Automated network configuration

Big networks can be correct

Michael Shields, Google Engineering
RIPE 57, October 2008

olicy generation for the network

udit for correctness and policy adherence

nsure completeness of your architectural standards

odeling

y equipment

ovision power and space

ck and stack

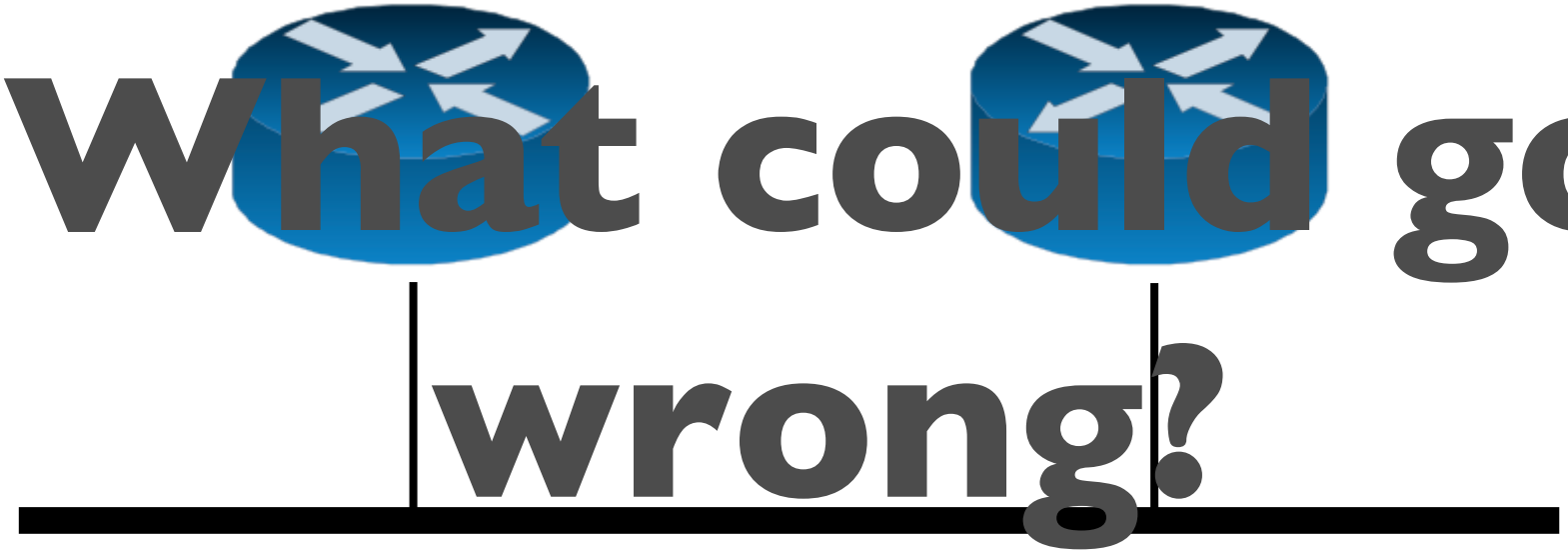
erconnect

onfigure the device based on your documentation

ve another engineer check your work

ofit!

We've done it a thousand times. What could go wrong?



```
interface ethernet [x/y]
 ip address [address] [netmask]
 vrrp 1 priority [120, 100]
 vrrp 1 authentication cisco
 vrrp 1 timers advertise 3
 vrrp 1 timers learn
 vrrp 1 ip [address]
 no shutdown
```

```
interface ethernet 1/0
 ip address 10.1.0.2 255.255.255.0
 vrrp 1 priority 120
 vrrp 1 authentication cisco
 vrrp 1 timers advertise 3
 vrrp 1 timers learn
 vrrp 1 ip 10.1.0.10
 no shutdown
```

```
erface ethernet 1/0
 address 10.1.0.2 255.255.255.0
rp 1 priority 100
rp 1 authentication cisco
rp 1 timers advertise 3
rp 1 timers learn
rp 1 ip 10.1.0.10
 shutdown
```

```
interface ethernet 1/0
 ip address 10.1.0.2 255.255.25
 vrrp 1 priority 120
 vrrp 1 authentication cisco
 vrrp 1 timers advertise 3
 vrrp 1 timers learn
 vrrp 1 ip 10.1.0.10
 no shutdown
```

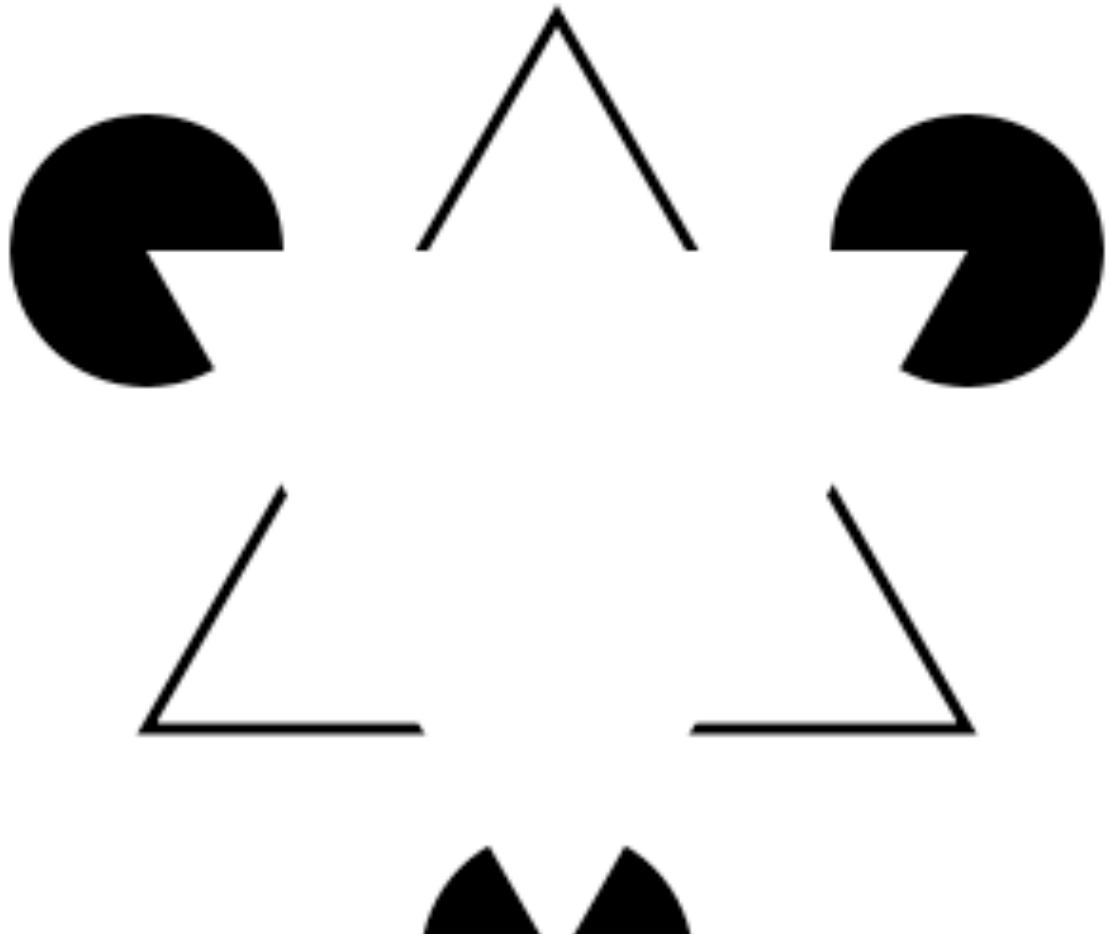
```
erface ethernet 1/0
 address 10.1.0.2 255.255.255.0
rp 1 priority 100
rp 1 authentication cisco
rp 1 timers advertise 3
rp 1 timers learn
rp 1 ip 10.1.0.10
 shutdown
```

```
interface ethernet 1/0
 ip address 10.1.0.2 255.255.25
 vrrp 1 priority 120
 vrrp 1 authentication cisco
 vrrp 1 timers advertise 3
 vrrp 1 timers learn
 vrrp 1 ip 10.1.0.10
 no shutdown
```

eople will cut and paste. Errors propagate.

umans expect to see patterns.
Our brains see things that aren't there.

rors happen, and you won't find all of them just by looking.



ost common types of errors:

- Missing or incorrect security ACLs
- Incomplete BGP meshes (mysterious blackholing)
- Incomplete MPLS mesh
- Incomplete or incorrect QoS configuration
- IP address confusion

pical response: Add more procedures

- End up with a mass of procedures and policies that look the same
- Can’t ever keep the corpus of documentation self-consistent
- Can’t ever keep the network consistent with documentation
- You’ll go blind trying. Need to create abstractions so we can think big.

he more actions you take, the more mistakes you'll make.

it's not automated, it will not scale.

[Corollary: if your network can be managed by hand, it is small.]

orrect networks scale better. This is a competitive advantage

ut you already have a network. It’s up and running.
ou’re good at your job, so your network is pretty good.

s good enough. But how do you get to better?

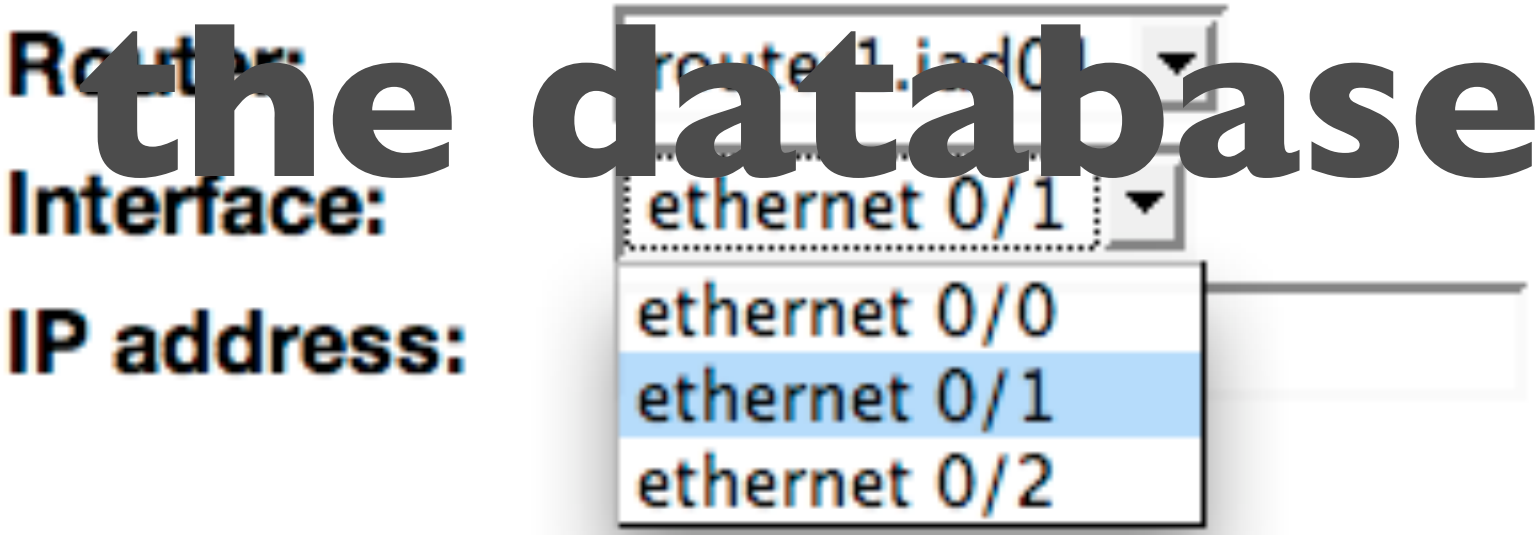
hat should you do instead?
and how do you get there?

onfigurations are templates with variable substitution

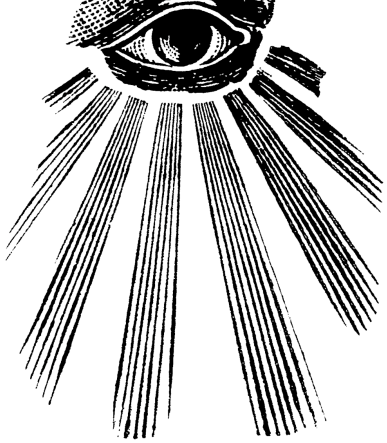
nforce policy by tools, not only by documentation

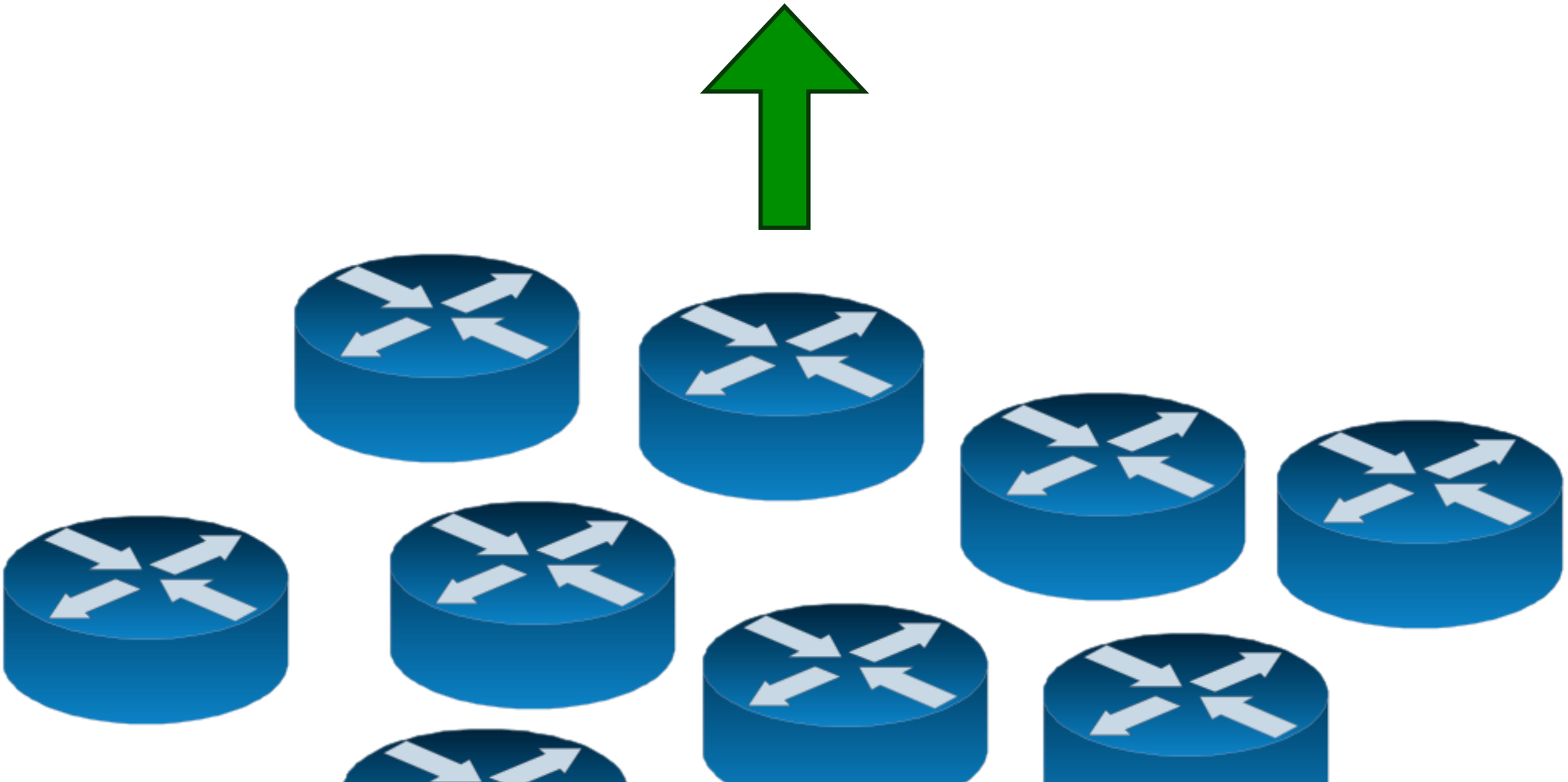
orces you to find your exceptions

Don’t touch the router until it’s in the database

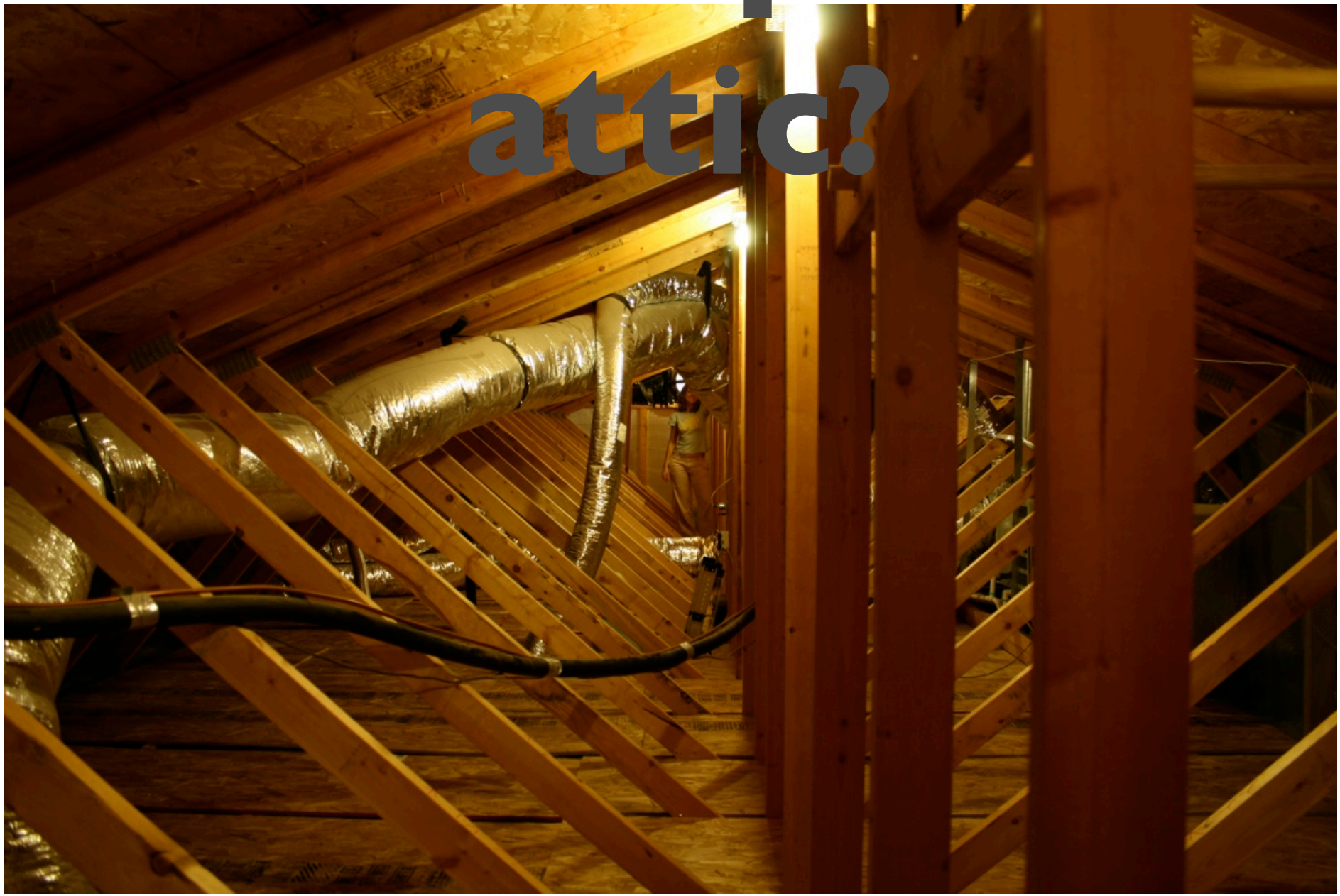


Subnet	Interface	Customer
10.1.0.2/24	ethernet 0/1	6829 — E. Blofeld, Inc.
10.1.0.3/24	ethernet 0/2	3189 — Disco Volante
10.1.0.4/23	ethernet 0/3	17942 — Thanet Alloy





What's up in the attic?



```
interface ethernet 1/0
 ip address 10.1.0.2 255.255.255.0
 vrrp 1 priority 100
 vrrp 1 authentication cisco
 vrrp 1 timers advertise 3
 vrrp 1 timers learn
 vrrp 1 ip 10.1.0.10
 no shutdown
```

```
interface ethernet 1/0
 ip address 10.1.0.1 255.255.255.0
 vrrp 1 priority 100
 vrrp 1 authentication cisco
 vrrp 1 timers advertise 3
 vrrp 1 timers learn
 vrrp 1 ip 10.1.0.10
 no shutdown
```

Why are things wrong?
(in order of how easy to fix)

1. Bugs in code for initial population of database
2. Actual configuration errors
3. Deliberately unusual configuration

s not enough to generate your configs once.

ou should be able to recheck the actual network state against th
enerated network state at any time.

Actual Google audit

```
otocols {
...
ospf {
  ...
  area 0.0.0.0 {
    {% for interface in dev.physicalinterface_set.all %}
    {% for unit in interface.logicalinterface_set.all %}
    {% if unit.ospf_metric %}
    interface {{ unit.name }} {
      metric {{ unit.ospf_metric }};
    }
    {% endif %}
    {% endfor %}
    {% endfor %}
  }
  ...
}
```

Router	Type	Loopback	IS-IS NET
router1.iad01	Cisco AGS+	192.0.2.38	49.0001.0000.0000.000a.00
router2.iad01	Cisco AGS+	192.0.2.39	49.0001.0000.0000.000b.00
router1.lhr07	Cisco 4500M	192.0.2.207	49.0001.0000.0000.000c.00

```
terface serial 1
p address 10.0.0.2 255.0.0.0
p ospf network point-to-multipoint

ncapsulation frame-relay
rame-relay map ip 10.0.0.1 201 broadcast
rame-relay map ip 10.0.0.3 202 broadcast
rame-relay map ip 10.0.0.4 203 broadcast

uter ospf 1
etwork 10.0.0.0 0.0.0.255 area 0
```

```
interface serial 1
 ip address 10.0.0.2 255.0.0.0
 ip ospf network point-to-multipoint
 ip router isis
 isis metric 503 level-2
 isis password ISISPASSWORD level-2
 encapsulation frame-relay
 frame-relay map ip 10.0.0.1 201 broad
 frame-relay map ip 10.0.0.3 202 broad
 frame-relay map ip 10.0.0.4 203 broad
!
router ospf 1
 network 10.0.0.0 0.0.0.255 area 0
!
router isis
 passive-interface serial 1
 maximum-paths 6
 net 49.0001.0000.0000.000a.00
 is-type level-2-only
 metric-style wide
...
```

data) → old configuration

(data) → new configuration

ou can work at the level of the forest, not the trees

Allow for one-offs with jails

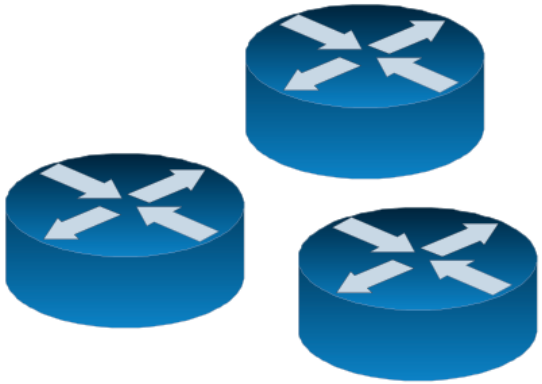
Ad hoc
isolation jail

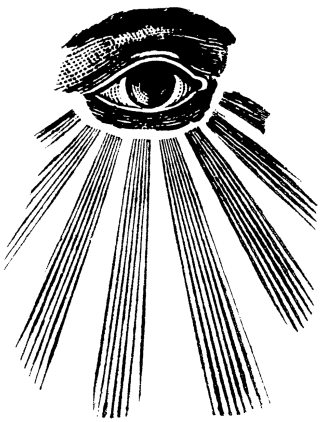
Configuration collector — RANCID is a good start

Database

Configlet generator

Comparison engine





configs
(as-built)

$\stackrel{?}{=}$

configs
(generated)

enter all routine operations around an audit.

x one class of problems at a time, networkwide.
BGP mesh is a good place to start.

ontinuously compare your generated configs
gainst actual configs, and get diffs to zero.

lake jails to isolate nonstandardness.

mall networks don’t need this.
g ones do.
he transition is the hard part.

ou can only get there incrementally.

Out of the crooked timber of humanity,
no straight thing was ever made.” —Kant

Questions?

Michael Shields

mshields@google.com

P.S. We're still hiring.