Our use of DevOps practices
Intro to Matt

SFMIX
AS12276 | 63055

Cumulus
AS62491

Square
AS15211

Tumbrlr
AS33612

VideoBox
AS36472

NANOG
AS19230 (56 - 59)
SFMIX ‘design patterns’

Slow start (established in 2006)
• Consider a site when owner welcomes space, power, cross-x’s along with an identified transport provider
• Initially transport agnostic (MPLS, dim, wave)

Mature as demand warrants
• Migrate to dark when possible (started last week!)
• Partner w/ colo sales ops & marketing for outreach/education

Regional patriotic effort
• Built and consumed by local volunteers & organizations
• No paid staff, no port or membership fees (yet)
Current Operations

• 10G switches in 5 locations
  — 2x San Francisco, San Jose, Santa Clara, Fremont

• 47 participant ASN’s
  — Regional eyeball heavy (WISP, FTTH), 4 DNS roots, my-first-BGP organizations, RouteViews
  — No major CDN’s (strange incumbent-only location policies)

• 54 active ports
  — 1/3\textsuperscript{rd} 10G, remainder 1G mix of fiber & copper

• 12Gbps peak
  — Average 3Gbps (5 min sampling)
Constraints

• Self funded
  — No grants, no venture capital, no sponsoring parent corp.
  — Keeps mission extremely focused and consistent

• Less than $25,000 USD of expenses over 9 years
  — Optics, patch cords, ARIN, insurance, non-profit incorporation, tax filing, hosted services, stickers
  — Significant increase when 5th IXP entered region (mostly legal efforts)
Constraints (cont.)

• Donations
  — Switches (no strong bias: SFP+, SSH, port security, etc.)
  — Optics (reliability over cost, flexOptix since 2014)
  — Backhaul (dark & waves — both plentiful in the region)
  — Servers (VM compatible, modern power draw, 1U)

• 3 switch NOS’s
  — NX-OS (Nexus 3048), JunOS (EX3300), & Cumulus Linux
  — Varied SNMP MIB’s, not all IXP-Manager compliant
Rationalize Infrastructure

• Labor is our #1 constraint
  — Always seems to be limit, regardless of org structure
  — Volunteers have day jobs, let’s not waste their time
  — Majority task: provisioning (switch config, DNS, website)

• Preference public cloud over private or internal
  — Keep OpEx costs down in short term, avoid large one-time CapEx expenditure
  — Cloud where services are affordable (yes - email hosting) & appropriate (no - route server)
(Public Cloud) Infrastructure

GitHub

- ‘Source of truth’, config. backups, website assets, documentation, Ansible

Google Apps

- Email hosting

Google Drive

- Design assets, contracts/legal papers

Duo Security

- 2 factor authorization

JIRA

- Ticketing: systems, provisioning, legal
(Private Internal) Infrastructure

- VM management
- Members mailing list
- Route server (IPv4 ~70% utilization)
- Looking glass
- Monitoring
• Critical for multi-vendor architecture
  – Common tool for compute VM’s & network bare metal
• Follows common “DevOps” tenants
  – Open Source, agent’less, friendly to many NOS’s
  – Input data as flat files and/or dynamic sources (ie: database)
  – Testing can occur in VM’s, outside production
  – Changes get checked into git VCS repository
• Minor tweaking required
  – Wrote Jinja2 (template engine) “text filters” to use IP address manipulation library, now stock in Ansible since v1.9
• Attend Elisa’s tutorial tomorrow! Or RIPE71 or NANOG65
participants:

- name: RouteViews
  url: 'http://www.routeviews.org/
  asn: !!int 6447
  email: help@routeviews.org
  v4_ip: 206.197.187.28/24
  max_prefixes_v4: 1
  v6_ip: '2001:504:30::ba00:6447:1/64'
  max_prefixes_v6: 1
  join_date: 4/1/2015
  switch_ports:
    - switch: switch01.sfo02
      port: swp36
      speed: 1G
      mac_addr: '00:25:90:d7:ea:33'
      cross_x_ports: within rack
router id {{ ipv4_route_server }};
#!/bin/sh

usage() { echo "Usage: $0 [-4 | -6] filename" 1>&2; exit 1; }

while getopt "4:6:" flag; do
    case "$flag" in
        4) filename=${OPTARG} prog=birdcl ;;
        6) filename=${OPTARG} prog=birdcl6 ;;
    esac
done

if [ ! -f "${filename}" ]; then
    echo "Unknown file"
    exit 1
fi

validate=$(echo "configure check \""$filename"\"" | "${prog}" )

if echo "$validate" | grep -q "Configuration OK" ; then
    exit 0
else
    echo "$validate" | logger
    exit 1
fi
- name: Verify BIRD is installed
  pkgng:
  name:{{ item }}
  state=present
  with_items:
    - bird
    - bird6

- name: Copy validation script
  copy:
    src=bird_validate.sh
    dest={{ bird_validate }}
    mode=a+rx

- name: Validate generated BIRD v4 config
  template:
    src=birdv4
    dest=/usr/local/etc/bird.conf
    backup=true
    validate='{{ bird_validate }} -4 %s'
  register: birdv4

- name: Reload BIRD v4
  shell: echo configure | /usr/local/sbin/birdcl
  when: birdv4.changed
% ansible-playbook -i inventory playbook.yml

TASK: [Compute current git checkout hash] *************************************************
changed: [localhost]

GATHERING FACTS *************************************************
ok: [sfo02-routeserver-freebsd]

TASK: [route-server | Copy validation script] *************************************************
ok: [sfo02-routeserver-freebsd]

TASK: [route-server | Validate generated BIRD v4 config] ******************************
changed: [sfo02-routeserver-freebsd]

TASK: [route-server | Reload BIRD v4] *************************************************
changed: [sfo02-routeserver-freebsd]

PLAY RECAP *************************************************
sfo02-routeserver-freebsd : ok=4   changed=2   unreachable=0   failed=0
router id 206.197.187.20;

# 6447: RouteViews
protocol bgp R24 {
  local as 63055;
  neighbor 206.197.187.28 as 6447;
  passive on;
  import all;
  export all;
  route limit 1;
  table T6447;
  connect retry time 6000;
  rs client;
}
**SFMIX Lessons**

- DevOps approaches have been our savior
  - Significant hoards of ‘DevOps engineers’, much less than ‘NetEng who can script’ types – take advantage of this!
  - Only dependency is Git repo checkout, travel friendly hacking!
  - Small IXP’s have staffing challenges, unified tooling is ideal
  - Facilitates “Hackathons”: up to 7 people working on the same code base,.. at the same,.. *all while drinking*

- Future work
  - Dynamic inventory to IXP-Manager API, an IPAM, etc.
  - Make git repo public