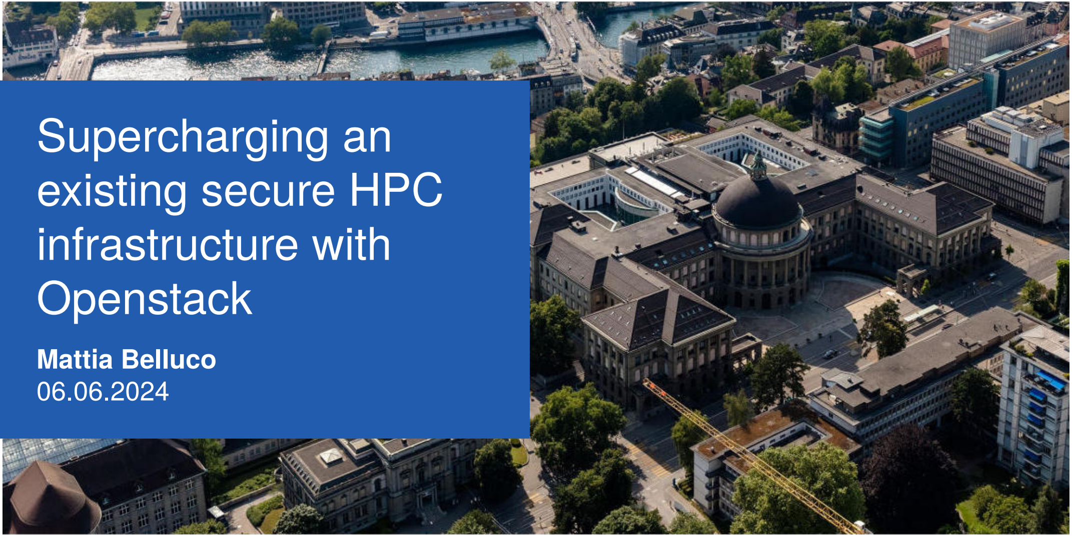
ETH zürich

Scientific IT Services

Supercharging an existing secure HPC infrastructure with Openstack

Mattia Belluco

06.06.2024



Outline

1. Scientific IT Services at ETH Zurich
2. Leonhard Med: a secure HPC infrastructure
3. Leomed: a flexible Trusted Research Environment
4. Takeaways
5. Acknowledgement

Outline

ETH Zurich

Founded in 1855

- Driving force of industrialisation in Switzerland

ETH Zurich today

- One of the world's leading universities for technology and the natural sciences
- Place of study, research and employment for 30,000 people from over 120 different countries

Reasons for success

- Excellent education
- Groundbreaking fundamental research
- Knowledge transfer that benefits society

SIS: Scientific IT Services

- A section of ETH Zürich IT Services
- About 45 experts in various areas of scientific computing
- With a background in different areas of science

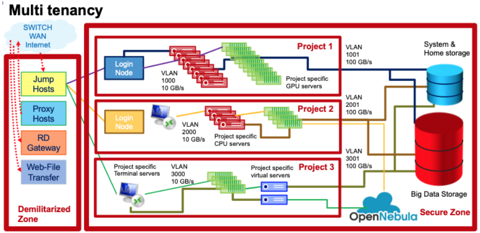
Outline

Leonhard Med

The first iteration

Starting in 2017, the HPC team within SIS had been operating a secure HPC infrastructure called **Leohnard Med**. It consisted of segmented HPC clusters with:

- dedicated baremetal login nodes and compute/GPU nodes
- access from trusted networks via jumphost and 2FA
- distinct network zones with strict inter-zone rules
- a Lustre system running on a dedicated network and a SSD based NAS for home directories
- a OpenNebula deployment to provide VMs



Leonhard Med

Analysis of the status quo

Shortcomings of the existing infrastructure:

- Cumbersome and time consuming process to create new projects requiring physical changes to the infrastructure and dedicated hardware procurement
- Whole compute/GPU nodes needs to be procured and provisioned to each project
- Fullfills only standard HPC use cases (no redeploy of OpenNebula following the 2020 HPC Centers hack)

Outline

1. Scientific IT Services at ETH Zurich
2. Leonhard Med: a secure HPC infrastructure
3. Leomed: a flexible Trusted Research Environment
4. Takeaways
5. Acknowledgement

Leomed

End of 2020: a new beginning

New team assembled and tasked with deploying the second iteration of the infrastructure, now called **Leomed**.

Chance to start from scratch?

External requirements

- Re-use the existing hardware
- Ability to interface with existing storage systems

Leomed

Design: General ideas

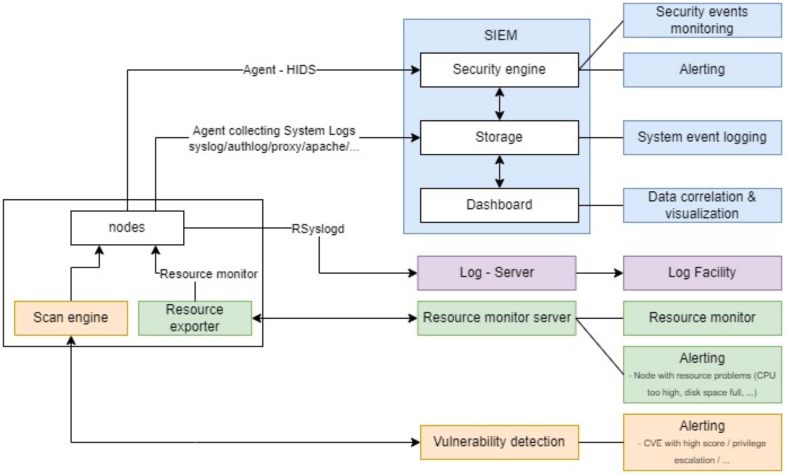
Provide a large degree of flexibility to tackle both current and future usecases

- Everything virtualized:
  - move the security boundary from the physical host to the VM
  - wrap the VMs in an extensive set of security groups (including egress rules)
- Openstack as the orchestration layer
- CephRBD and CephFS as the only storage backends
- Offload the network infrastructure provisioning to our network team (with shiny 100 GbE switches)
- Infrastructure as code from top to bottom (Ansible + internal ETH APIs)
  - Bonus: Total redeployment should be possible in a limited time frame (days)

Leomed
Design: Security

- No direct user access to the Openstack API
- Run only supported components (needs to be easy to upgrade)
- Separate monitor stacks for different security domains
- HIDS agents deployed on all VMs
- NIDS running on the most sensitive VMs (reverse-proxy)
- SIEM with lots of dashboards to keep everything under control

Leomed
Logging and monitoring



Leomed

Implementation: deploy Openstack

Drawing from the experience of running an Openstack deployment based on Ubuntu packages running directly on bare metal nodes:

- **We want to decouple Openstack components from the baremetal host underneath**
- We want to stay distribution agnostic
- We don't want to deploy Openstack on top of k8s as we are worried that it could add an unneccesary (for our target scale) additional layer to debug in case of issues.
- We want a fully virtualized network layer (OVN with Geneve tunnels)

Kolla ansible wins us over for the speed of deployment, the prebuilt container images, and the fact that's easier to customize to our needs.

Leomed

Implementations: Validation process

Testing phase out of decommissioned hardware to help us:

- Validate architectural choices
- Gather a first hand idea of possible issues
- Write the Ansible playbooks to provision and configure the required components

Leomed

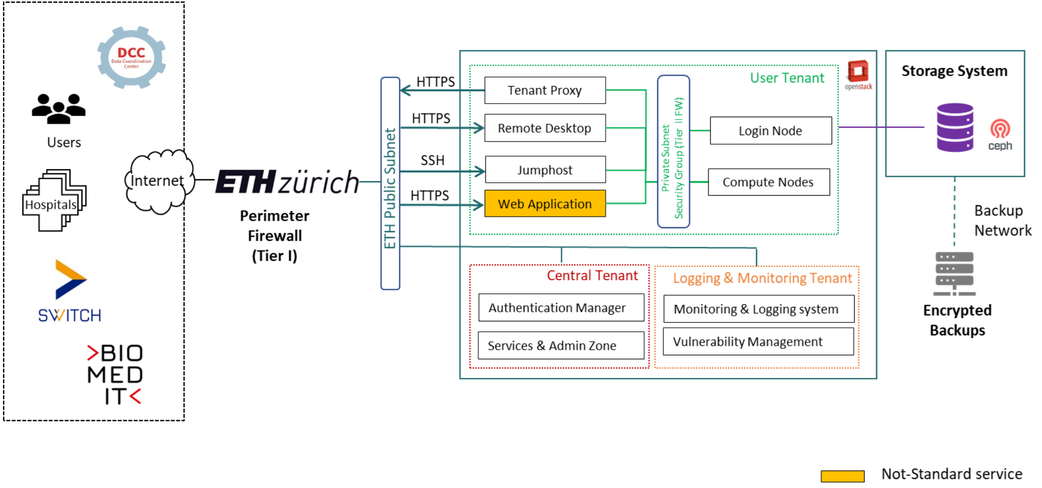
Implementations: Iron out some kinks

A few of the issue we encountered:

- OVN network paths got inconsistent and drive crazy our switches (packets from the same mac address are coming out of two distinct control nodes).
- Duplicate ports with same MAC address on OVN chassis make VM unreachable (manually delete the spurious ports fixed the issue)
- We cannot start a VM with more of 2 TB of RAM (easy fix: change machine type in kvm by appeding "-hpb" to the default one)
- Rabbit meltdown (required a cluster redeploy to fix and a global service restart to recreate the queues)
- Removing the default allow all egress rule makes VM unbootable (needs explicit rules for DHCP and metadata server)
- More recently: no way to boot a VM with more than 1.5 TB of RAM when using PCI passthrough (fix in QEMU > 7.1)

Leomed

How it ended up looking



Leomed

How to move users and workloads from the old infrastructure to the new one

- Migration happened in stages to create minimal disruption to users
  - compute nodes can be easily swapped for equivalent VMs (and eventually be live migrated), GPU nodes less so: bespoke VMs needs to be created beforehand
  - each node needs to be removed from the existing batch queue system and handed over to us for cabling, cleanup, inspection and provisioning
- More than 2 PB of data to be migrated from the Lustre system
  - We endep up virtualizing LNET routers during the migration

Leomed
Numbers

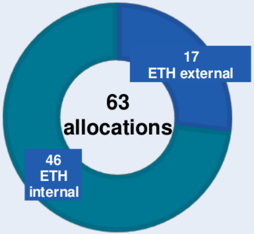
> 507 researchers

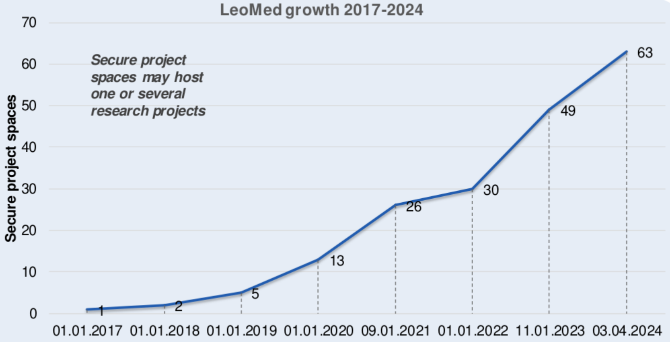
> 3.2 PB secure data storage

> 10 **Data Providers (DPs)** Swiss (e.g. USZ, EOC) and international (UCSF)

> 1500 **Data Transfer Requests** from DPs in active reseach projects

Leomed Customers Affiliation (Eth-internal Or External)





Leomed
Numbers

How big is the whole infrastructure?

We are currently managing 713 VMs spread over about 100 hypervisors with an average total power consumption of 90 KW (we have lots of GPUs nodes)

Leomed

Bonus 1: What about K8s?

The Magnum project looked very promising but because of the rapid development cycle of K8s we opted for Kubespray to stay more current with K8s versions.

The kubespray deployed K8s clusters are hosting only services controlled by operators.

New concept has been recently introduced with dedicate per-project K8s cluster that can be managed by users via a Rancher managed API proxy.

Leomed

Bonus 2: Is there such thing as too many security group rules?

Security Groups

sisrp-monitoring	ALLOW IPv4 icmp from 10.215.92.35/32
	ALLOW IPv4 6556/tcp from 10.215.92.35/32
	ALLOW IPv4 21047/tcp from 10.215.92.35/32
sisrp-nfs	ALLOW IPv4 2049/tcp to 129.132.249.43/32
	ALLOW IPv4 2049/tcp to 10.215.92.30/32
	ALLOW IPv4 2049/tcp to 129.132.249.11/32
	ALLOW IPv4 2049/tcp from 10.215.92.0/23
sisrp-slurm	ALLOW IPv4 80/tcp to fb235f37-611d
	ALLOW IPv4 9100-9101/tcp from fb235f37-611d
	ALLOW IPv4 30000-32767/tcp to fb235f37-611d
	ALLOW IPv4 3128/tcp to 0f70baba-b351
	ALLOW IPv4 1965-1966/tcp to 129.132.1.108/32
	ALLOW IPv4 to sisrp-slurm
	ALLOW IPv4 from sisrp-slurm
	ALLOW IPv4 443/tcp to fb235f37-611d
sisrp-ssh_jumphost	ALLOW IPv4 22/tcp from 26380325-a378
sisrp-default	ALLOW IPv4 636/tcp to 129.132.65.0/26
	ALLOW IPv4 514/udp to 129.132.30.112/28
	ALLOW IPv4 443/tcp to 192.42.198.13/32
	ALLOW IPv4 123/udp to 192.33.96.102/32
	ALLOW IPv4 601/tcp to 129.132.234.192/27
	ALLOW IPv4 1514-1515/tcp to 129.132.249.35/32
	ALLOW IPv4 9200/tcp to 129.132.249.35/32
	ALLOW IPv4 80/tcp to 192.42.198.13/32
	ALLOW IPv4 80/tcp to 129.132.249.14/32
	ALLOW IPv4 123/udp to 129.132.97.15/32
	ALLOW IPv4 443/tcp to 129.132.249.14/32
	ALLOW IPv4 53/udp to 129.132.98.12/32
	ALLOW IPv4 25/tcp to 129.132.235.70/32
	ALLOW IPv4 636/tcp to 129.132.9.60/32
	ALLOW IPv4 25/tcp to 129.132.235.75/32
	ALLOW IPv4 80/tcp to 37.218.245.50/32
	ALLOW IPv4 636/tcp to 129.132.9.3/32
	ALLOW IPv4 123/udp to 192.33.96.101/32
	ALLOW IPv4 6514/tcp to 129.132.30.112/28
	ALLOW IPv4 514/tcp to 129.132.234.192/27
	ALLOW IPv4 587/tcp to 129.132.235.70/32
	ALLOW IPv4 601/tcp to 129.132.30.112/28
	ALLOW IPv4 123/udp to 192.3.96.102/32
	ALLOW IPv4 53/tcp to 129.132.250.2/32
	ALLOW IPv4 443/tcp to 129.132.249.16/32
	ALLOW IPv4 55000/tcp to 129.132.249.35/32
	ALLOW IPv4 53/udp to 129.132.250.2/32
	ALLOW IPv4 443/tcp to 129.132.249.4/32
	ALLOW IPv4 68/udp from 10.215.0.0/16
	ALLOW IPv4 3128/tcp to 129.132.249.12/32
	ALLOW IPv4 80/tcp to 129.132.249.13/32
	ALLOW IPv4 53/tcp to 129.132.98.12/32
	ALLOW IPv4 443/tcp to 129.132.249.15/32
	ALLOW IPv4 123/udp to 192.3.96.101/32
	ALLOW IPv4 636/tcp to 129.132.183.39/32
	ALLOW IPv4 587/tcp to 129.132.235.75/32
	ALLOW IPv4 514/udp to 129.132.234.192/27
	ALLOW IPv4 80/tcp to 169.254.169.254/32
	ALLOW IPv4 6514/tcp to 129.132.234.192/27
	ALLOW IPv4 80/tcp to 129.132.153.36/32
	ALLOW IPv4 514/tcp to 129.132.30.112/28
	ALLOW IPv4 123/udp to 129.132.29.5/32
	ALLOW IPv4 443/tcp to 37.218.245.50/32
	ALLOW IPv4 443/tcp to 131.152.229.51/32
	ALLOW IPv4 67/udp to 0.0.0.0/0
	ALLOW IPv4 80/tcp to 131.152.229.51/32
	ALLOW IPv4 514/tcp to 129.132.249.17/32
	ALLOW IPv4 443/tcp to 129.132.249.13/32
	ALLOW IPv4 636/tcp to 129.132.183.128/27
sisrp-ctr	ALLOW IPv4 8443/tcp from 129.132.249.97/32

We assign granular security group rules to regulate network connections: they include rules for egress and for intra-tenant communication.

So far no limits has been reached.

Outline

Team size and effort

Openstack can be a very effective tool to be used as a foundation to build upon, provided some precautions are taken:

- Assess the health of the projects you plan to use unless you have the bandwidth or the resources to take over/outsource maintenance and/or development.
- When running a cloud is not your main business bringing inexperienced team members up to speed can be tricky.
- Factor in some external help: preemptively make contact with companies offering consultancy services both for training purposes and help in case of issues.
- Leverage the community for help: Openstack and Ceph have several point of presence online (IRC, mailinglists) and often one can find solutions to his/her problem just perusing the archives.
- Scaling can be challenging in team that work vertically such as ours, as the effort to manage a growing fleet of baremetal machines can be very time consuming.

Outline

Acknowledgement

The Leomed team

The HPC team that provided assistance and know-how during the intense months required for the migration to the new system.

Mattia Belluco
Cloud Architect
mattia.belluco@id.ethz.ch

ETH Zurich
IT Services
OCT G 35
Binzmühlestrasse 130
8092 Zürich, Switzerland
https://sis.id.ethz.ch