

Clouds on Clouds: OpenStack and Kubernetes With Cloud-Barista

Ph.D. Seokho Son
ETRI Special Fellow
CNCF Ambassador



Who am I



 seokho-son

 seokho-son

 seokho son (CNCF / K8s Slack)

Ph.D. in Computer Science,

Principal Researcher & Special Fellow,  

Ambassador,  **CLOUD NATIVE**
COMPUTING FOUNDATION 

Maintainer, CNCF Cloud Native Glossary



Maintainer, Kubernetes Website



Lead, Kubernetes SIG-Docs Subproject



Lead, Kubernetes SIG-Docs Korean I10n Team



Maintainer, M-CMP Platform



Maintainer, Cloud-Barista



Who am I

WooJu Son (손우주)

D+95

우주아빠
father of
the universe

WooJu == Universe (Space)

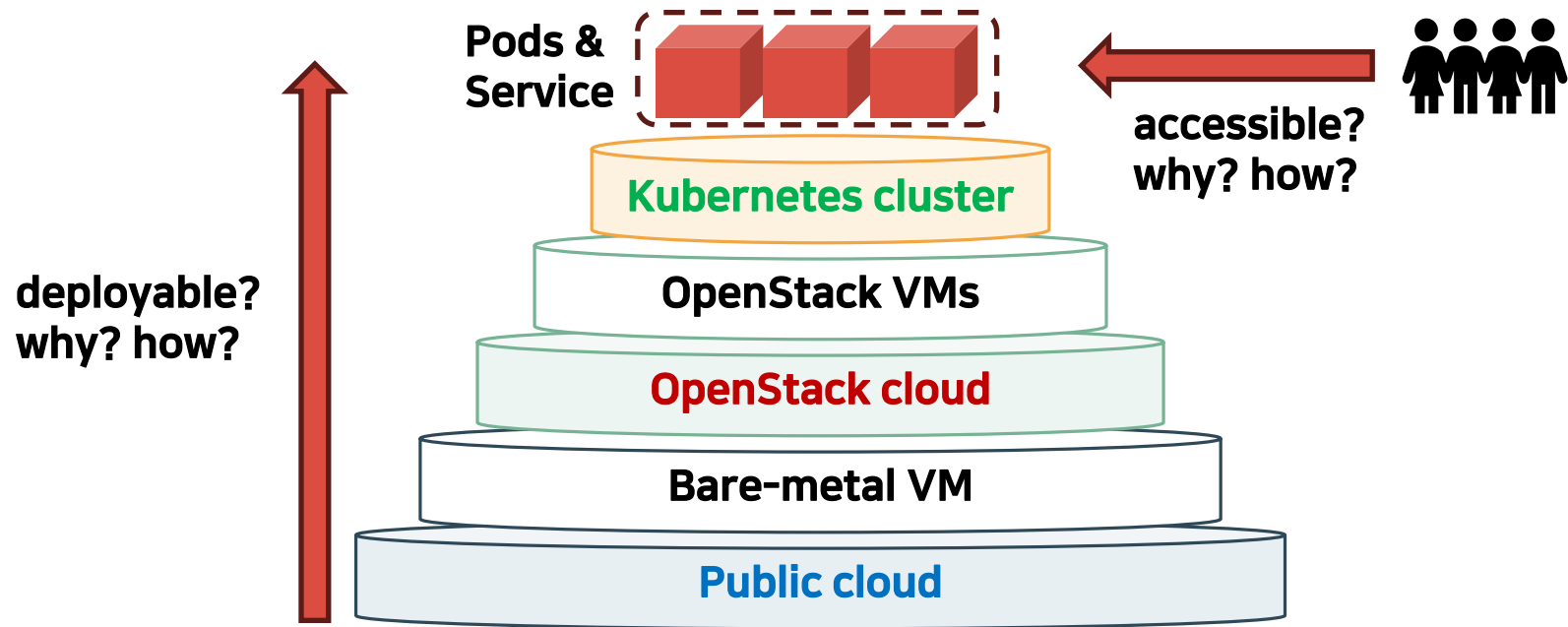


Understanding the title..

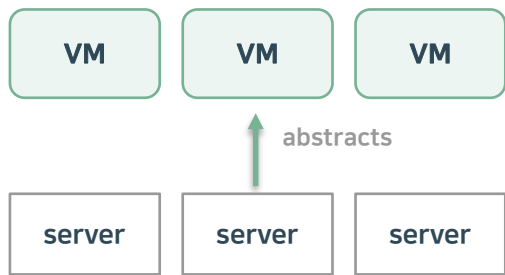
Clouds on Clouds: **OpenStack** and **Kubernetes** **With Cloud-Barista**

Session for beginners..
experimental and not practical

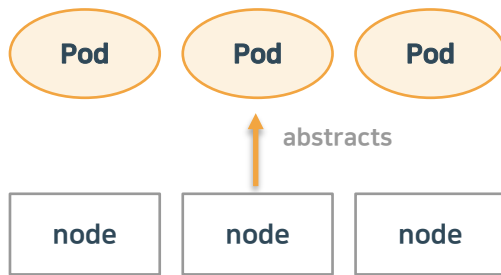
Lab: a **cloud** on a **cloud** on a **cloud**



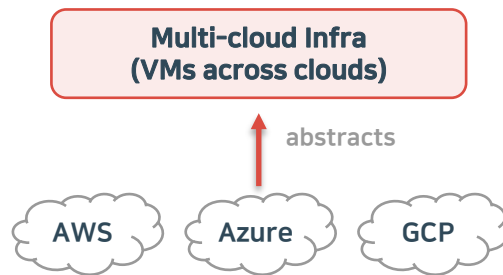
Three open sources, three different roles



hands you a VM



hands you a Pod (container)



hands you a multi-cloud Infra

OpenStack

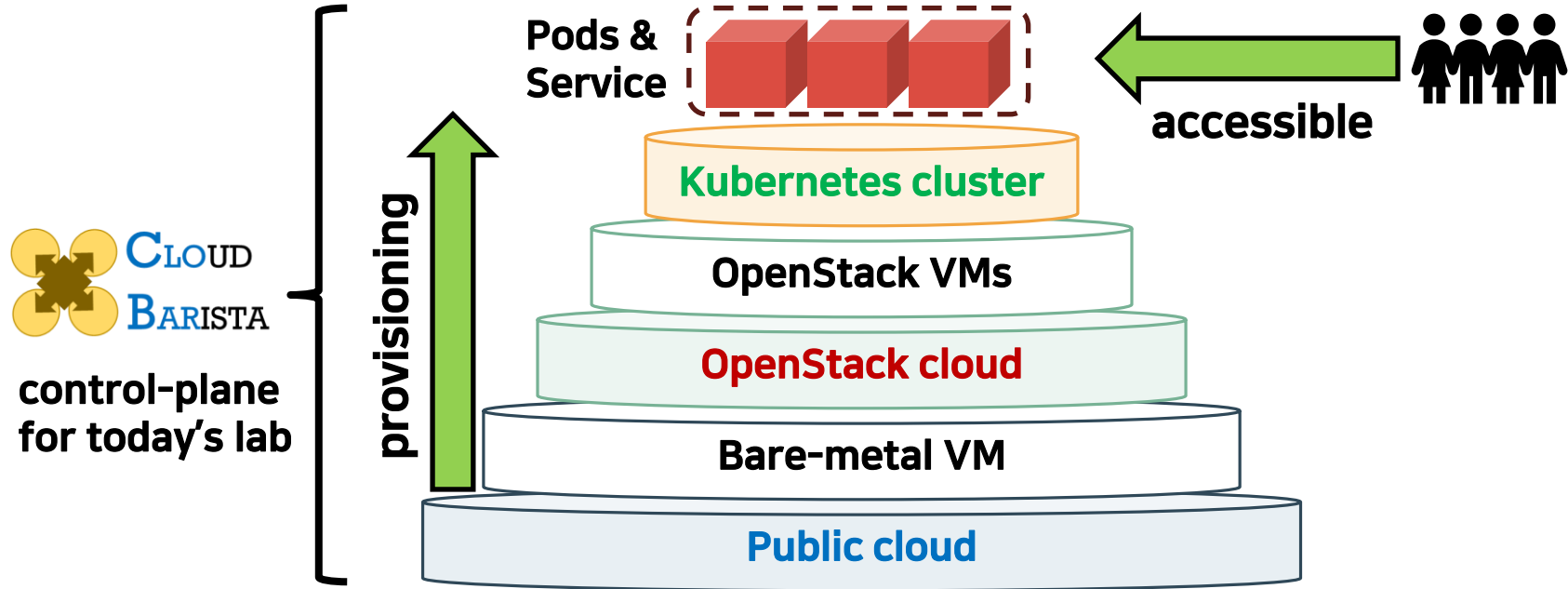
Kubernetes

Cloud-Barista

Family of CNCF & LF

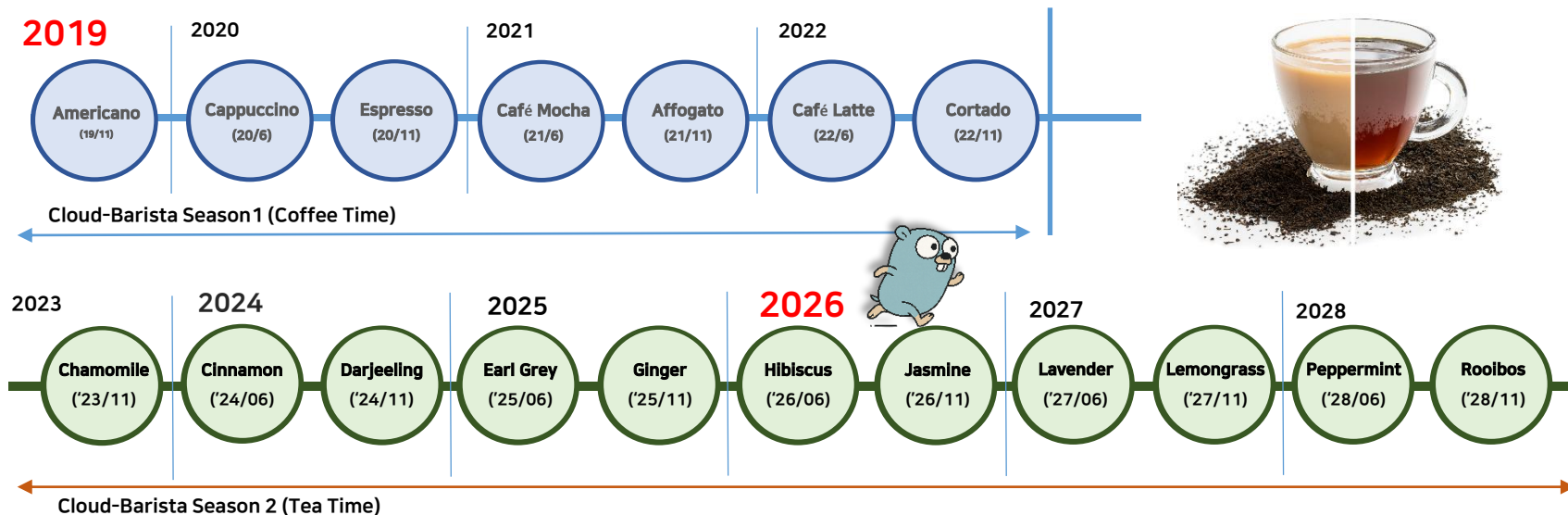
Friend

Lab: a **cloud** on a **cloud** on a **cloud**



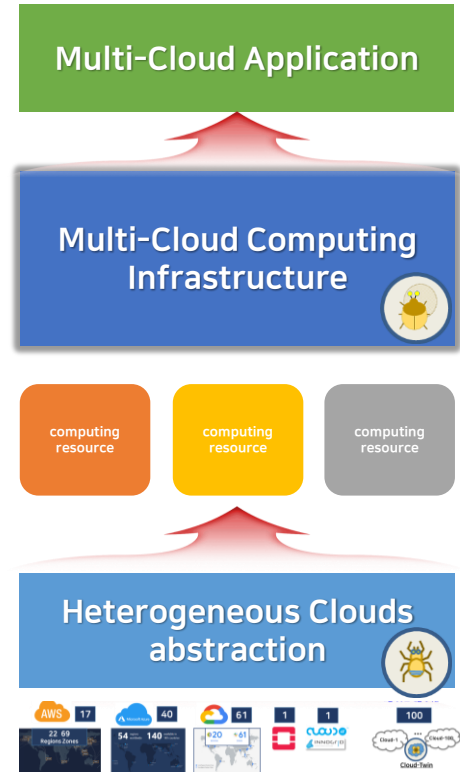
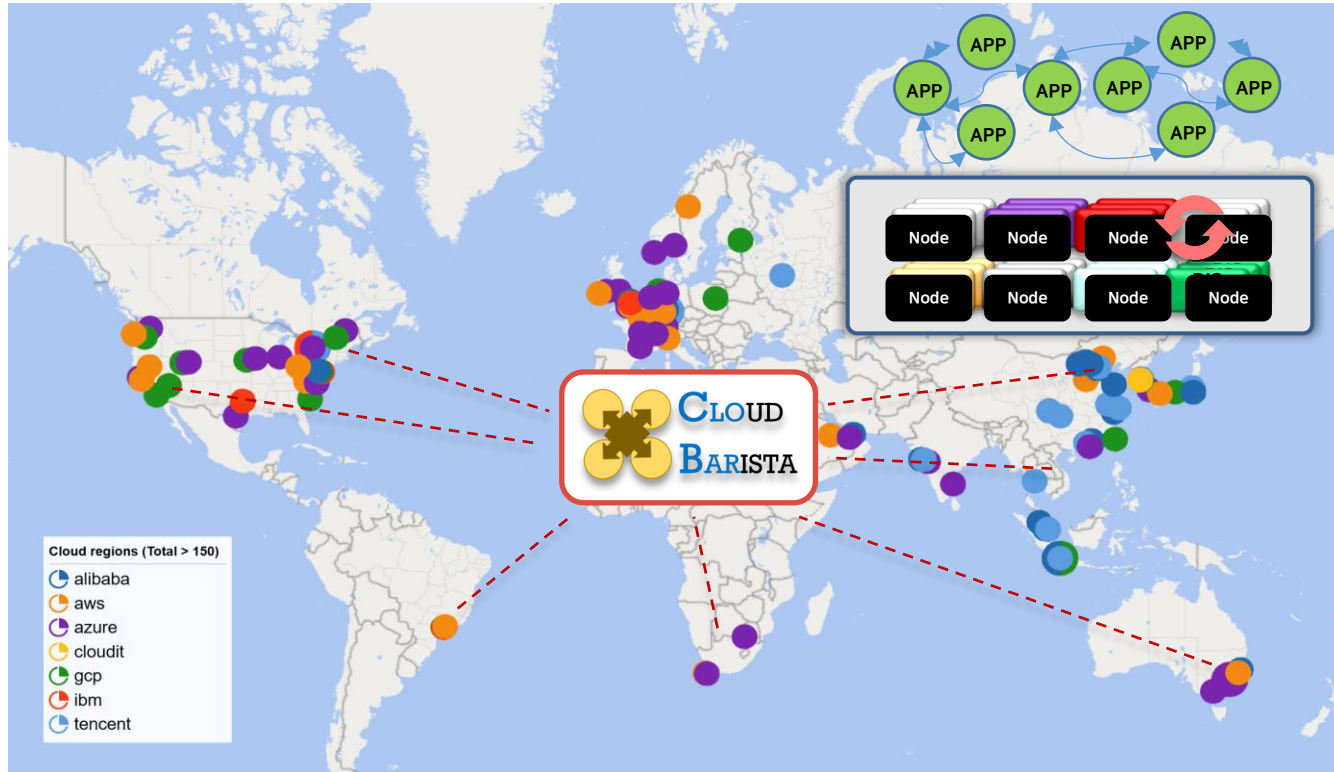
Cloud-Barista is an Open Source (Apache 2.0)

Project and community to provide essential functionalities for building multi-cloud solutions



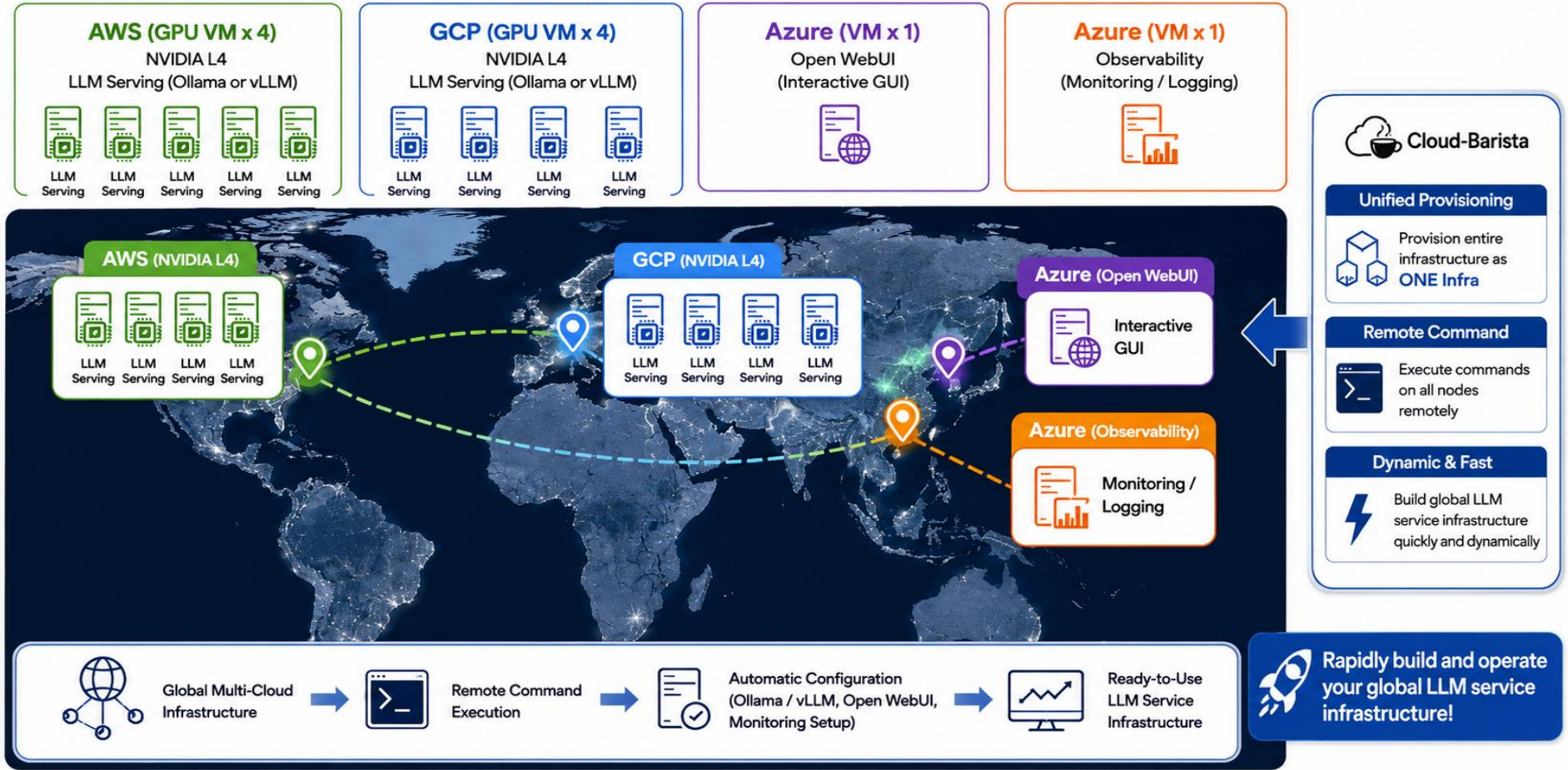
github.com/cloud-barista/cb-tumblebug

Manages Multi-Cloud Infrastructures !



Cloud-Barista: Multi-LLMs demonstration

Provision and configure a global LLM service infra dynamically



```
son@SON:~/go/src/github.com/cloud-barista/cb-tumblebug$  
son@SON:~/go/src/github.com/cloud-barista/cb-tumblebug$  
son@SON:~/go/src/github.com/cloud-barista/cb-tumblebug$  
son@SON:~/go/src/github.com/cloud-barista/cb-tumblebug$  
son@SON:~/go/src/github.com/cloud-barista/cb-tumblebug$  
son@SON:~/go/src/github.com/cloud-barista/cb-tumblebug$ m
```



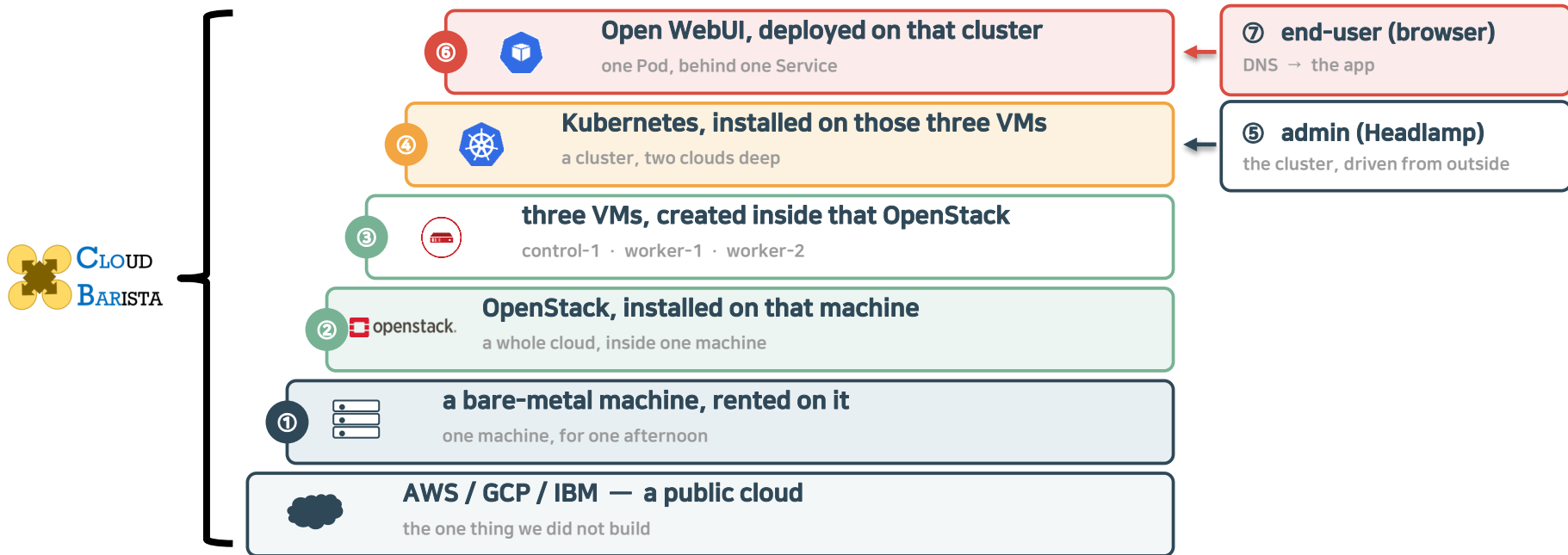
cloud-barista

opens API server.

"make up" "make init"

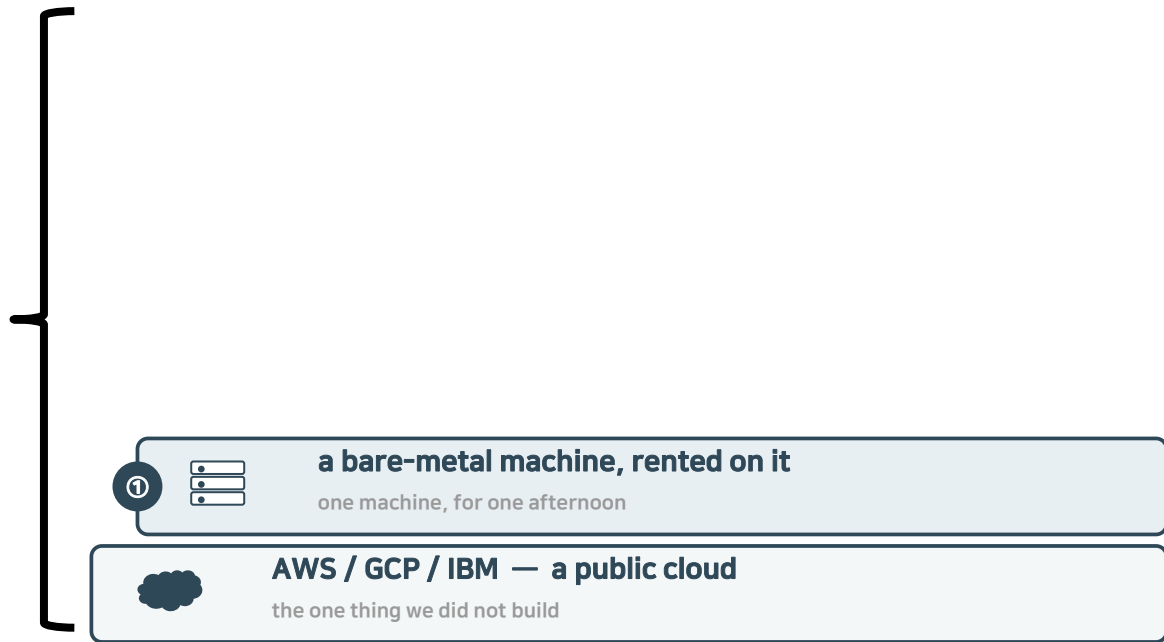
make everything ready.

Lab: a cloud on a cloud on a cloud



This is an experiment, not an architecture.

Lab: a cloud on a cloud on a cloud



This is an experiment, not an architecture.

Need bare-metal (OpenStack is hard to be deployed in VMs)

cloud VM

no /dev/kvm → QEMU software emulation → far **too slow**

OpenStack wants to run VMs here
= **nested virtualization**



the VM you rented

the cloud's own hypervisor

⚙️ **VT-x already taken**



a physical server

CPU ⚙️ **VT-x / AMD-V**

cloud bare-metal machine

/dev/kvm is there → KVM hardware acceleration → **real speed**

a real VM

a real VM

a real VM



openstack.
your OpenStack

⚙️ **VT-x is yours**



a physical server

CPU ⚙️ **VT-x / AMD-V**

A hypervisor needs the real CPU. So rent the machine, not a slice of it.

and this is not exotic — the big clouds all rent whole machines by the hour

AWS

EC2 *.metal

Google Cloud

c3 / c4a metal

IBM Cloud

Bare Metal Servers

Alibaba Cloud

ECS Bare Metal (ebm)

localhost:1324

MapTopolNetBoardAPI

ProvisionControl

MC-Infra Recommendation

PolicyLocation-based

ProviderALL

vCPU1

Mem(Gi)0.5

GPUselect

Count

VRAM(GB)

SpecIDEx: 2xlarge

SpecArchx86_64gx3-32x160x214

ImageUbuntu 22.04

Disk(GB)Ex: 250

WorkloadNodeK8s

Register Resources

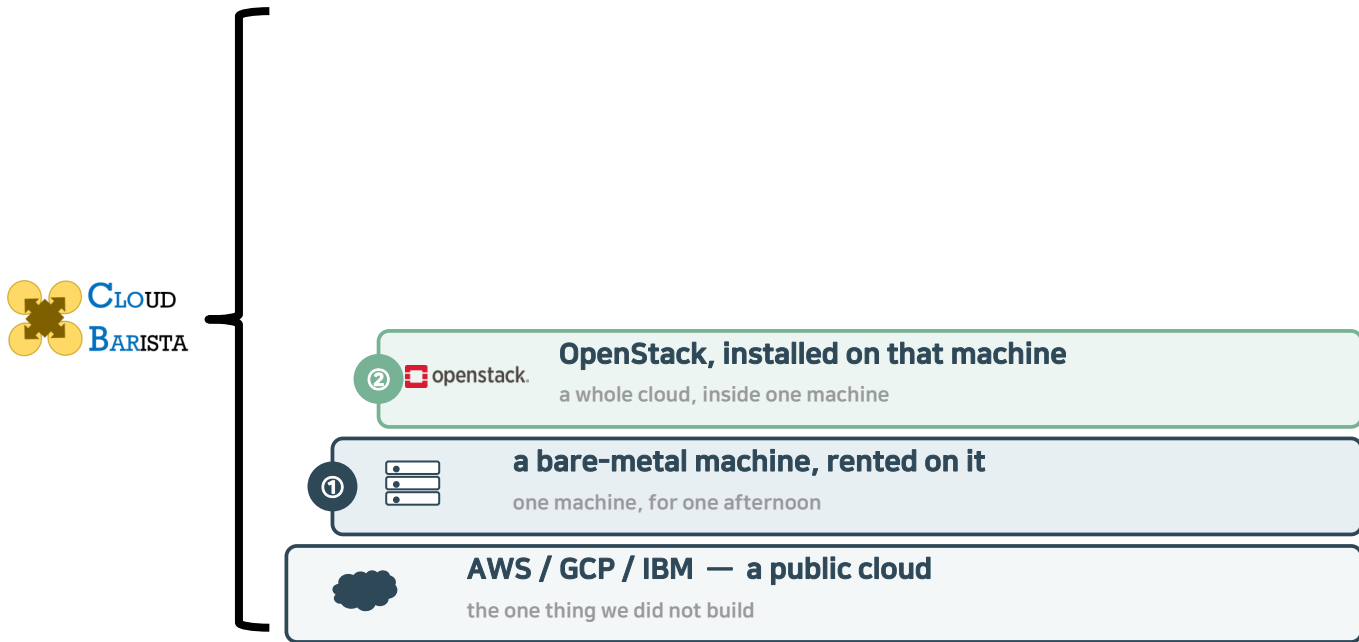
Template Management

A world map showing resource allocation points across various countries. The map is centered on Europe and Africa, with labels for many countries in both English and Arabic. The points are represented by small icons of different colors and shapes, indicating different resource types or providers. The map is interactive, with a zoom in (+) and zoom out (-) button in the top left corner. The map is also labeled with 'Kalaallit Nunaat' in the top left corner.

defaultadmin

© OpenStreetMap contributors

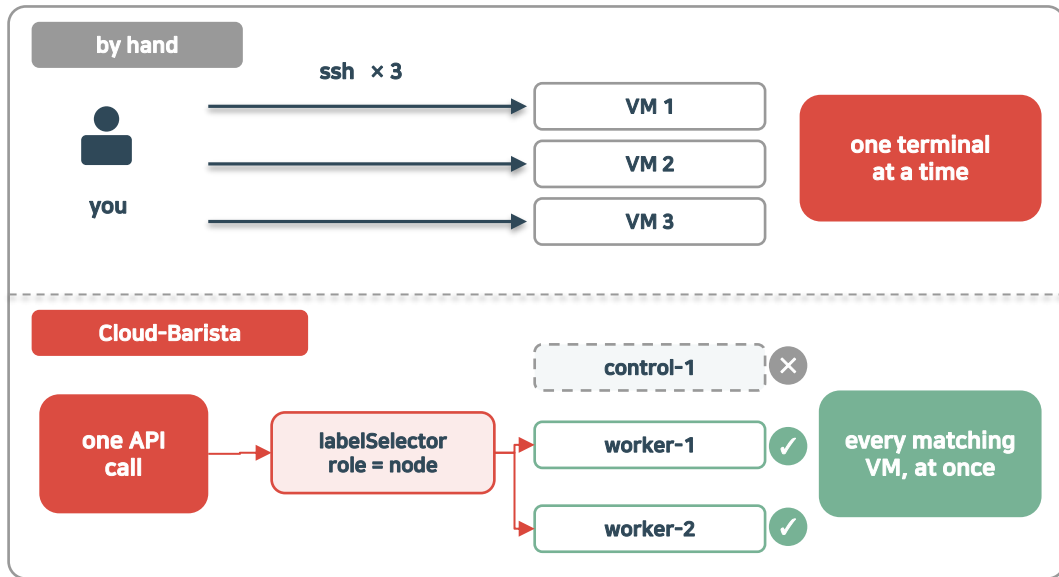
Lab: a cloud on a cloud on a cloud



This is an experiment, not an architecture.

Getting OpenStack onto that machine

① how does a command reach the machine?



② which installer?

DevStack

one script, one machine

dev · demo ★ this lab

Kolla-Ansible

containers, driven by Ansible

production

by hand

the install guide, service by service

learning

one remote command → one script → 17 minutes → a working cloud

Get Status

Get Access ▾

Remote Command

Remote File Transfer

Snapshot Management

ScaleOut NodeGroup

Update Firewall

Control L4 NLB

Delete Infra

Resource Control

Release Resources

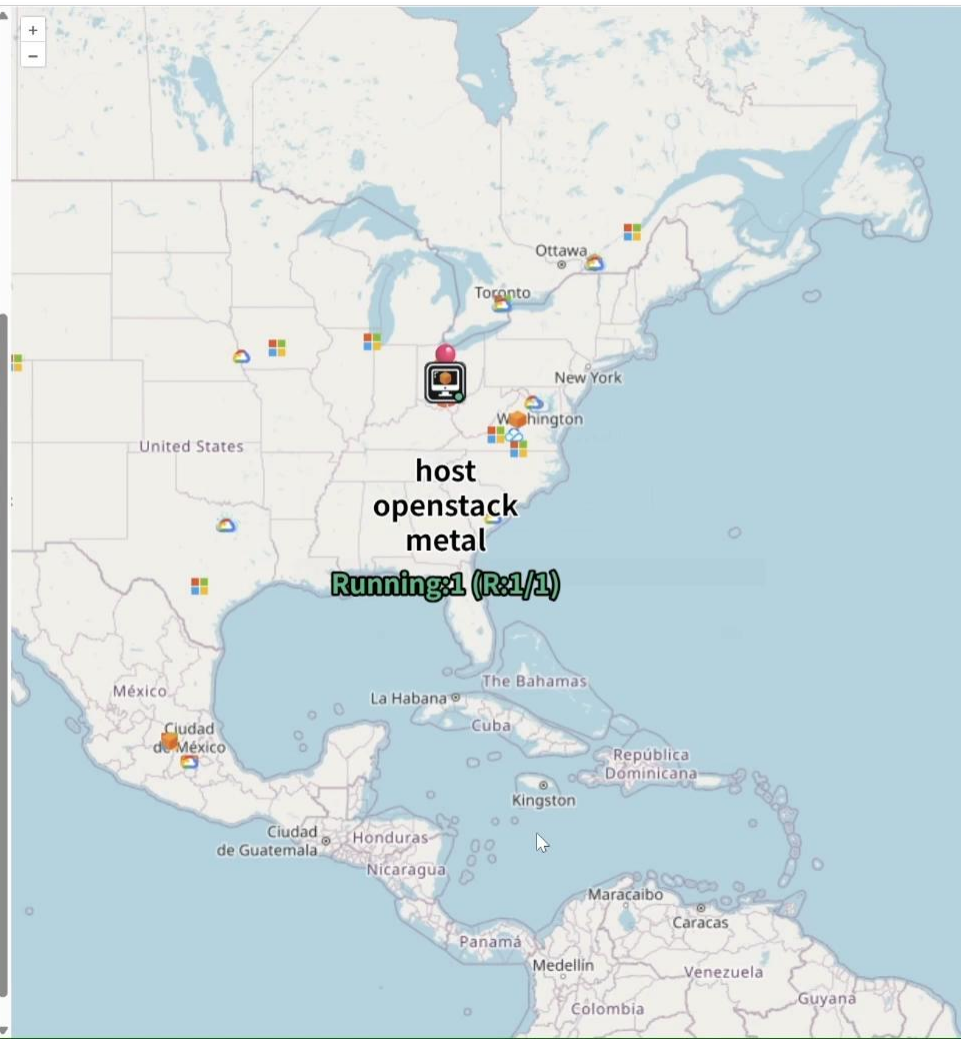
Resource Control ▾

VNet default-shared-aws-us- ▾

Security host-openstack-metal- ▾

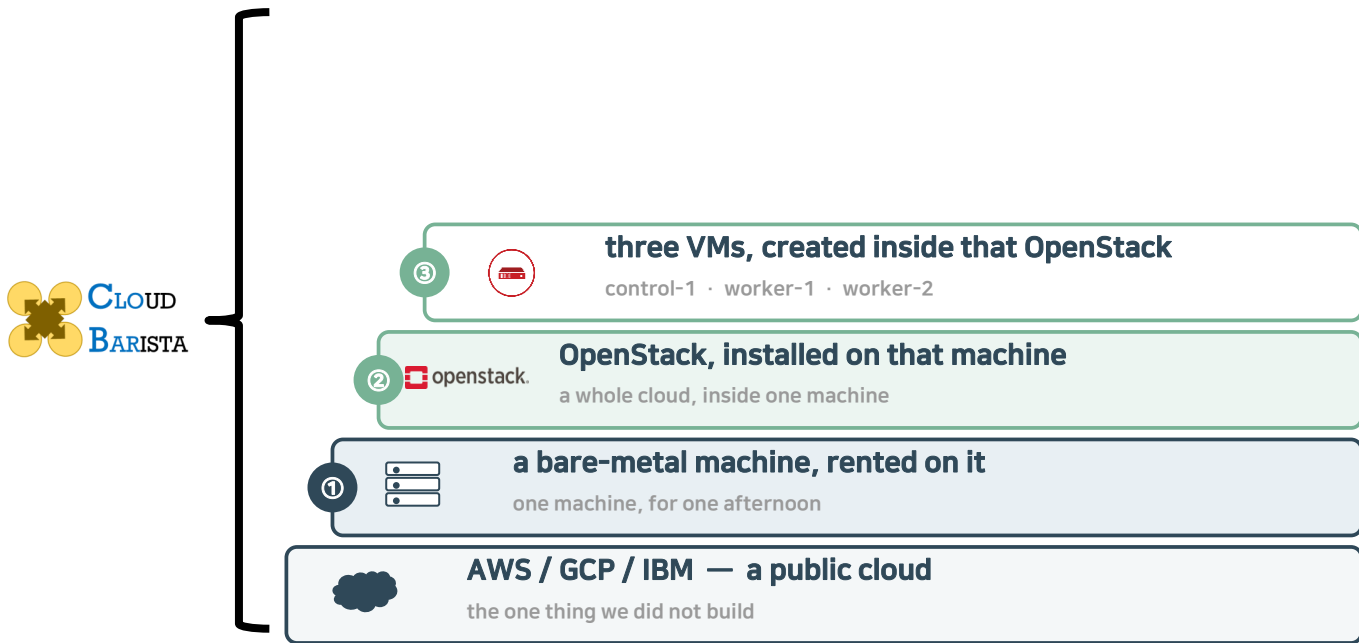
SSHKey host-openstack-metal- ▾

Connect ▾



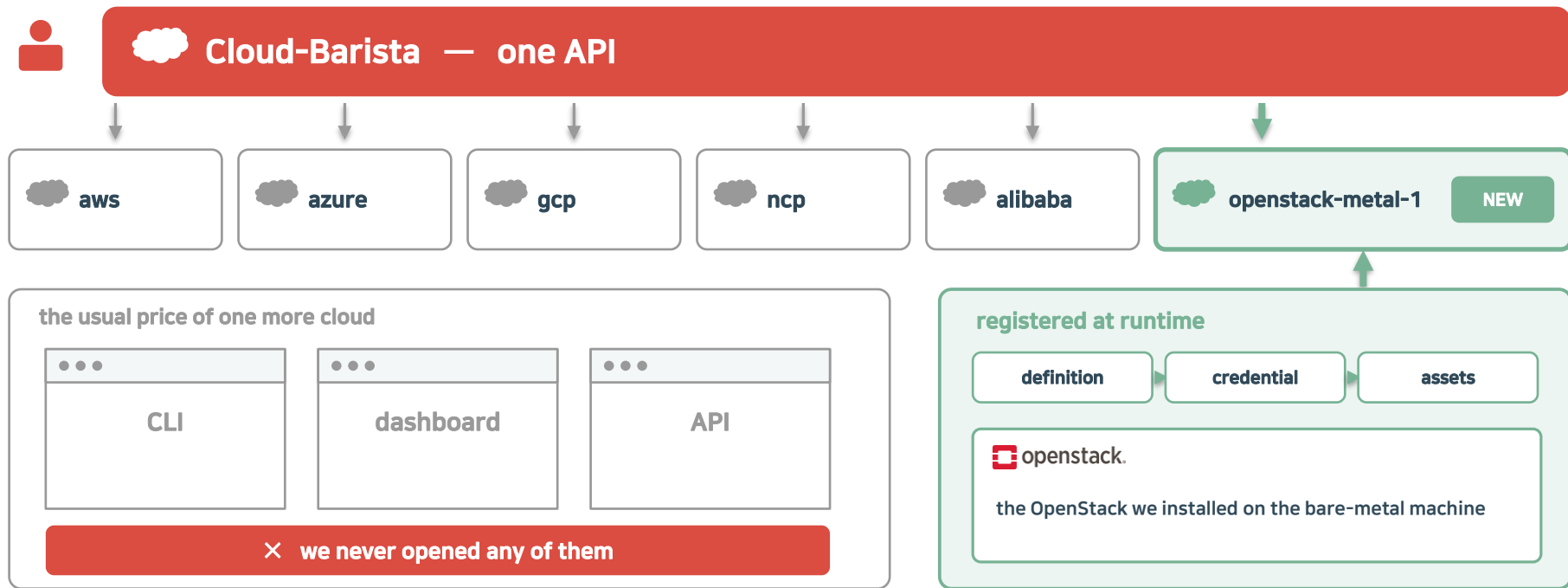
\$4.0800/h+

Lab: a cloud on a cloud on a cloud



This is an experiment, not an architecture.

You do not have to learn a second cloud



One more provider. The same control plane.

프로젝트

프로젝트 / API 액세스

API 액세스

API 액세스

- Compute
- 블록
- 네트워크
- 관리
- 인증

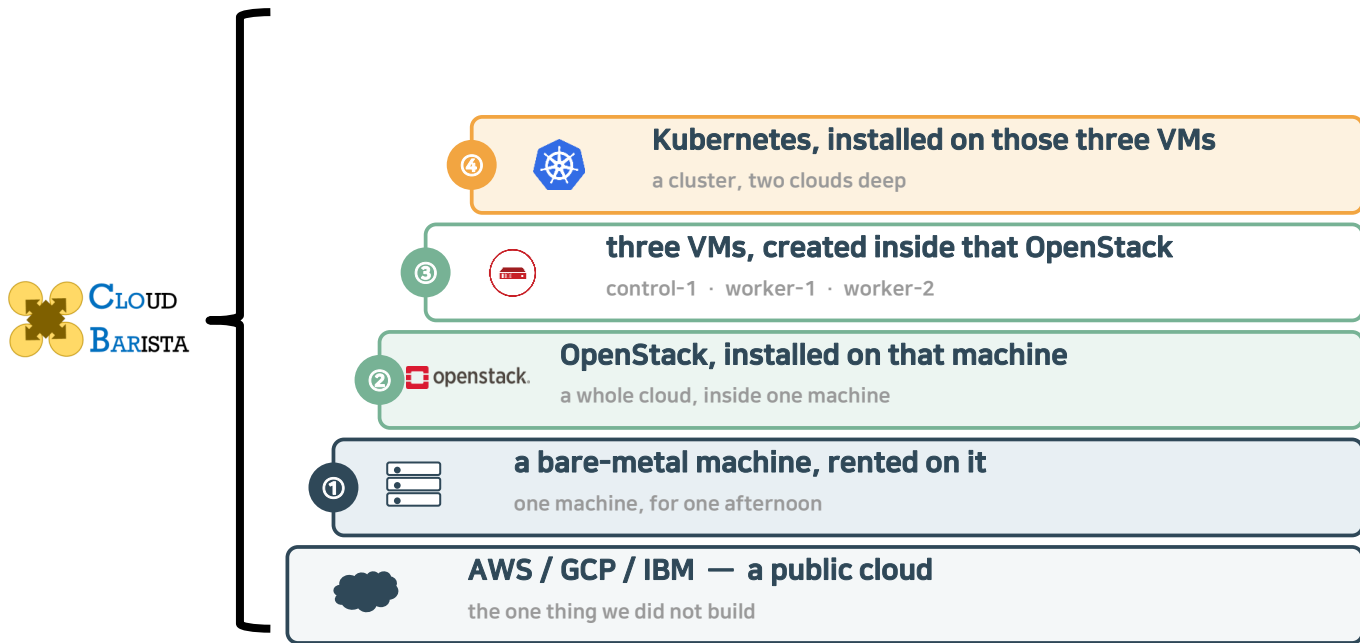
인증서 보기 OpenStack RC File 다운로드

12 항목 표시

서비스	서비스 Endpoint
Block Storage	http://3.144.76.134/volume/v3
Compute	http://3.144.76.134/compute/v2.1
Compute_Legacy	http://3.144.76.134/compute/v2/7e98f4488d464c9d827b659493f5b96c
Identity	http://3.144.76.134/identity
Image	http://3.144.76.134/image
Load Balancer	http://3.144.76.134/placeholder/load-balancer/v2.0
Network	http://3.144.76.134/networking
Placement	http://3.144.76.134/placement
Share	http://3.144.76.134/share/v1/7e98f4488d464c9d827b659493f5b96c
Shared File System	http://3.144.76.134/share/v2
Sharev2	http://3.144.76.134/share/v2
Sharev2_Legacy	http://3.144.76.134/share/v2/7e98f4488d464c9d827b659493f5b96c

12 항목 표시

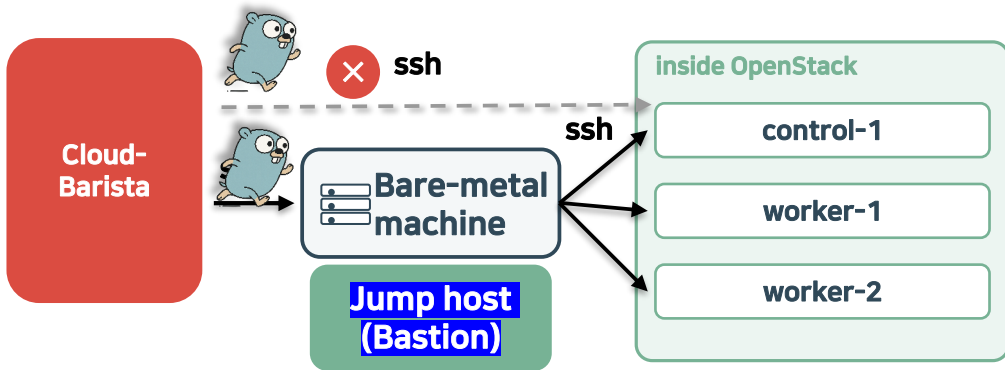
Lab: a cloud on a cloud on a cloud



This is an experiment, not an architecture.

Getting Kubernetes onto the three VMs

① the three VMs have no public address



one remote command · labelSelector · hopping through one machine

② which installer?



kubeadm

the official way, node by node

this lab



k3s

one small binary

edge · tiny



Kubespray

Ansible, many nodes

production

no managed cluster here

control plane first, then the workers join — all through the jump host

Get Status

Get Access ▾

Remote Command

Remote File Transfer

Snapshot Management

ScaleOut NodeGroup

Update Firewall

Control L4 NLB

Delete Infra

Resource Control

Release Resources

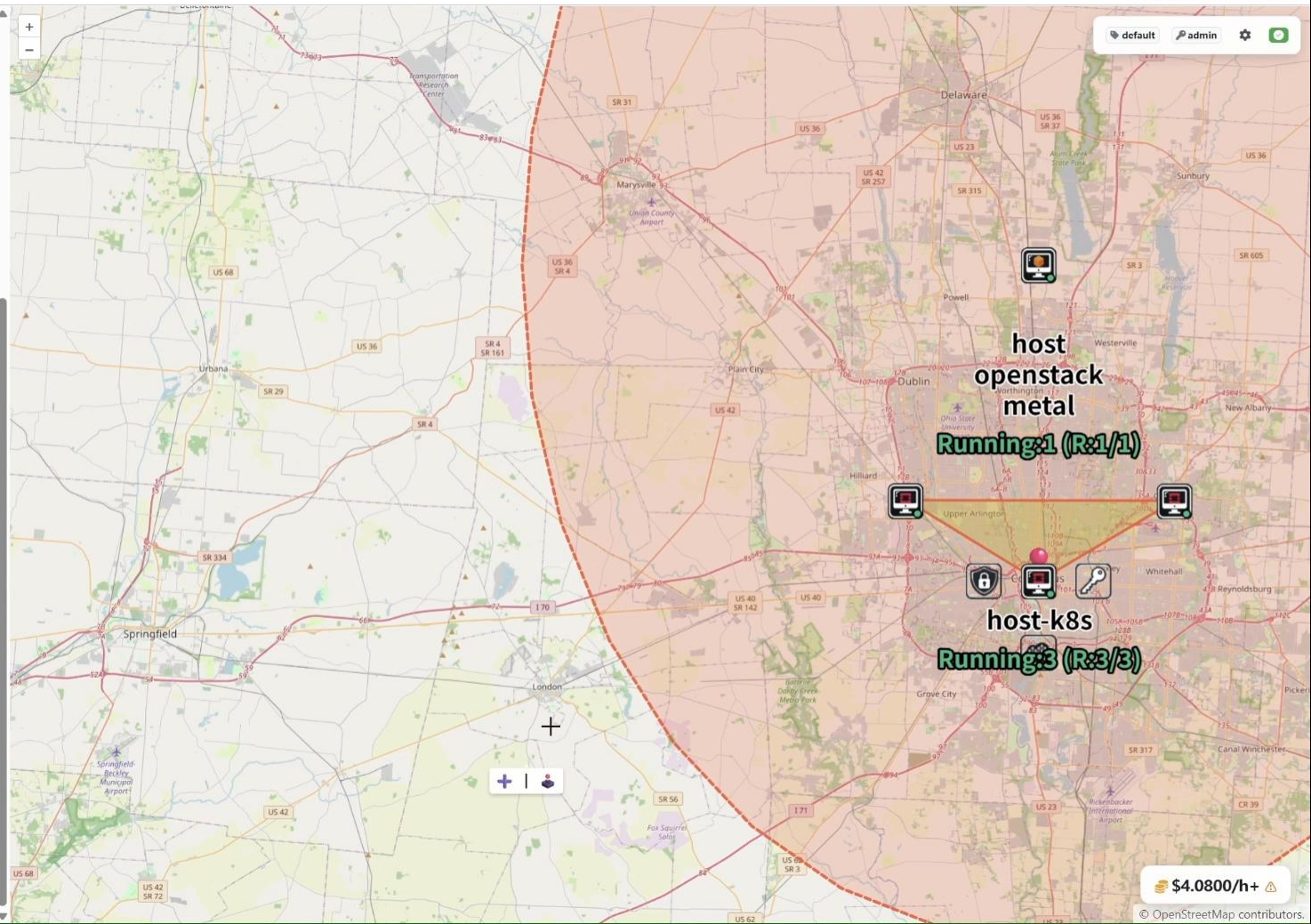
Resource Control ▾

VNet default-shared-aws-us ▾

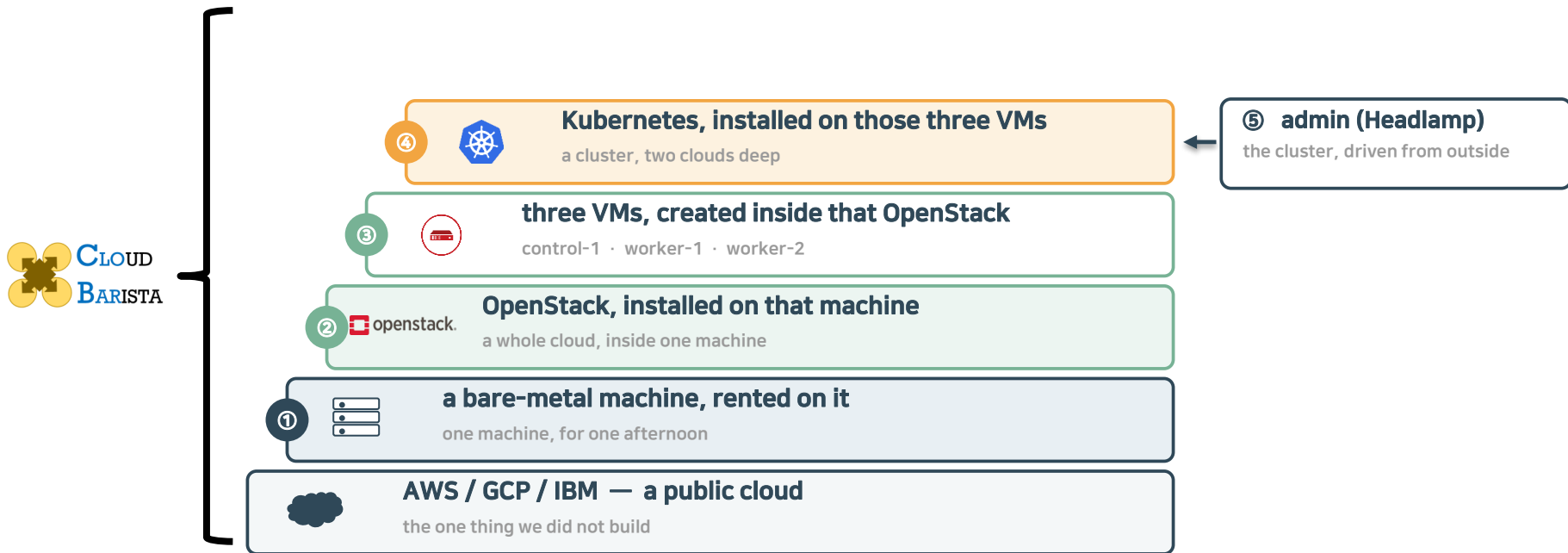
Security host-openstack-metal- ▾

SSHKey host-openstack-metal- ▾

Connect ▾

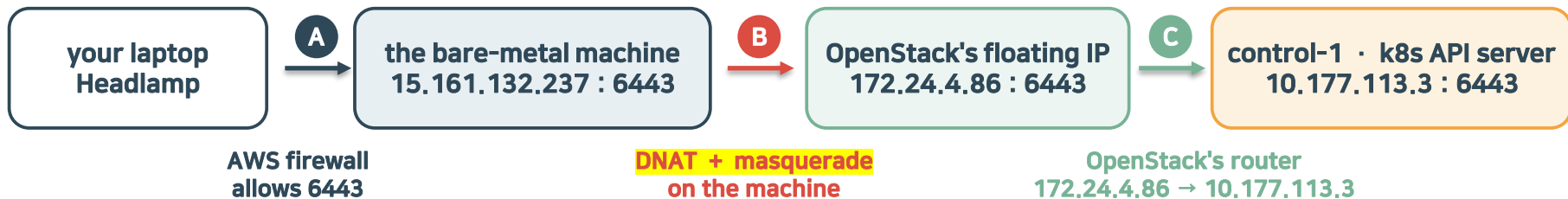


Lab: a cloud on a cloud on a cloud



This is an experiment, not an architecture.

Reaching the cluster from your laptop



what happens at

TO: 15.161.132.237
FROM: your laptop



TO: 172.24.4.86
FROM: the machine

DNAT (modify destination)

masquerade (modify sender)

forget the sender and the reply never comes home

kubeconfig — one line matters

~~server: https://10.177.113.3:6443~~

true inside the cluster only

server: https://15.161.132.237:6443

and the certificate has to list it too (SAN)

The only public address is the machine's. Everything after it is a rewrite.

localhost:1324

MapTopolNetBoardAPI

ProvisionControl

MC-Infra Control

Req Log

Select ID

Infra ID

host-openstack-metal

Node ID

g1-1 (g1)

Pub IP

3.144.76.134 (g1-1, g1)

Control Infra

Get Status

Get Access

Remote Command

Remote File Transfer

Snapshot Management

ScaleOut NodeGroup

Update Firewall

Control L4 NLB

Delete Infra

defaultadmin

host openstack metal

Running:1 (R:1/1)

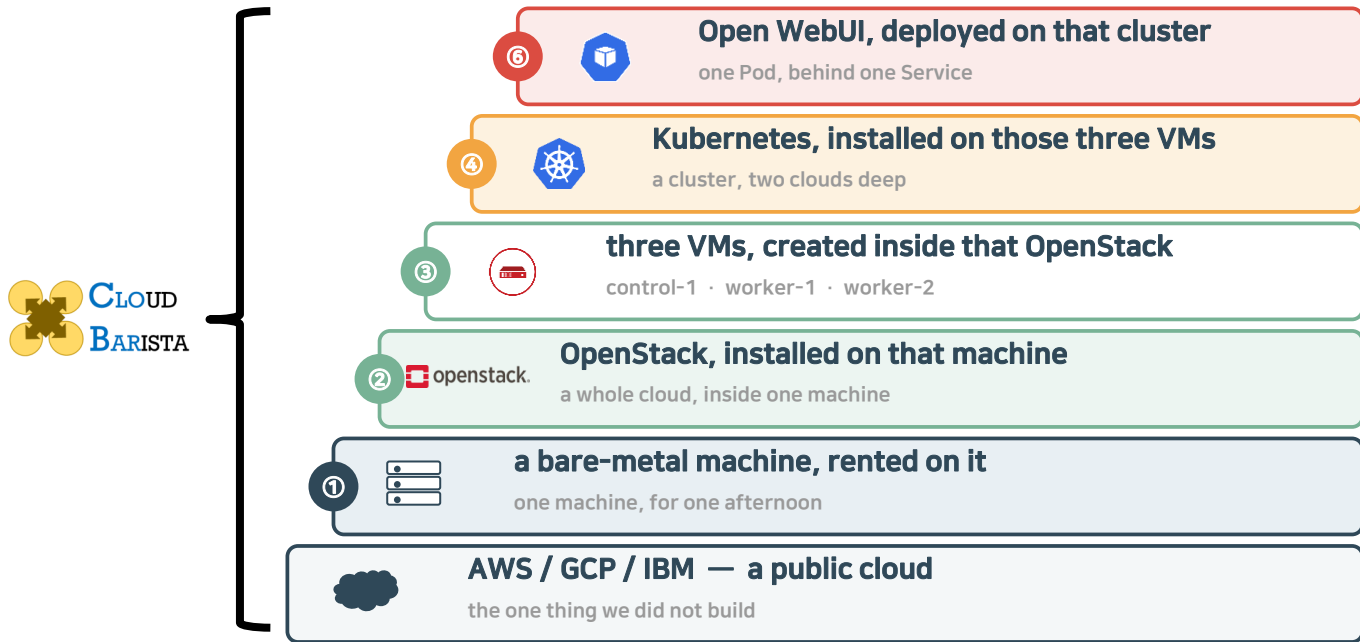
host-k8s

Running:3 (R:3/3)

\$4.02 done

OpenStreetMap contributors

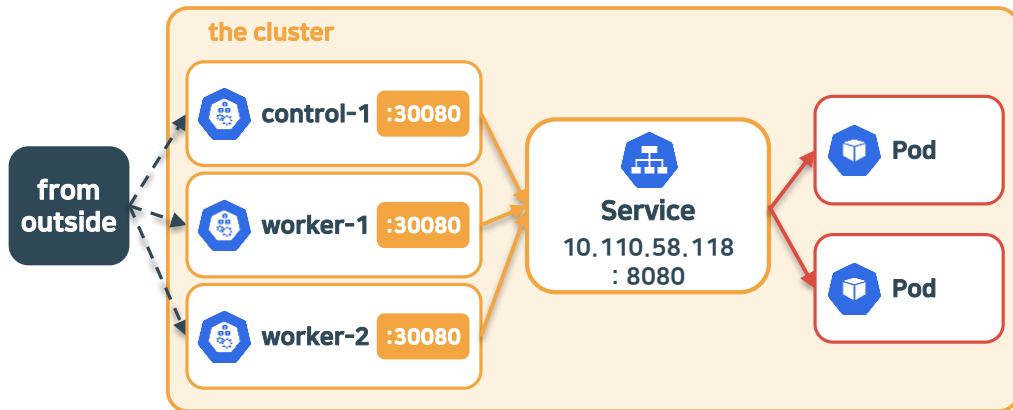
Lab: a cloud on a cloud on a cloud



This is an experiment, not an architecture.

Open WebUI on the cluster

① from a port on a node to the container



any node's :30080 works — even a node the Pod is not on

② why a NodePort?

ClusterIP

inside the cluster only

NodePort

a door on every node

★ this lab

LoadBalancer

the cloud hands you one

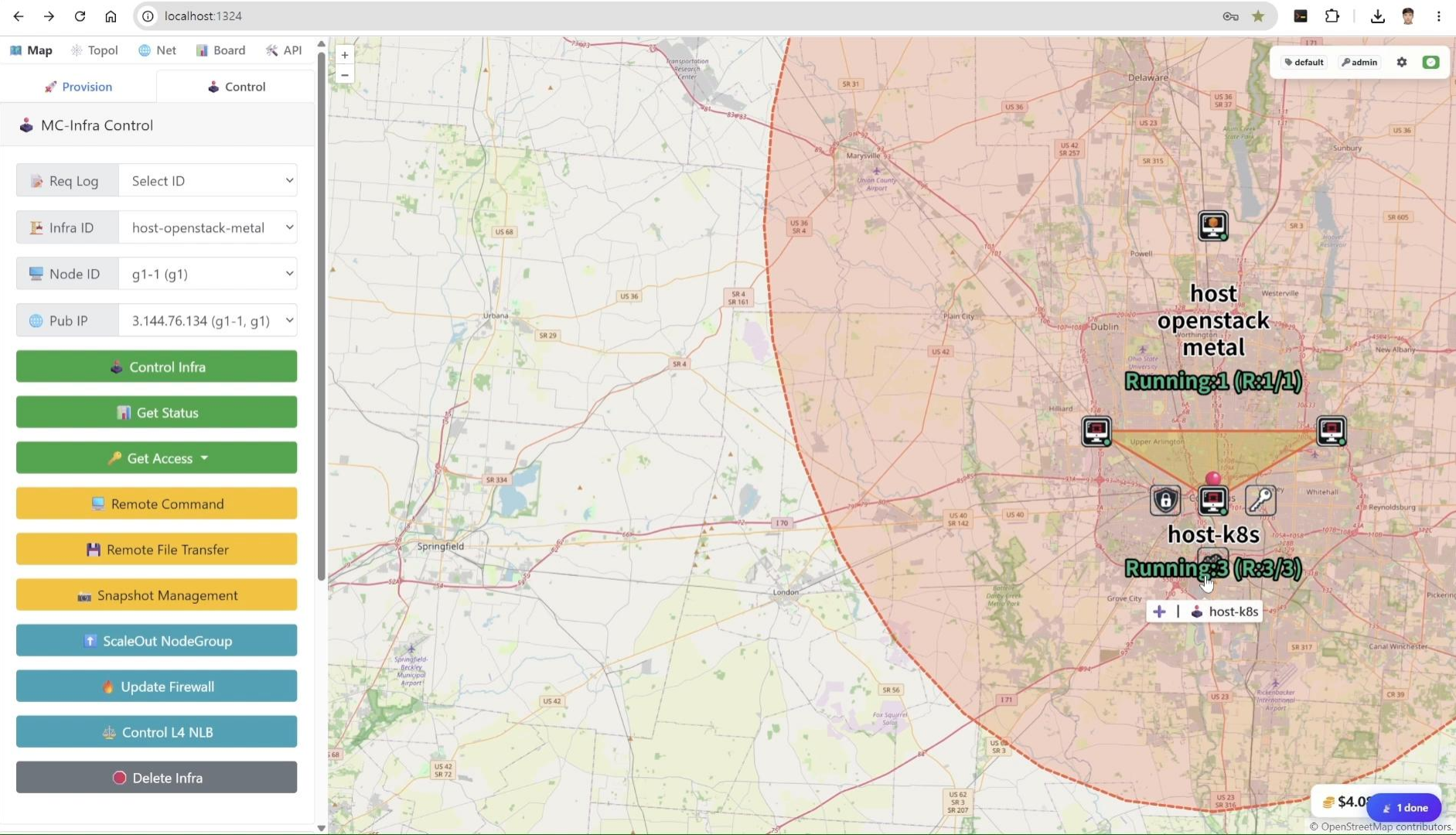
no cloud to ask

Ingress

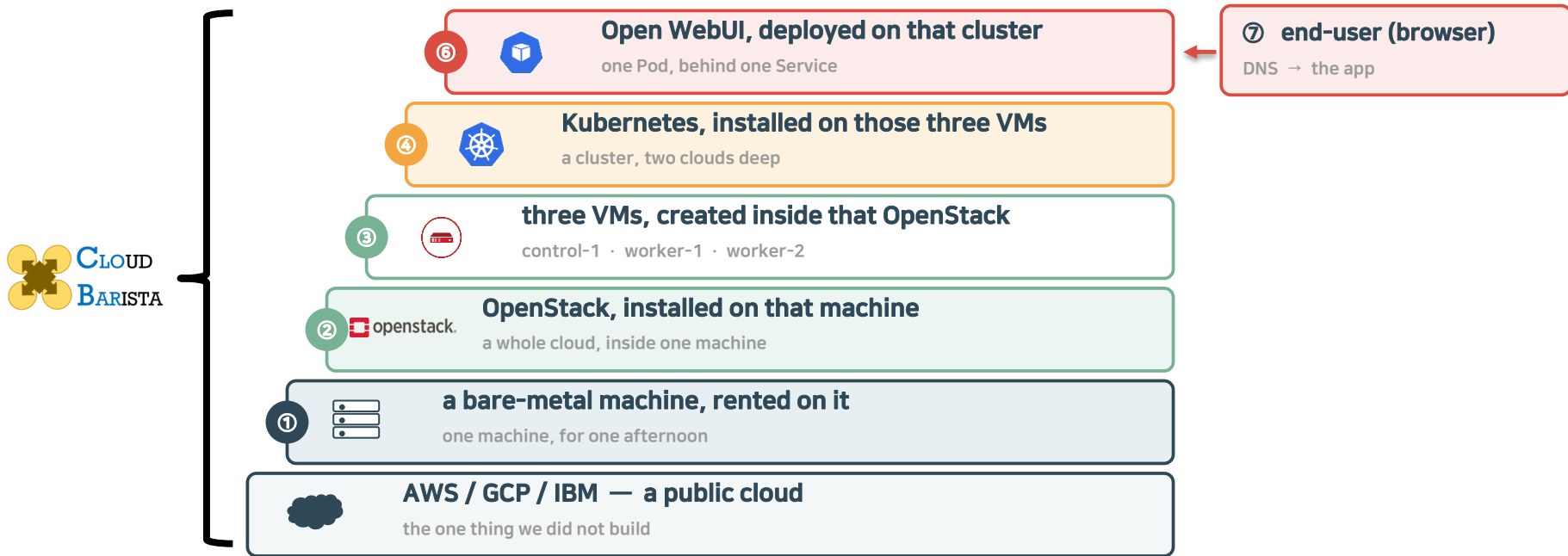
one name, many apps

needs a controller

NodePort opens the same door on every node. Kubernetes finds the Pod.

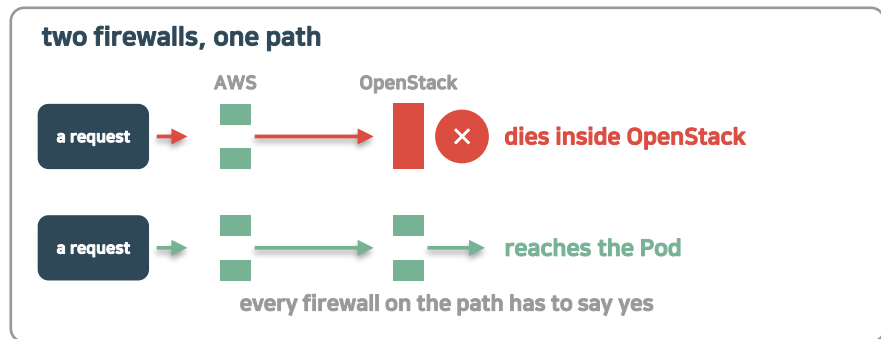
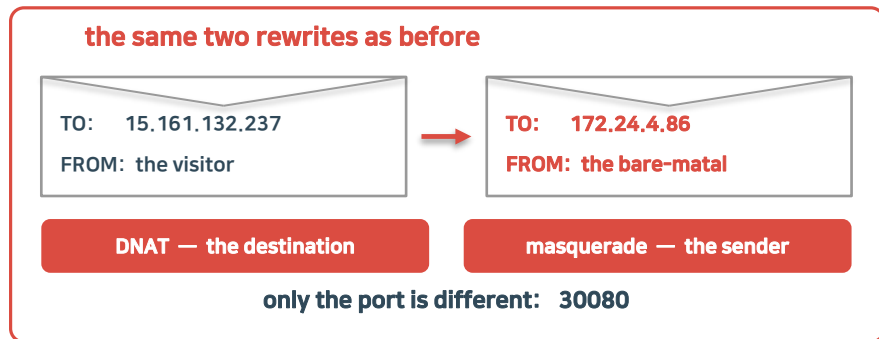
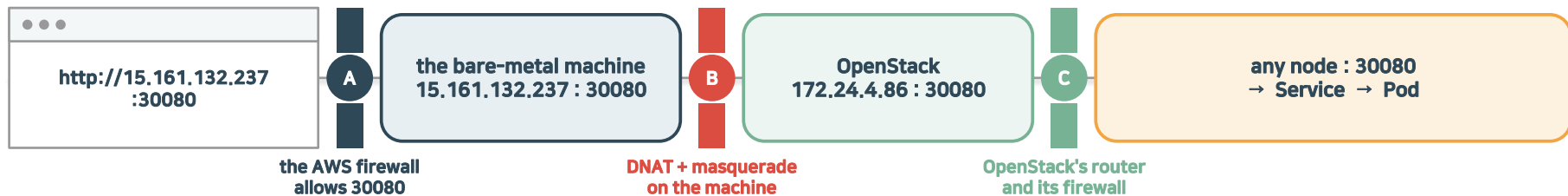


Lab: a cloud on a cloud on a cloud



This is an experiment, not an architecture.

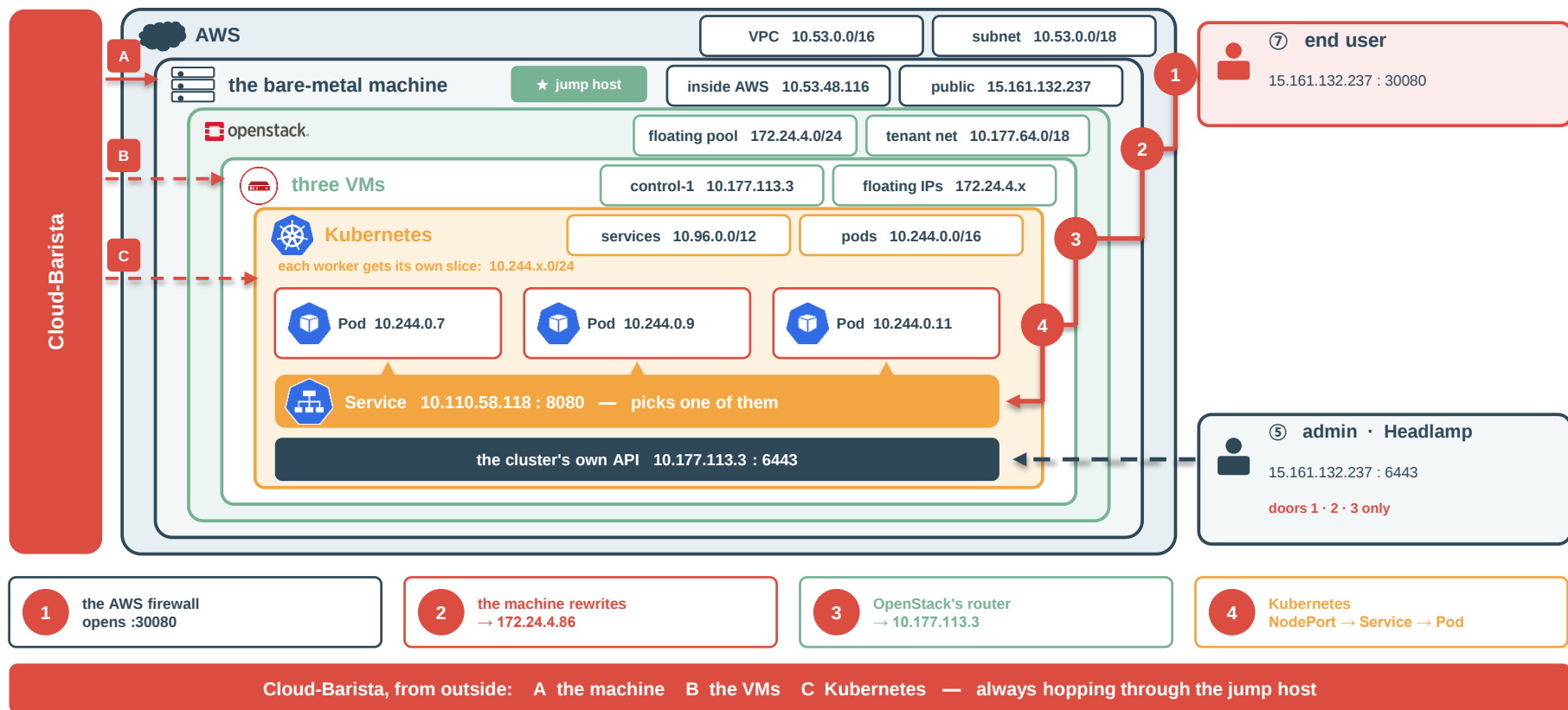
Letting end users in



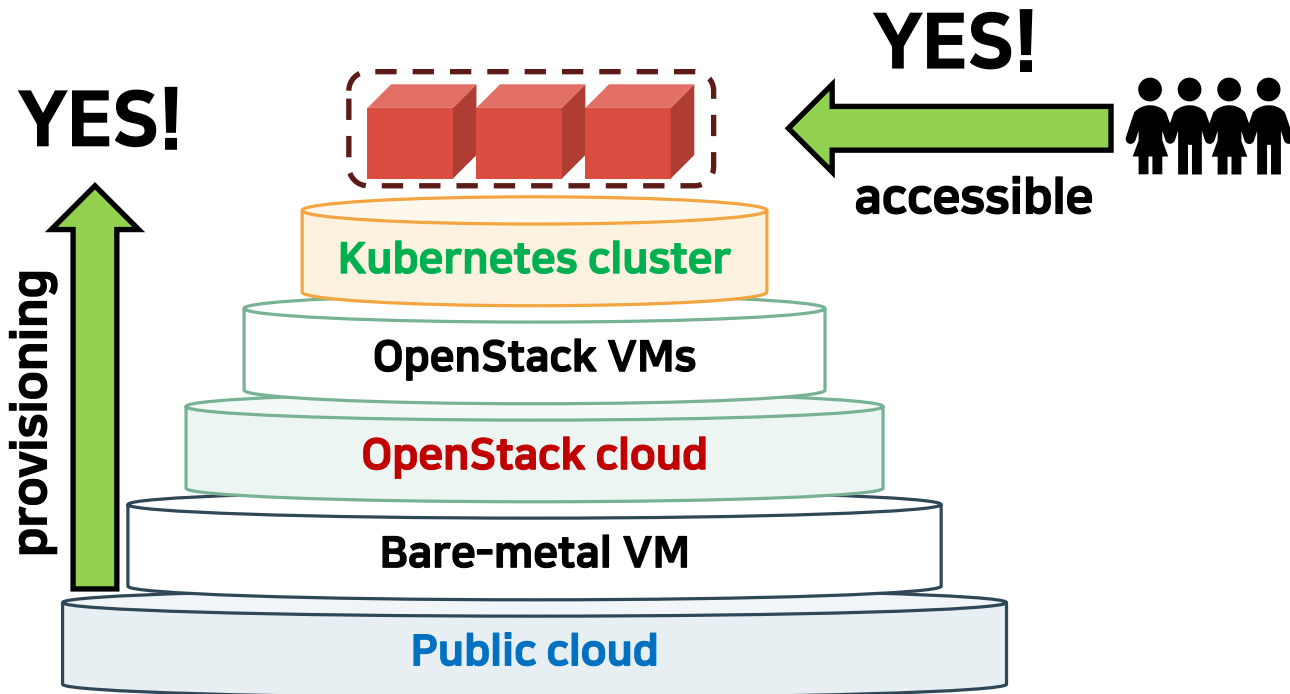
`http://15.161.132.237:30080` — an ordinary URL, four rewrites deep.



Every address, every door



Yes: a **cloud** on a **cloud** on a **cloud**



If any of this was interesting

CB-Tumblebug (Multi-Cloud Infra Management) 🙋

go report **A+** go **76.4%** go.mod **v1.23.0** repo size **35.1 MiB** **GO** reference Visual Codebase API Doc Swagger

license **Apache-2.0** release **v0.10.0** release(dev) **v0.10.9** build **passing** Slack **SIG-TB**

all contributors **49**

github.com/cloud-barista/cb-tumblebug

★ Open Source Together ★

- Your Feedback and Contribution
- Keep This Open Source Alive!

CLOUD NATIVE
COMPUTING FOUNDATION

About Projects Training Community Blog & News [Join](#) 🔍

SANDBOX PROJECTS

The CNCF Sandbox is the entry point for early stage projects.

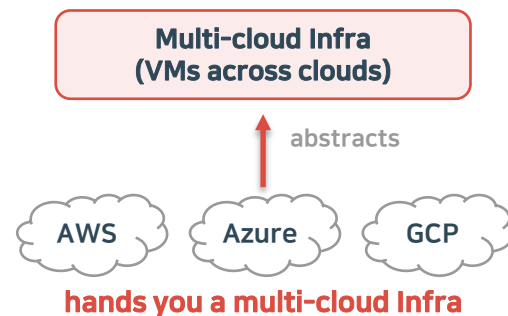
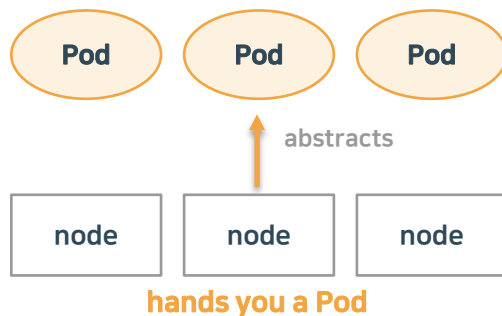
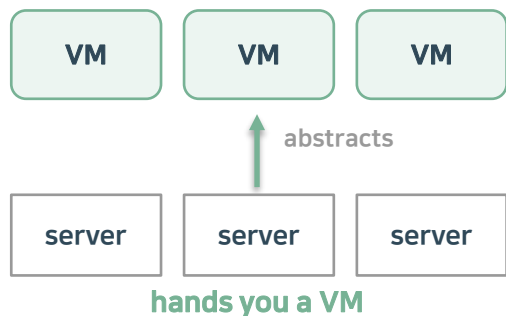
Our Sandbox Projects



planning to apply

Project Name	Icon	Description
Aeraki Mesh		Service Mesh
Akri		Automation & Configuration
Antrea		Cloud Native Network
Armada		Scheduling & Orchestration
Athenz		Key Management
Atlantis		Automation & Configuration
Bank-Vaults		Security & Compliance
BFE		Service Proxy
BPFman		Security & Compliance
Cadence		Automation & Configuration
Capsule		Scheduling & Orchestration
Carina		Cloud Native Storage
Cartography		Security & Compliance
Carvel		Application Definition & Image Build

One day. Three open sources, **in one Home**



OpenStack

Kubernetes

Cloud-Barista

Family of CNCF & LF



**OPEN
KOREA**

SOURCE SUMMIT

THE LINUX FOUNDATION

Thank you!



seokho-son



seokho-son

