SONiC Development for Large Scale Operations

Guohui Wang
Engineering Director, Alibaba Group
Alibaba and the SONiC community

- Alibaba joined SONiC community in 2017;
- Active participation and contributions
  - TACACS
  - VLAN trunking
  - sonic-telemetry
  - SWSS warm reboot
  - SysDB and routing performance optimization
SONiC Adoption in Production

- Picture from SONiC website
Running SONiC Fleets in Production

- Configuration
- Monitoring
- Maintenance and failure recovery
- Software upgrade and iterations

Do things in better ways with the open platform!
Configuration Management with structured APIs

- CLI-based configuration
  - Interaction latency in seconds
  - Hard to parse, program and verify

- Configuration with structure gRPC APIs
  - API latency in ms
  - Easy to model, program and verify

BGP neighbor AS number update:

ASW-xxxx# config
ASW-xxxx (config) # bgp xxxxx
ASW-xxxx (config-router)# neighbor 1.1.1.1
remote-as yyyyy
ASW-xxxx (config-router)# do show running-config
...
Parse running config to verify
...
ASW-xxxx (config-router)# exit
ASW-xxxx(config)# exit

BGP neighbor AS number update:

SET BGP_NEIGHBOR/1.1.1.1/asn yyyyy;
GET BGP_NEIGHBOR/1.1.1.1/asn;
Configuration Management with structured APIs

- gNMI-based data schema
- Device abstraction layer
  - virtual DB paths to hide system details;
  - white-list keys for access control;
Traditional Device Monitoring

- SNMP polling inspection every 5 minutes
  - Long monitoring delay, inflexible structure, legacy code;
- Syslog monitor
  - Noisy data, hard to analyze;
- Black-box monitoring:
  - No internal software states;
Event-Based Device Monitoring

- Fast channel with gRPC
- Real time push notifications
- More detailed software states
- Easy to analyze with structured events
SONiC multi-DB optimization

- Single-instance RedisDB is a system-wide bottleneck
- SONiC re-structured to support multi-DB instances
  - Multi-instance DB configuration at build time;
  - Dynamic binding of DB clients and instances;
  - Separate DB instances for routing/monitoring/management, over 50% improvement on route installation perf;
Maintenance Example: Device Isolation

- Disruptive device isolation:
  - No switch and server coordination, purely rely on the propagation of link down event;
  - Large amount of packet losses between event 1, 2 and 3 on both outgoing and incoming traffic;
Graceful Device Isolation

- Coordinated events between switch and server through customized protocol;
- Graceful traffic failover and device maintenance;

1. Link disable
2. Tx disable
3. NIC failover
4. Disable ack
5. Route withdraw
6. Link shutdown

Packet losses of a TCP flow during device isolation

Original Graceful isolation
Software Upgrade

- From hot-fix/cold-fix to modular software upgrade
- Debian package and docker based upgrade;
- Tooling for modular version control;
- Non-disruptive software upgrade with docker and system warm reboot
Faster Iterations with AliNOS Emulator

- Fully virtualized SONiC device running in a VM
  - QEMU-kvm + ONIE x86_64-kvm;
  - SONiC + VM specific platform modules;
  - SDK/SAI + vASIC simulation model;

- Application Scenarios
  - Development and integration test;
  - Operation rehearsal;
  - Software verification and troubleshooting;

![Diagram of SONiC components and emulation setup](image)
Lessons learned

• Dangers are in the grey zone
  • Tricky issues come from platform/firmware/hardware-software interoperability problems;
  • Monitoring and real time fault detection is the key;
• Building systems with operations in mind
  • From CLI/SNMP to RPC-based system ;
  • Customized protocols for graceful operations;
  • Non-disruptive upgrade with docker and system warm reboot;
• Automated testing and operation rehearsal bring faster iterations;
Call to Action

• Building SONiC with strong operation supports
  • Operation tooling: failure handling, trouble-shooting;
  • Software-driven management interfaces;
  • System and network visibility;
  • Version control and software iterations;