LHCb Online migration to ISC Kea -
The modern DHCP server

Hristo Mohamed for the LHCb Online Team
CHEP 2019, 4-8 Nov, Adelaide Australia
LHCb Online Infrastructure

- Over 2000 physical machines, over 400 self hosted virtual machines
- LHC Run 3 \(\Rightarrow\) 6000+ DHCP enabled devices, Over 4000 physical machines, over 600 virtual machines, self-hosted Cloud Native solutions,
- Desire to use commodity (soft|hard)ware for all purposes
- Avoid complex network configuration (in an already complex network)
- Do not rely on fancy protocols to deliver basic service (DHCP)
- Need highly available, scalable and easy to manage solution
- Easy instrumentation using APIs
ISC DHCP - battled tested and proven

- Enterprise-grade solution for IP address-configuration needs
- Complete open source solution for implementing DHCP servers, relay agents, and clients
- Hundred of thousands of users
- Basically every System Administrator has experience with it
- Proven and tested - 20 years since first release
2019 > 1999

- ISC DHCP is stateful
2019 > 1999

- ISC DHCP is *stateful*
2019 > 1999

- ISC DHCP is *stateful*
2019 > 1999

- ISC DHCP is **stateful**
- ISC DHCP has no REST API (OMAPI is absolutely horrible)
- Static configuration files, data pipeline just to keep those in sync (ouch!)
- Constant service reloads
- Master/Slave configured through same config files
- “Database backend” support through LDAP
- The above is also version v1
Enter Kea the modern DHCP Server

- ISC Kea is **stateless**
- Multiple Kea Servers can share the same DB instances
- ISC Kea REST API
- Configuration can live mostly (Kea v1.5) in database - BASE (Cassandra) and ACID (MySQL, PostgreSQL) support
- No more service reloads
- High Availability Mode
- Extensible through custom hooks
- Premium features come at a low cost (550 USD / year) - Premium hooks library, RADIUS integration, Forensic Logging, full configuration in the DB
Three steps to win

- Deployment of Kea instance per subnet (obsolete DHCP helper addresses)
Three steps to win

- Deployment of Kea instance per subnet
- Have a highly available DB backend
Three steps to win

- Deployment of Kea instance per subnet
- Have a highly available DB backend
- Get instrumentation for other engineers
Multiple Kea instances per subnet

- Super small memory footprint (largest is the Prometheus metrics exporter) -
  
  Private + Shared = RAM used Program
  1.3 MiB + 559.5 KiB = 1.9 MiB kea-ctrl-agent
  6.1 MiB + 658.5 KiB = 6.7 MiB kea-dhcp4
  15.6 MiB + 282.5 KiB = 15.8 MiB kea-exporter

- Can live inside of a container (host networking) - Opens up a whole realm of deployment possibilities

- You can use service discovery and systemd units (e.g. Consul CaS operations, zookeeper)
Have a highly available DB backend

- We went with PostgreSQL and Patroni (https://github.com/zalando/patroni)
- Multiple DBs in passive (read only) mode $\Rightarrow$ One active (read+write) DB instance
Have a highly available DB backend

- We went with PostgreSQL and Patroni (https://github.com/zalando/patroni)
- Multiple DBs in passive (read only) mode $\Rightarrow$ One active (read+write) DB instance
Proper instrumentation

- Easy to use REST API $\implies$ Fast creation of tools for other engineers
- Authentication support through RADIUS
- For exotic cases one can instrument through modifications in the DB itself

(Warning sending HUP to Kea for a reload, sometimes causes it to... freeze? and need a full restart)
Grumpy Sysadmin

Puppet
Provisioning
Kea Plugin
DNS Plugin

Foreman

Add/Remove/Edit entry

Consul
Send to a healthy instance

KEA Instance
KEA Instance
KEA Instance

Read/Write

DB Backend

Read/Write

Hristo Mohamed, LHCb Online migration to ISC Kea - The modern DHCP server, CHEP 2019, Adelaide
Summary

- ISC DHCP is aging
- ISC Kea provides a lot of modern features needed in a modern data center
- ISC Kea moves the HA aspect to a different area and focuses on being a DHCP server
- Rich REST API for easy instrumentation
Questions?