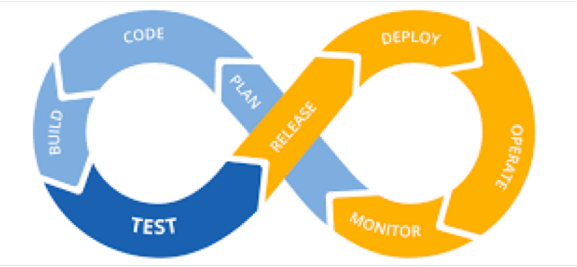
In [ ]:
```
starting_sw3a = PathConstraints(startLocation=start_node)
destined_outside = HeaderConstraints(dstIps="8.8.8.8")

diff = bfq.differentialReachability(headers=destined_outside, pathConstraints=starting_sw3a).an
```

Demo #2

Batfish use cases

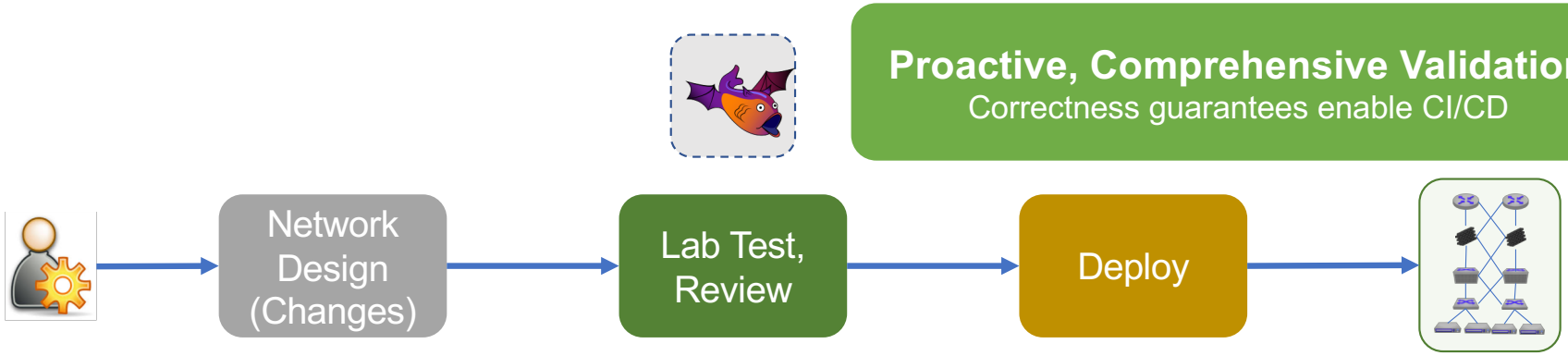
How can you use Batfish proactively?

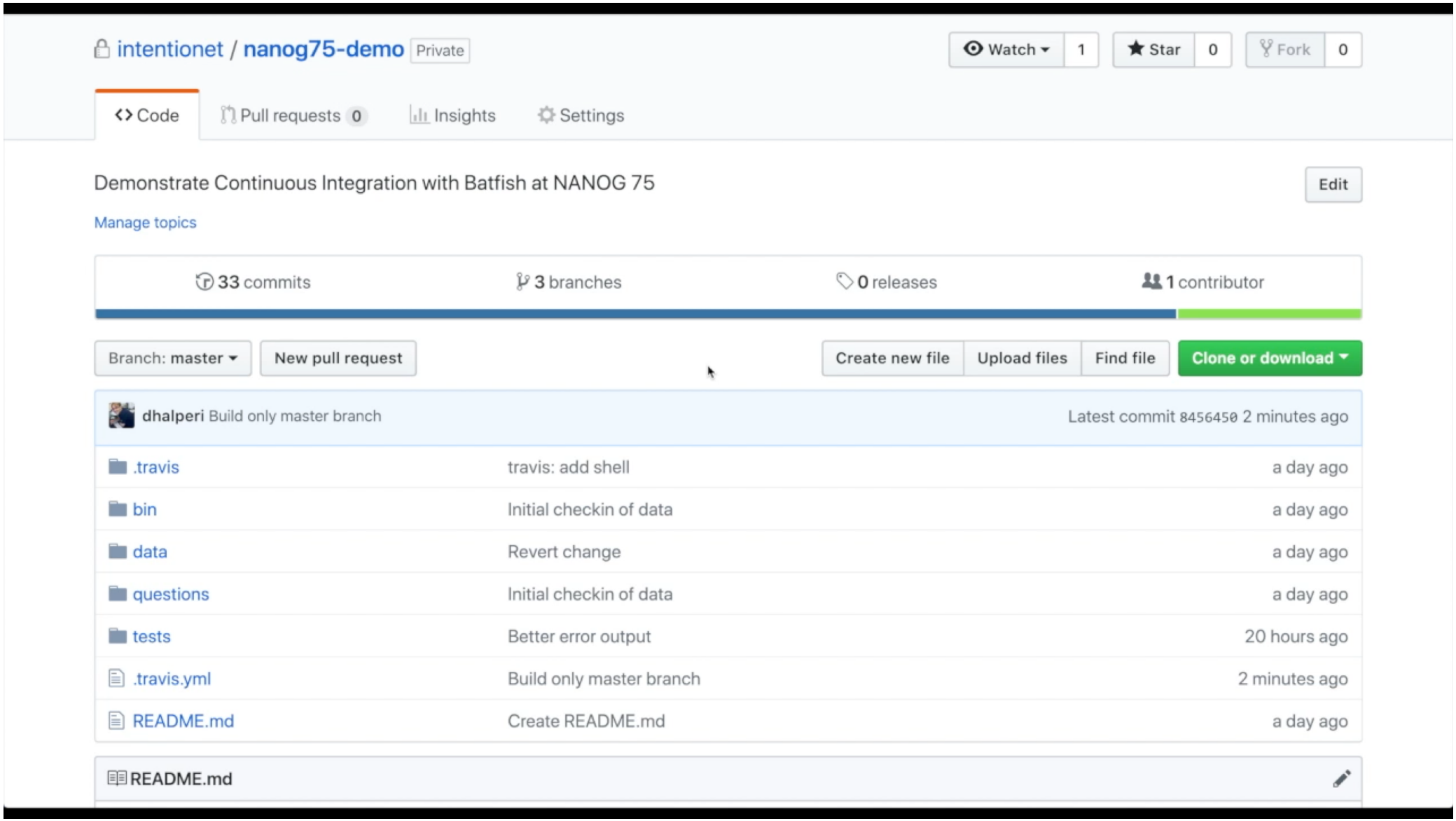
- Build a CI/CD pipeline
  - Proactive / pre-deployment validation
  - Continuous / post-deployment validation



- Test specific network scenarios
  - Test DR (Disaster Recovery) plan
  - Test network maintenance MOP

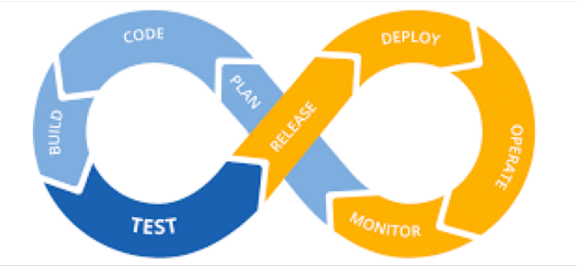
Batfish + automation = Continuous Integration




intentionet / nanog75-demo Private
Watch 1
Star 0
Fork 0
Code
Pull requests 0
Insights
Settings
Demonstrate Continuous Integration with Batfish at NANOG 75
Edit
Manage topics
33 commits
3 branches
0 releases
1 contributor
Branch: master
New pull request
Create new file
Upload files
Find file
Clone or download
dhalperi Build only master branch
Latest commit 8456450 2 minutes ago
.travis
travis: add shell
a day ago
bin
Initial checkin of data
a day ago
data
Revert change
a day ago
questions
Initial checkin of data
a day ago
tests
Better error output
20 hours ago
.travis.yml
Build only master branch
2 minutes ago
README.md
Create README.md
a day ago
README.md

How can you use Batfish proactively?

- Build a CI/CD pipeline
  - Proactive / pre-deployment validation
  - Continuous / post-deployment validation



- Test specific network scenarios
  - Test DR (Disaster Recovery) plan
  - Test network maintenance MOP

jupyter NANOG75 Last Checkpoint: an hour ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

"What if" ... one of my border routers went down?

In [ ]:
```
failed_bor_1a = bf_fork_snapshot(base_name=snapshot, deactivate_nodes=['bor-001a.sea0'])
```

In [ ]:
```
starting_sw3a = PathConstraints(startLocation=start_node)
destined_outside = HeaderConstraints(dstIps="8.8.8.8")

diff = bfq.differentialReachability(headers=destined_outside, pathConstraints=starting_sw3a).an
    snapshot=failed_bor_1a, reference_snapshot=snapshot).frame()
show(diff)
```

In [ ]:
```
route_diff = bfq.routes().answer(snapshot=failed_bor_1a, reference_snapshot=snapshot).frame()
route_diff.head()
```

In [ ]:
```
route_diff.groupby(['Node']).size()
```

In [ ]:
```
route_diff.groupby(['Reference_Protocol', 'Entry_Presence']).size()
```

Getting started with Batfish is easy

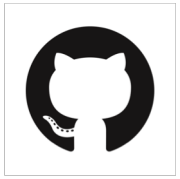
- One line with Docker
  https://github.com/batfish/batfish/#how-do-i-get-started

- Advanced Jupyter tutorials
  - **Getting started** with Batfish
  - **Validating configuration** settings
  - Introduction to **Route analysis**
  - **Analyzing ACLs** and Firewall rules

- **Provably safe** ACL and Firewall **changes**
- Introduction to **Forwarding Analysis**
- Introduction to **Forwarding Change Validation**
- Analyzing the **Impact of Failures**

Batfish: Open source tools for network validation

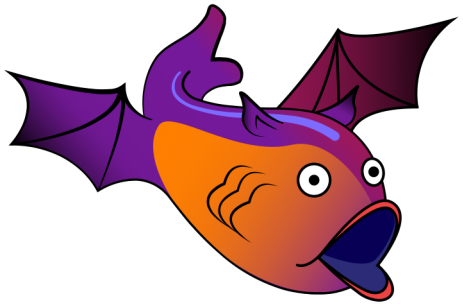
batfish-org



@batfish

batfish.org

Batfish and Pybatfish: Using open source tools to validate network configuration



Dan Halperin

INTENTIONET

NANOG 75

@dhalperi

BACKUP

Additional Batfish tutorials:

Control plane & Data plane analysis

- Routing Analysis (pull RIBs/FIBs. Confirm presence/absence of routes. see notebook on)
- Enhanced traceroute. Shows multipath and shows why paths are taken (aka, which type of route, filters, metrics. With and without ACLs. (see notebook)
- Comprehensive path queries: Reachability to prove *no* packets outside expected set can actually reach device. (see notebook)