REANNZ LUNCHTIME SESSION 17 JULY 2019

UNLEASHING THE HIVEMIND

BUILDING SCALABLE NETWORKS

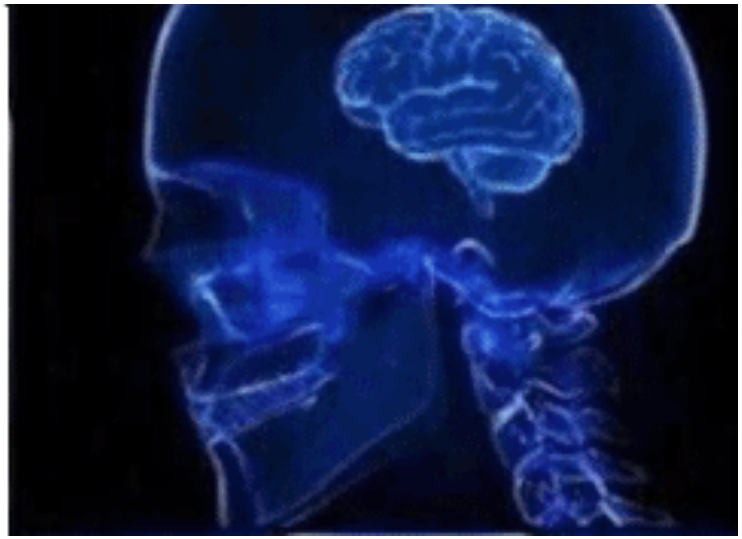
AARON MURRIHY
SENIOR NETWORK ENGINEER
aaron.murrihy@reannz.co.nz

REANNZ

4 STAGES OF NETWORK PROGRESSION

STAGE 1

Configuring switches by hand



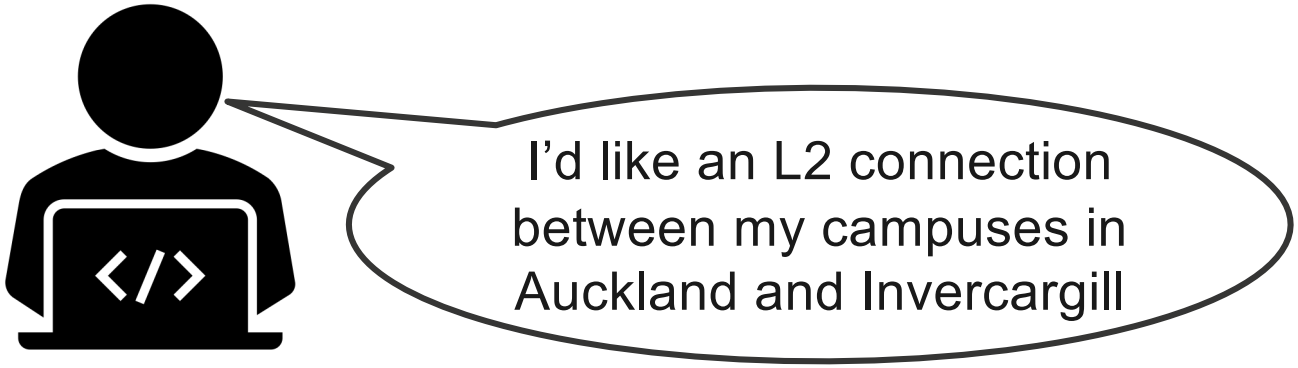
STAGE 2

Using Bash and ClusterSSH to configure multiple hosts at the same time

```
aaron@nms-wlg:~$ for vlan in `grep "set vlans" rnz02.set | grep "vlan-id" | awk '{print $5}'`; do echo "$vlan is on the
following interfaces"; grep "set interfaces" rnz02.set | grep "vlan members" | grep $vlan | awk '{print $3}' ; done
```

STAGE 3

Provide the intended outcome and have the network configure itself



STAGE 4

Have the user provide the intent and take a long lunch

THE FOUR STAGES

What do we need to get from here to here?!

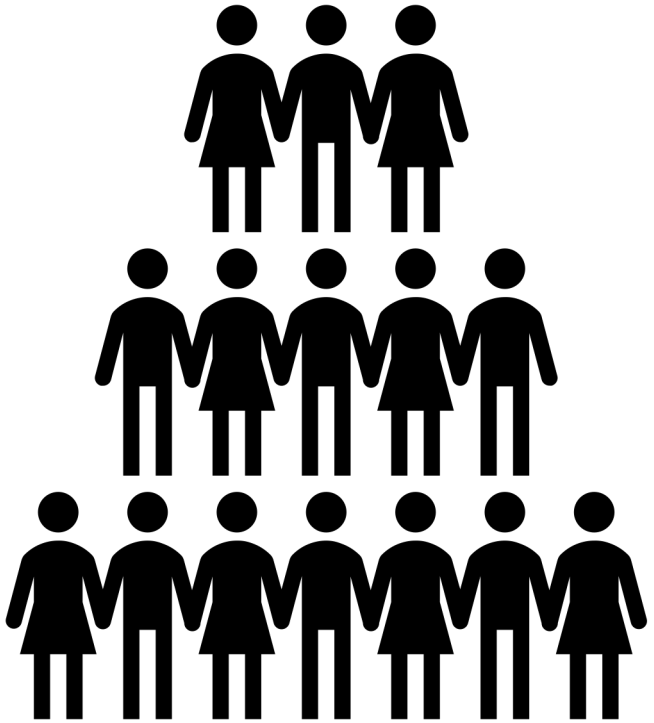
1. Configuring switches by hand
2. Using Bash and ClusterSSH to configure multiple hosts at the same time
3. Provide the intended outcome and have the network configure itself
4. Have the user provide the intent and take a long lunch

HIVEMIND!

- Get everyone on the same page
- With the same goals
- Empowered to drive improvement
- Sharing information

- How?

- Software engineers have this sorted. Learn from them!



TREAT NETWORK CONFIG LIKE CODE

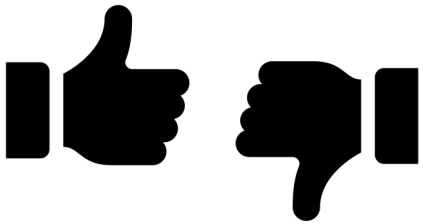
TREATING CONFIG LIKE CODE

1. Documentation
2. Revision control
3. Peer review
4. Standardised templating
5. Testing

CONFIG AS CODE

DOCUMENTATION

- Assumptions
- Architectures
- Tooling
- Process
- Future thinking



CONFIG AS CODE

REVISION CONTROL

- Replication of hosts
- Network config diffs
- Changelogs

PEER REVIEW

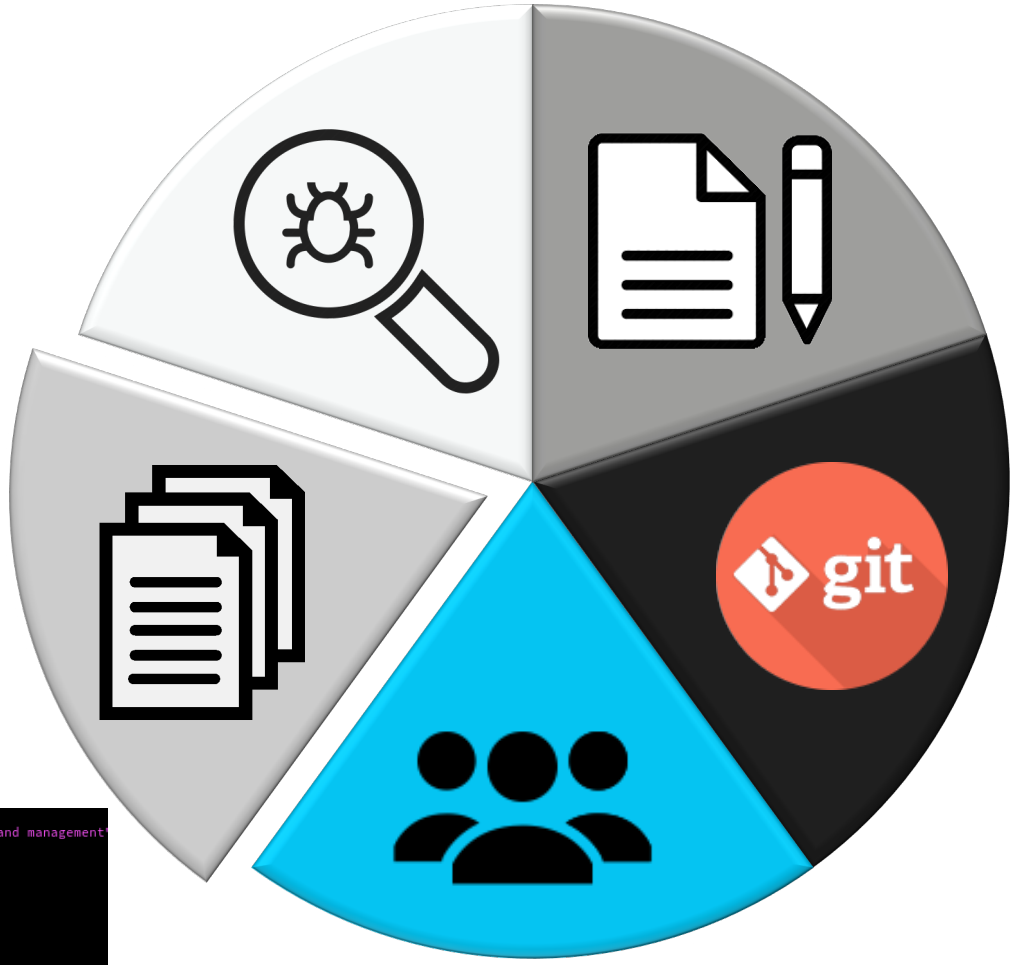
- Documentation
- Bespoke architectures
- Config Diffs

```
-       unit 2851 {
-          description "rnz01 management";
-          family inet {
-             mtu 1500;
-             address 172.24.149.249/31;
-          }
-       }
        unit 3013 {
           description "BFR rnz-staff to and05";
           family inet {
@@ -994,10 +987,7 @@
              filter {
                 input protect-re;
              }
-             address 172.24.133.1/32 {
-                primary;
-             }
-             address 172.24.149.1/32;
+             address 172.24.133.1/32;
          }
          family inet6 {
             filter {
```

CONFIG AS CODE

STANDARDISED TEMPLATING

- Based on documentation
- Host and service templates
- Minimal input requirements
- Most value, least work




```
AND02_CONFIG_TEMPLATE = """
set interfaces {{ AND02_FW_PORT }} unit {{ MGMT_WAN_VLAN }} description "{% filter upper %}{{ MEMBER_CODE }}{% endfilter %} inband management"
set interfaces {{ AND02_FW_PORT }} unit {{ MGMT_WAN_VLAN }} vlan-id {{ MGMT_WAN_VLAN }}
set interfaces {{ AND02_FW_PORT }} unit {{ MGMT_WAN_VLAN }} family inet mtu 1500
set interfaces {{ AND02_FW_PORT }} unit {{ MGMT_WAN_VLAN }} family inet address {{ MGMT_WAN_WLG_FW_AND02_ADDR }}
set routing-instances {{ MEMBER_CODE }}-mgmt instance-type vrf
set routing-instances {{ MEMBER_CODE }}-mgmt interface {{ AND02_FW_PORT }}.{{ MGMT_WAN_VLAN }}
set routing-instances {{ MEMBER_CODE }}-mgmt route-distinguisher 38022:{{ ROUTE_DISTINGUISHER }}
set routing-instances {{ MEMBER_CODE }}-mgmt vrf-target target:38022:{{ ROUTE_DISTINGUISHER }}
set routing-instances {{ MEMBER_CODE }}-mgmt vrf-table-label
set routing-instances {{ MEMBER_CODE }}-mgmt routing-options static route 172.24.64.0/24 next-hop {{ MGMT_WAN_WLG_FW_AAA_ADDR_NO_MASK }}
set routing-instances {{ MEMBER_CODE }}-mgmt routing-options auto-export
"""
```

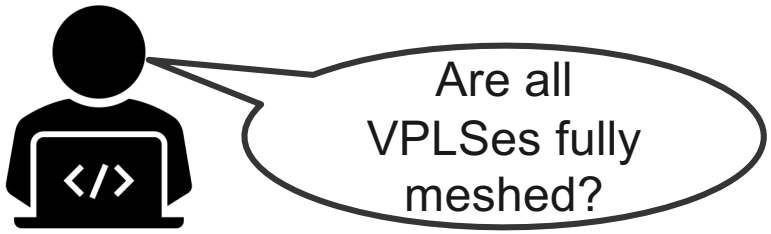
CONFIG AS CODE

TESTING

TESTING

- Monitoring
- Network config unit tests!

```
aaron@and02-wlg-re0# show routing-instances mae-mgmt
instance-type vrf;
interface xe-1/2/1.3333;
interface xe-1/2/1.3334;
interface xe-1/2/1.3336;
interface ge-2/0/0.3255;
interface irb.123; ## 'irb.123' is not defined
route-distinguisher 38022:15018;
vrf-target target:38022:15018;
```



FINALLY

NOW WE HAVE A BASE ON WHICH TO AUTOMATE

- Out of Scope

SORRY!

P.S. If anyone wants to talk actual automation, tooling, frameworks, etc, I'd love to chat.

THE END

QUESTIONS?

AARON MURRIHY
aaron.murrihy@reannz.co.nz
help@reannz.co.nz


REAWNZ