Ironic and Edgy
by Dmitry Tantsur and Ilya Etingof, Red Hat

In this talk...

• Why Edge Cloud?
  • Bare Metal provisioning at the Edge
  • Ironic at the Edge: now and in the future

Why Edge Cloud

• Low-latency, data-hungry applications:
  • IoT and smart homes
  • 8k video delivery
  • Economically viable
  • AI-managed data centres
  • Autonomous or self-driving data centres

Challenges at the Edge

• No living soul to "turn it off and on again"
• Remote management over slow, lossy and unreliable network
• Low footprint: limited space for management hardware
• Security concerns: larger control plane, unguarded locations

Bare metal on the raise

Why?
• Build, repair and re-purpose the cloud remotely
Trends:
• Converged infrastructure management e.g. servers, switches, storage, power
• Reliable and secure management protocols

The Ironic project

• Official OpenStack bare metal project since the Kilo cycle
  • Plugs into the Compute service (Nova)
- Lively upstream community:
  - **Rocky**: 359 commits, 81 contributors, 24 companies.
  - Established relationships with hardware vendors (Dell, HPE, Fujitsu, Lenovo, Cisco).
  - Support for old and new industry standards (IPMI, PXE, iPXE, Redfish, SNMP, UEFI).

**Ironic in OpenStack**

![Figure 1.1. Conceptual Architecture](image)

**Classic ironic architecture**

![Classic ironic architecture diagram](image)
**Ironic in action**

**Challenges at the Edge**

Challenges:

- PXE for boot management is unreliable
- DHCP over WAN unreliable and insecure
- IPMI is unreliable and insecure
  - Quiz: do you know about Cipher 0?
- AMQP for RPC
- Low bandwidth

**Reshaping Ironic for the Edge**

Solutions:

- Federated Ironic
- Booting via virtual media or UEFI HTTP boot
- DHCP-less boot over virtual media
- Direct image streaming
- HTTP-based protocols instead of IPMI

**Federated architecture**

To decentralize and distribute ironic, yet maintaining joint view on nodes:

- Conductors groups
  - Grouping conductors and nodes together
- Lightweight RPC
  - A prototype exists to replace AMQP-based RPC with JSON-RPC
Architecture with conductor groups

Federated architecture
If distributed/replicated database is also not an option, the following ideas are being considered (but not yet implemented):

- Per-conductor database
  - Requires substantion rework of the API implementation.
- Federating API proxy
  - prototype: github.com/dtantsur/ironic-proxy

Booting is fragile
Network boot is complicated and unreliable

- Network stack initialization
- Boot image transfer

The history of network booting

- PXE: BOOTP/DHCP -> TFTP
- iPXE: BOOTP/DHCP -> HTTP/iSCSI
- UEFI: BOOTP/DHCP -> HTTP/iSCSI
- Virtual Media: HTTP, SMB, NFS

Redfish: virtual media
BMC pulls image and serves it to the system
Features:

- Layer-3 based deployment possible
- Ensures authentic boot image
- Ability to cache boot images

Non-network boot over virtual media

- Ironic deploy image still requires DHCP
- Virtual Media offers virtual floppy o/
Image streaming

- Streaming images directly to the block device
- Idea: distributing images via Bit-Torrent

Summary: Ironic has an Edge

The upcoming features:

- Federated architecture
- Reliable boot methods
- Efficient image delivery