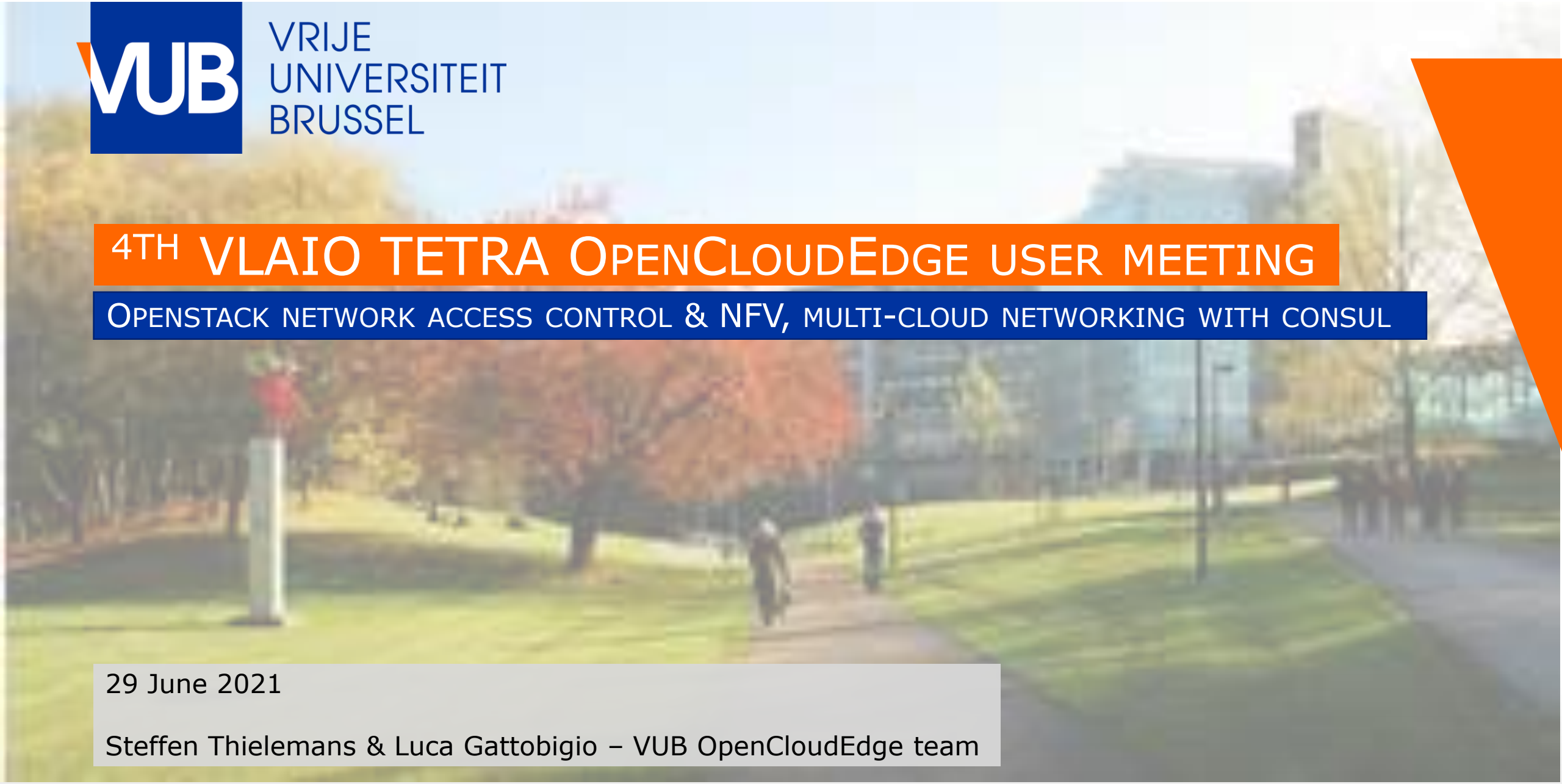


VRIJE UNIVERSITEIT BRUSSEL

4TH VLAIO TETRA OPENCLOUDEDGE USER MEETING

OPENSTACK NETWORK ACCESS CONTROL & NFV, MULTI-CLOUD NETWORKING WITH CONSUL

29 June 2021

Steffen Thielemans & Luca Gattobigio – VUB OpenCloudEdge team

AGENDA

- **Recap from previous user committee meeting**

- **Multi-cloud networking via Consul** (20 min)

- **OpenStack Role-Based Access Control Policies on networking & NFV** (20 min)

- **Other news, future work, Q&A**

RECAP PREVIOUS MEETING

- Our on-premise OpenStack and Kubernetes infrastructures are operational
  - OpenStack deployed & upgraded via kolla-ansible.
  - **NFV-based approach to manage tenant's external network accessibility**

- Multi-cloud deployments using Terraform
  - **We introduce multi-cloud networking via Consul**

- Edge
  - On-premise IoT project where we will integrate resource-constrained devices via lightweight **edge nodes**

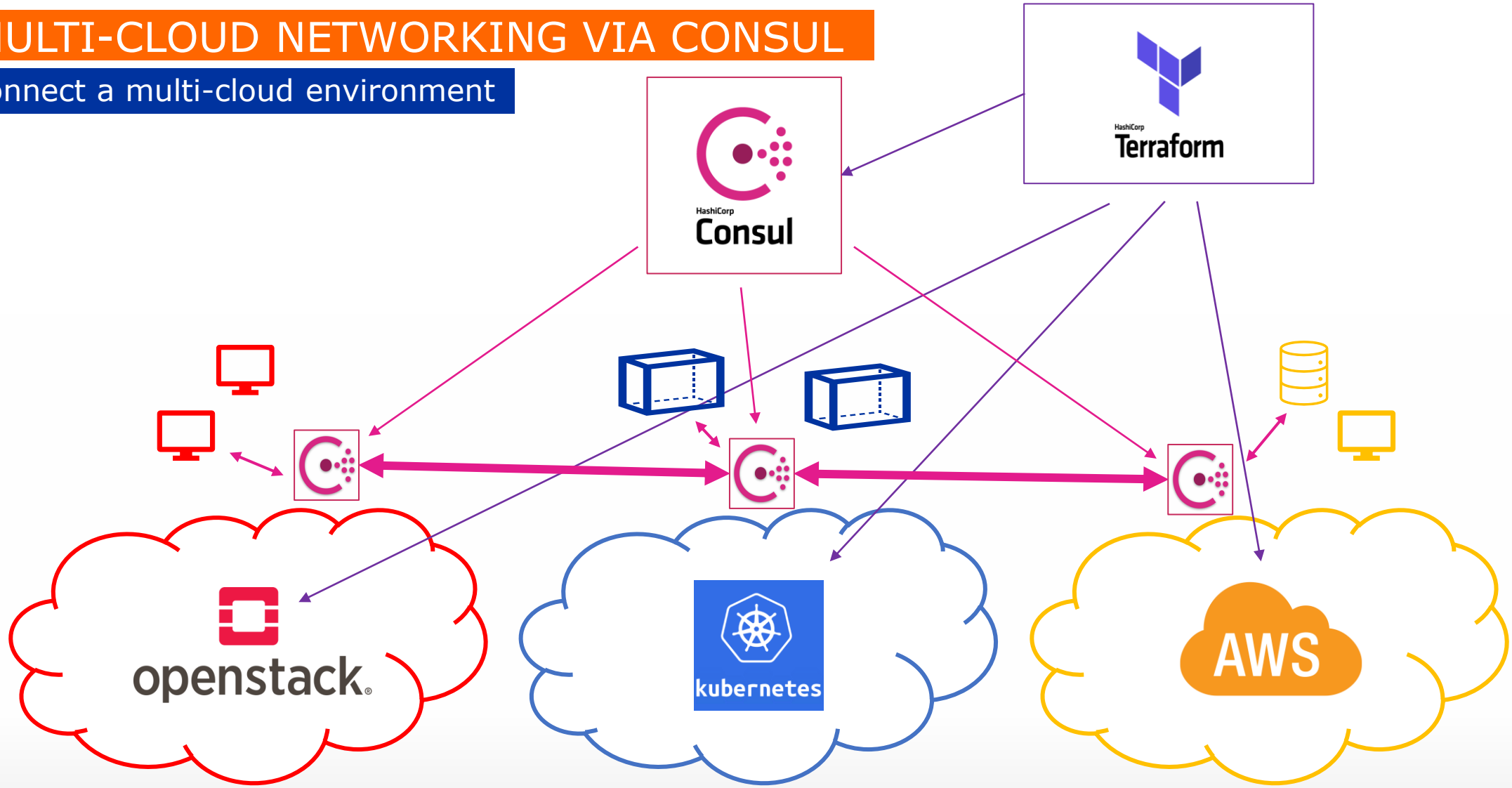
- Always open to suggestions regarding use-cases and desired future work

AGENDA

- **Recap from previous user committee meeting & news**
- **Multi-cloud networking via Consul** (20 min)
- **OpenStack Role-Based Access Control Policies on networking & NFV** (20 min)
- **Other news, future work, Q&A**

MULTI-CLOUD NETWORKING VIA CONSUL

Connect a multi-cloud environment



MULTI-CLOUD NETWORKING VIA CONSUL

What is Consul?

- Opensource tool for deploying a service mesh
- Multiple features: Centralized registry, service discovery, health checks, zero trust network, load-balancer, Key-Value store...
- Constantly updated and supported by Hashicorp and the community
- Part of Hashicorp's “Cloud-oriented” suite of tools

MULTI-CLOUD NETWORKING VIA CONSUL

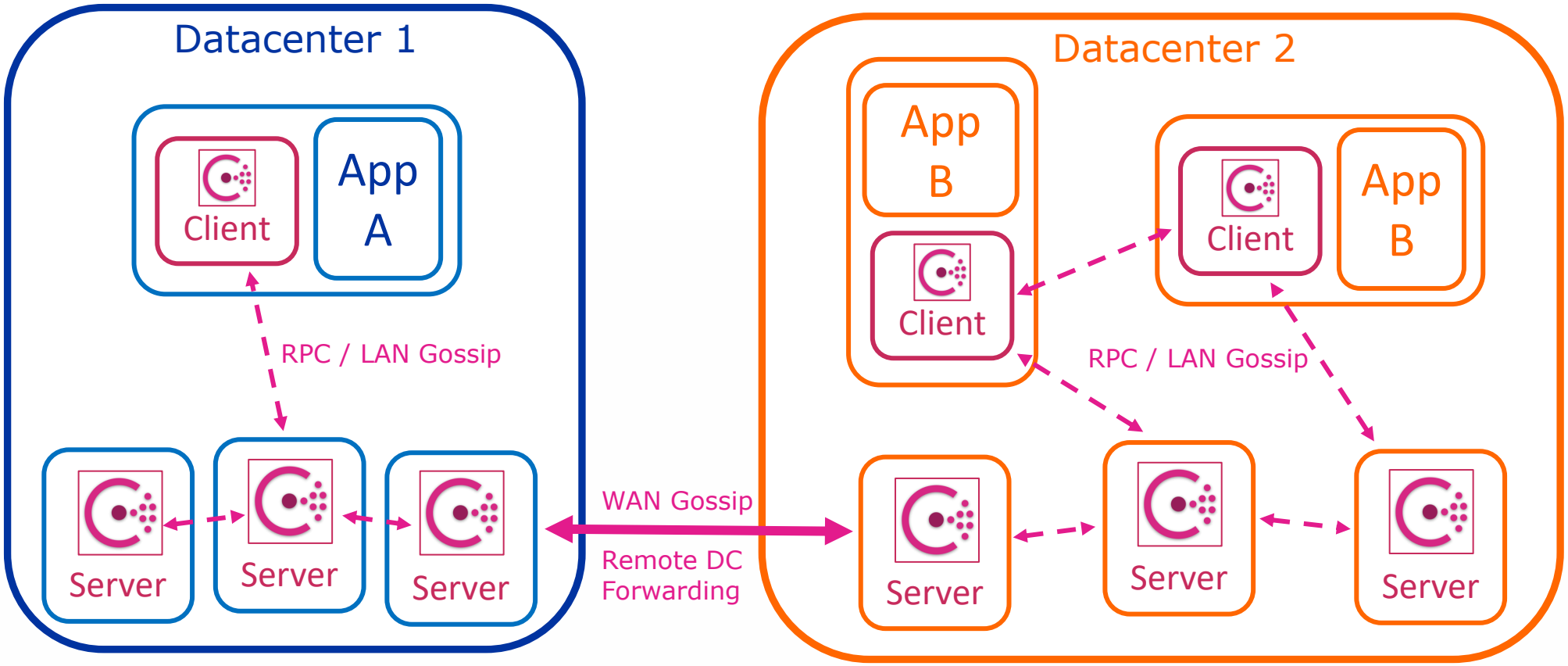
Why Consul?

Consul can be compared with other service mesh tools (Istio, Ambassador, Linkerd)

- HashiCorp (HCL language, Terraform integrations)
- Works with Envoy proxy
- Can be easily deployed in both VMs and containers
- Perfect for multi-cloud environment

MULTI-CLOUD NETWORKING VIA CONSUL

Architecture

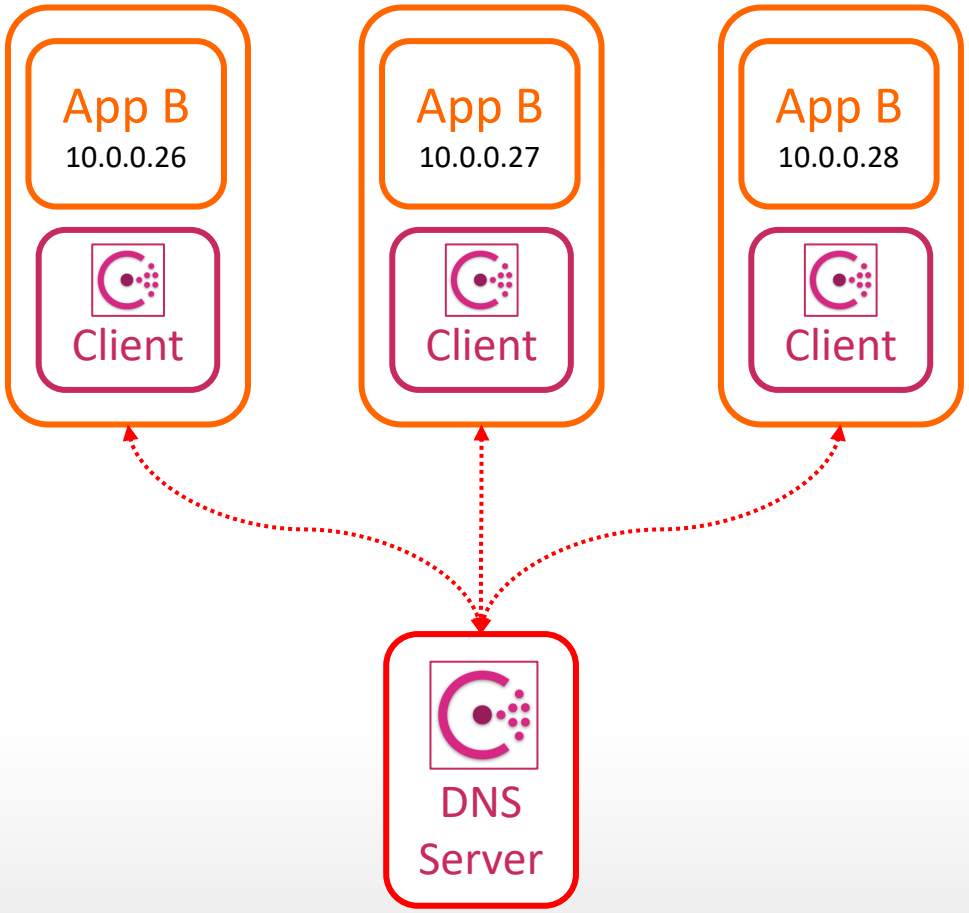


Name	Datacenter	IP	Status
App A	1	192.168.1.45	Alive
App B	2	10.0.0.54, 10.0.0.98	Alive, Alive

Name	Datacenter	IP	Status
App A	1	192.168.1.45	Alive
App B	2	10.0.0.54, 10.0.0.98	Alive, Alive

MULTI-CLOUD NETWORKING VIA CONSUL

Service discovery and health checks

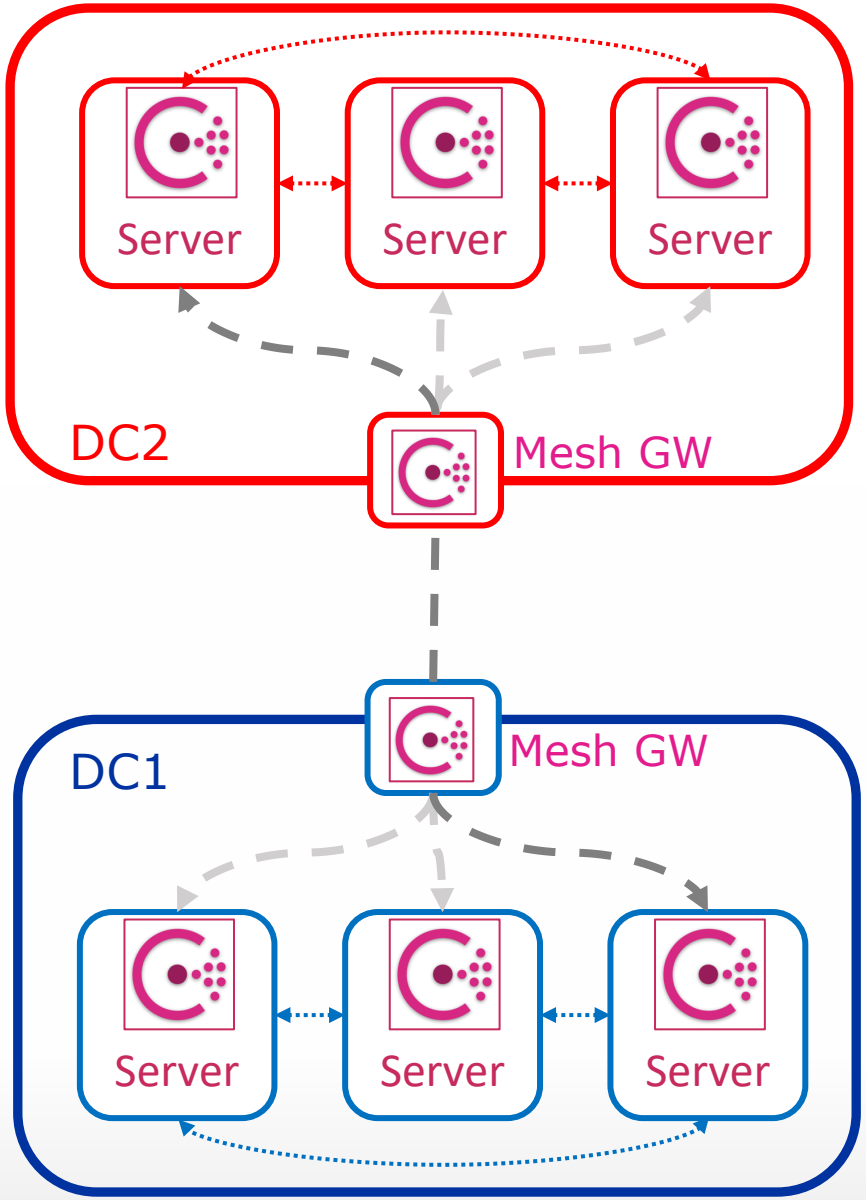


- Centralized Service Registry
- Real-time Health Checks
- Auto-join feature

MULTI-CLOUD NETWORKING VIA CONSUL

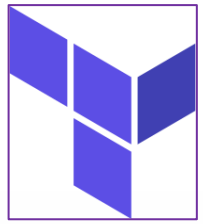
Multi-cloud Service Mesh

- Multi-cloud service mesh
- Mesh gateways
- Datacenter federation
- mTLS (mutual Transport Layer Security)



MULTI-CLOUD NETWORKING VIA CONSUL

Datacenter federation: Software used



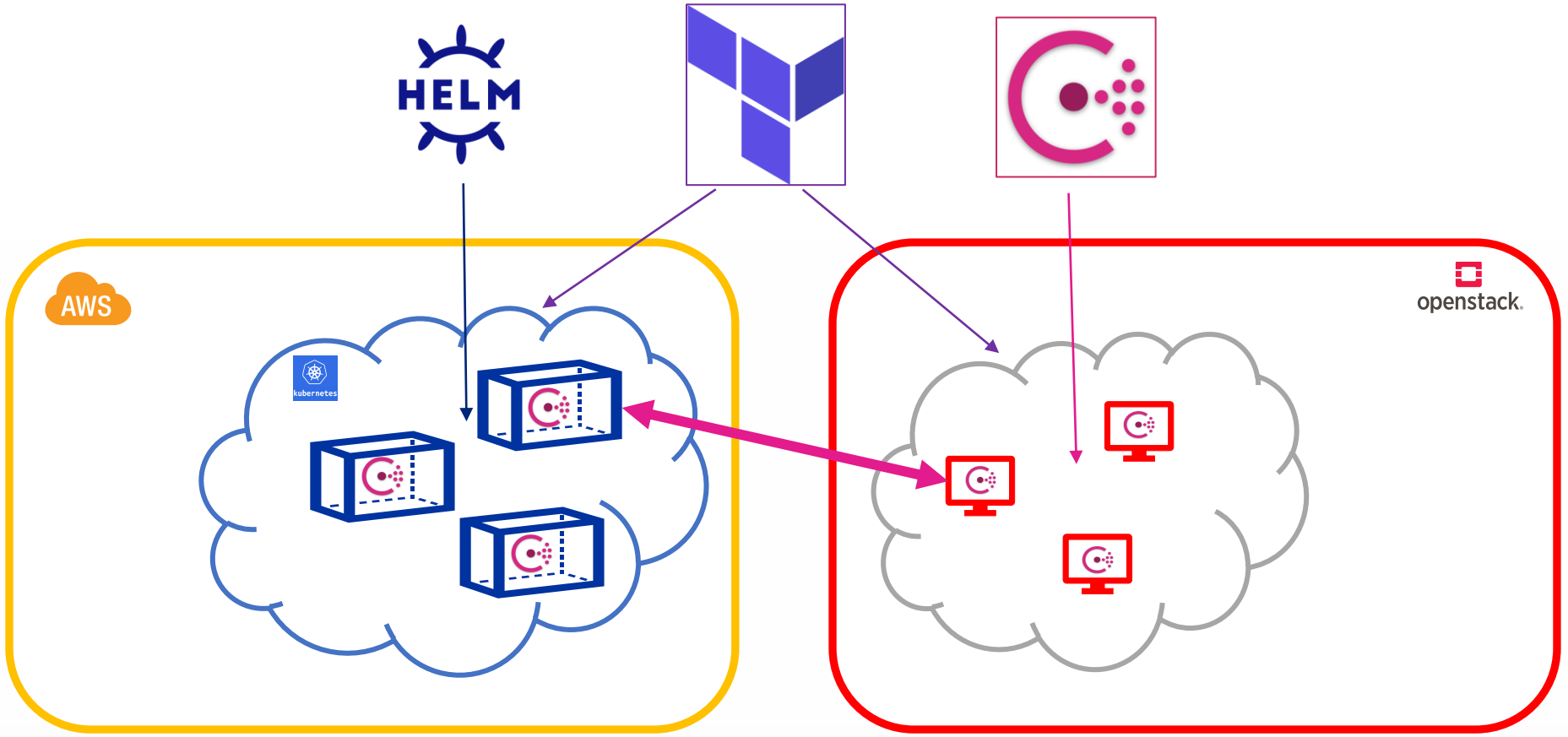
Terraform – Deploying the resources with IaC

Consul CLI and UI – Running on VMs

Deploying Helm charts in Kubernetes clusters

MULTI-CLOUD NETWORKING VIA CONSUL

Datacenter federation



MULTI-CLOUD NETWORKING VIA CONSUL

Datacenter federation: Helm-chart.yaml

```
global:
  datacenter: dc-aws

  federation:
    enabled: true
    createFederationSecret: true

  tls:
    enabled: true

server:
  replicas: 1

ui:
  enabled: true
  service:
    type: 'LoadBalancer'
```

```
meshGateway:
  enabled: true
  replicas: 1
  service:
    type: 'LoadBalancer’

connectInject:
  enabled: true
```

MULTI-CLOUD NETWORKING VIA CONSUL

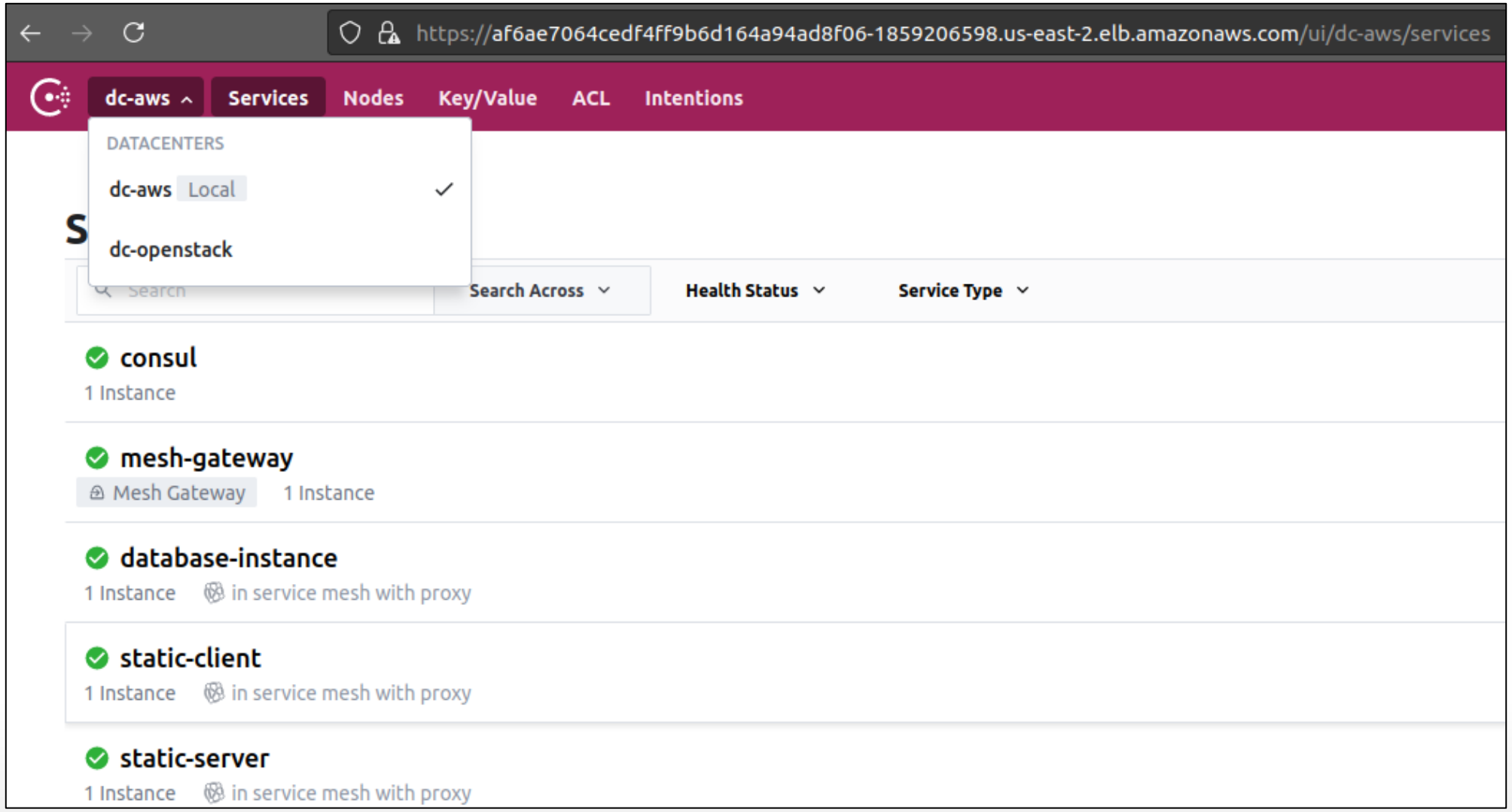
Datacenter federation: Consul-server.hcl

```
datacenter = "dc-os"

cert_file = "dc-os-server-consul-0.pem"
key_file = "dc-os-server-consul-0-key.pem"
ca_file = "consul-agent-ca.pem"

primary_gateways = ["10.20.28.173:443"]
primary_datacenter = "dc-aws"

server = true
bootstrap_expect = 1

advertise_addr = "{{GetInterfaceIP 'ens3' }}"
client_addr = "0.0.0.0"
```

```
ports {
    grpc = 8502
    https = 8501
    http = 8500
}

enable_central_service_config = true

ui_config {
    enabled = true
}

connect {
    enabled = true
    enable_mesh_gateway_wan_federation = true
}
```

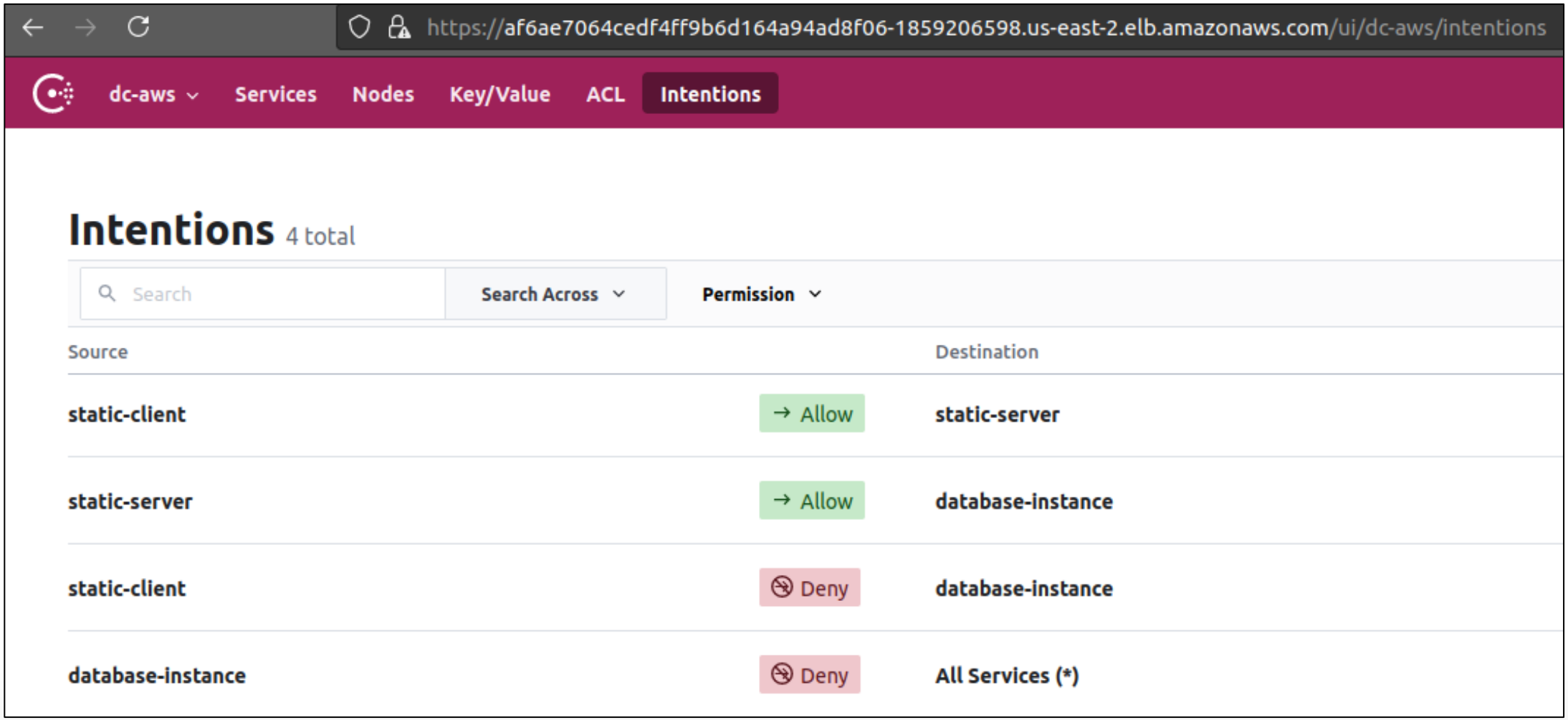
MULTI-CLOUD NETWORKING VIA CONSUL

Datacenter federation: Consul UI



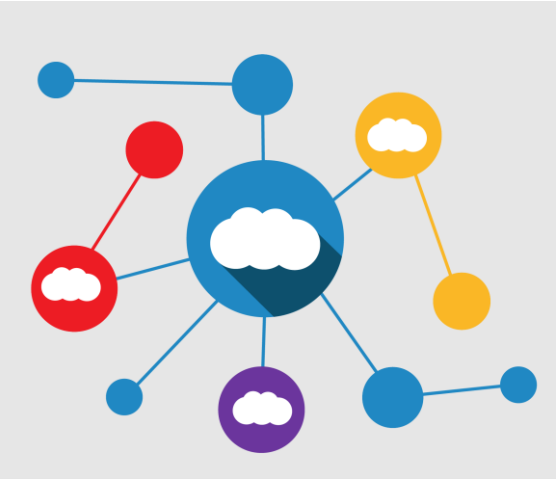
MULTI-CLOUD NETWORKING VIA CONSUL

Datacenter federation: Consul UI



MULTI-CLOUD NETWORKING VIA CONSUL

Future steps



- Consul to connect our infrastructure (Hybrid cloud/Multi-cloud)

- Infrastructure for all the use cases and future projects that require a multi-cloud environment

AGENDA

- **Recap from previous user committee meeting & news**
- **Multi-cloud networking via Consul** (20 min)
- **OpenStack Role-Based Access Control Policies on networking & NFV** (20 min)
- **Other news, future work, Q&A**

OPENSTACK ROLE-BASED ACCESS CONTROL ON NETWORK RESOURCES

INTRODUCTION

What we want: ***Provide* OpenStack access to external parties**

- For OpenCloudEdge user committee members (restricted access)
- For student projects (restricted access)
- For our own researches (complete access)

Problem: **on-premise OpenStack servers deployed within ETRO's Intranet**

- OpenStack's *external network* is VUB-ETRO's Intranet
- No OpenStack way to restrict access on an external network as cloud provider
  - OpenStack *security groups* are managed by tenant, not cloud provider

This means: **All tenants have access to the complete ETRO Intranet**

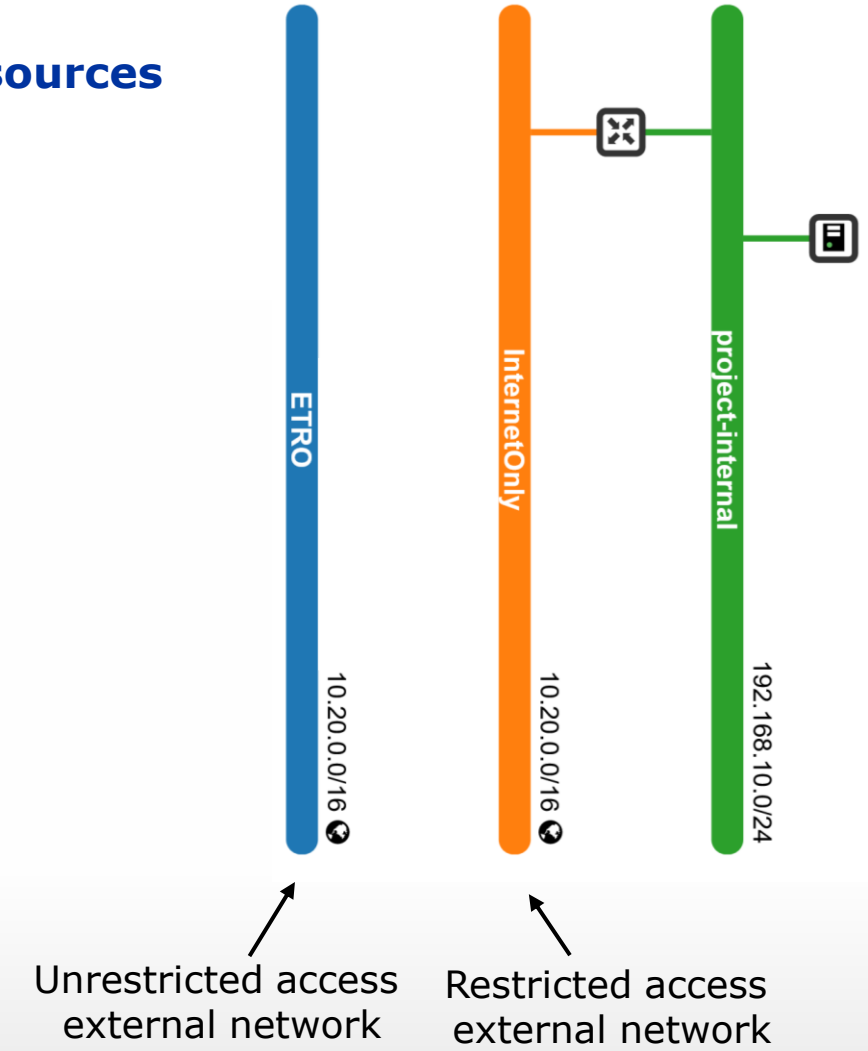
- **This is a major security risk**

OPENSTACK ROLE-BASED ACCESS CONTROL ON NETWORK RESOURCES

GOAL

Manage OpenStack tenant access to the *external network* resources

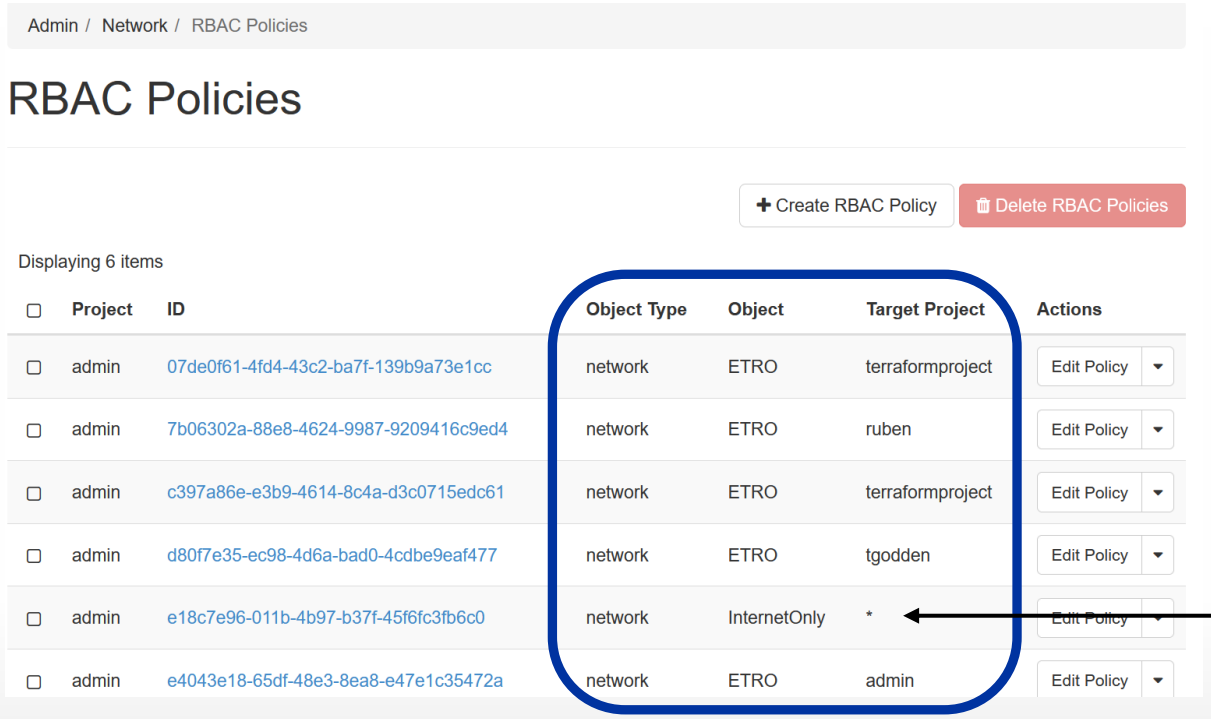
- Set up multiple external networks
  - ***InternetOnly*** restricted access *external network*
    - Available in all OpenStack projects
    - Intended for OpenCloudEdge partners & students
  - ***ETRO*** unrestricted access *external network*
    - Only available in approved OpenStack projects
    - Intended for internal projects by VUB Researchers



OPENSTACK ROLE-BASED ACCESS CONTROL ON NETWORK RESOURCES

OPENSTACK RBAC ON NETWORKS

- Control access to OpenStack **external** & **shared** networks
- Applied on **project**, **resource** and **subject** (*external* / *shared* network) basis

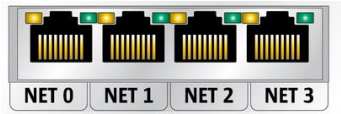


- Unrestricted ***ETRO*** external network available on approved projects
- Restricted ***InternetOnly*** external network available on all projects (wildcard)

OPENSTACK ROLE-BASED ACCESS CONTROL ON NETWORK RESOURCES

OPENSTACK EXTERNAL NETWORKS

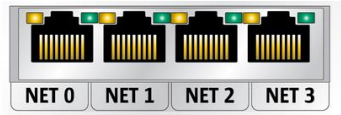
- OpenStack external networks are connected to actual networks



ETRO InternetOnly

- How to create ***ETRO*** and ***InternetOnly*** networks?

1. Multiple physical network interfaces
   - Requires changes to the physical network (routing & firewalling)

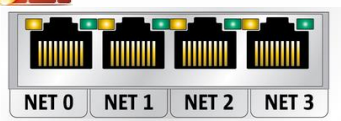


VLAN 1: ETRO
VLAN 2: InternetOnly

2. Single physical network interface with **VLAN** tagging
   - Requires changes to the physical network (routing & firewalling)

3. Single physical network interface & **Network Function Virtualization (NFV)**
   - All changes are constrained to software running on our servers
   - NFV firewall to restrict the ***InternetOnly*** network

NFV on our servers



ETRO

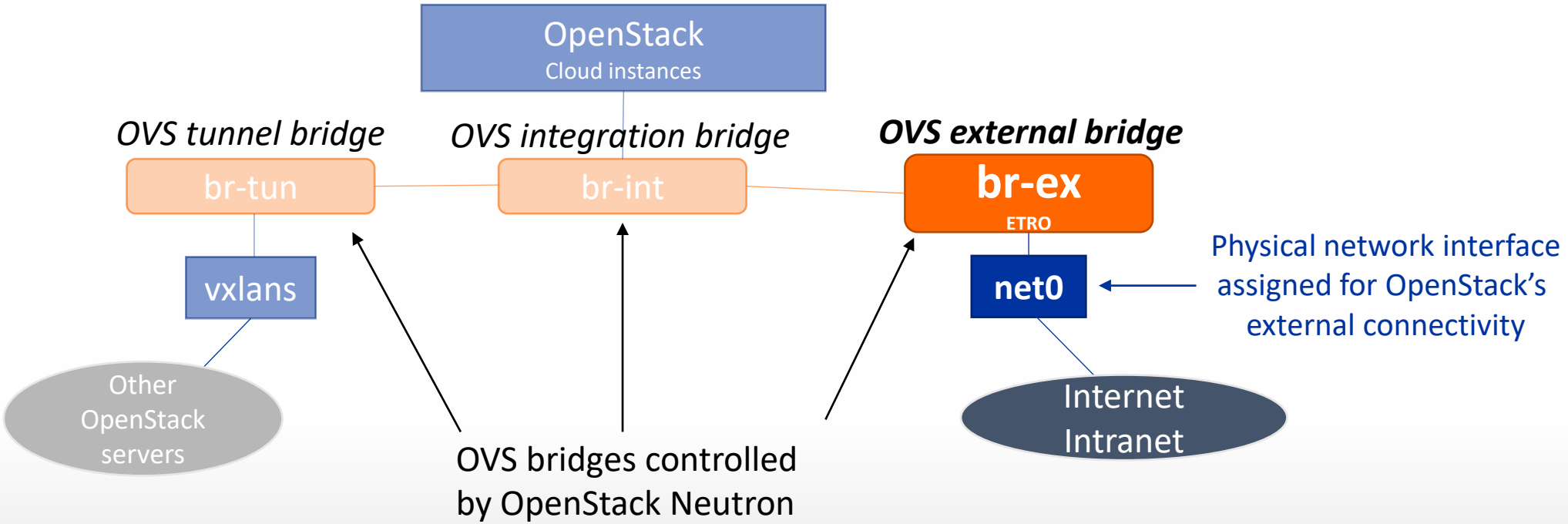
- For our project, we want everything constrained within our own infrastructure (=servers)

OPENSTACK ROLE-BASED ACCESS CONTROL ON NETWORK RESOURCES

A LOOK AT NEUTRON'S NETWORKING COMPONENTS

Neutron relies on 3 *open vSwitch* (**OVS**) bridges: *br-tun*, *br-int* & ***br-ex***

- ***br-ex*** provides external network connectivity to the OpenStack cloud instances



OPENSTACK ROLE-BASED ACCESS CONTROL ON NETWORK RESOURCES

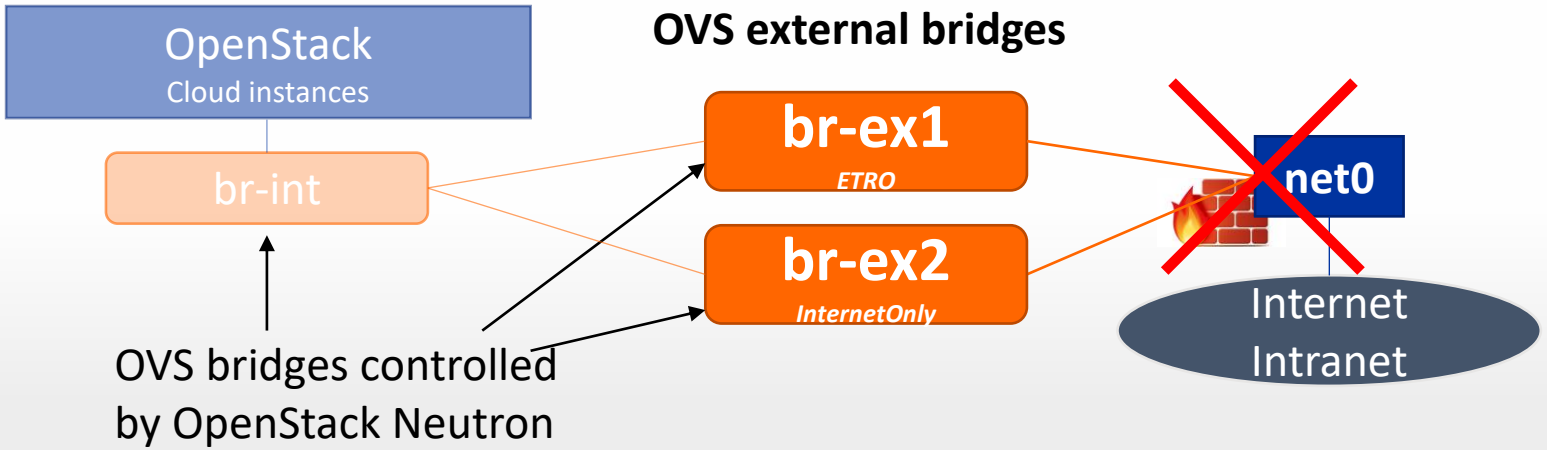
NFV FOR SPLITTING & MANAGING THE EXTERNAL NETWORK INTERFACE

Creation of additional OpenStack ***br-ex*** external networks

- Supported by Kolla-Ansible deployment tool (but barely documented)

Problem: A network interface (net0) can only be bound to a single OVS bridge at a time

- NFV can be used to split net0 into virtual networks & apply firewalling

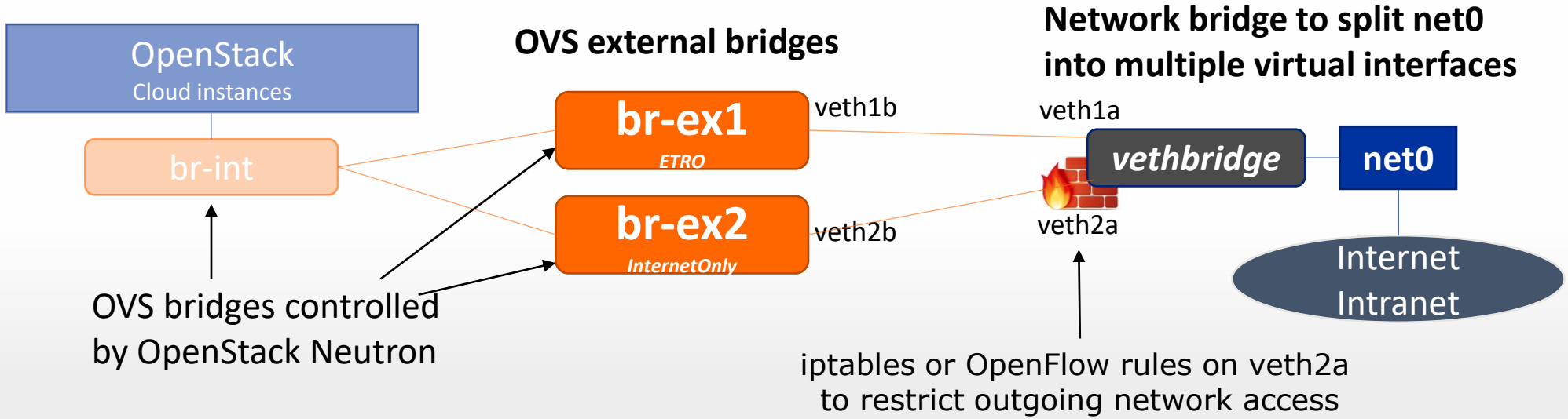


OPENSTACK ROLE-BASED ACCESS CONTROL ON NETWORK RESOURCES

NFV FOR SPLITTING & MANAGING THE EXTERNAL NETWORK INTERFACE

Splitting physical network interface net0 into multiple virtual network interfaces

- ***vethbridge*** deployed as **Linux Bridge** or **OVS bridge**
- **Virtual Ethernet interfaces** *veth1a* ↔ *veth1b* and *veth2a* ↔ *veth2b*
  - Similar to physical patch cables: what goes in on one side, comes out the other side
  - OVS *patch ports*
- Persistent *iptables/ebtables* **firewall rules** on the constrained veth2a interface
  - *OpenFlow rules*



OPENSTACK ROLE-BASED ACCESS CONTROL ON NETWORK RESOURCES

CONCLUSIONS

- OpenStack supports Role Based Access Control (RBAC) on network resources
  - Manage network access on per-project basis
- Network Function Virtualization (NFV) used create, control & manage the virtual networks
- Preliminary NFV experiment
  - *ETRO* and *Internet-Only* networks share the same (firewalled) subnets
    - Improvement: Different subnets and NFV firewall + routing
- OpenStack production environment recommendations:
  - Separated physical networks, VLANs or NFV on dedicated network device(s)

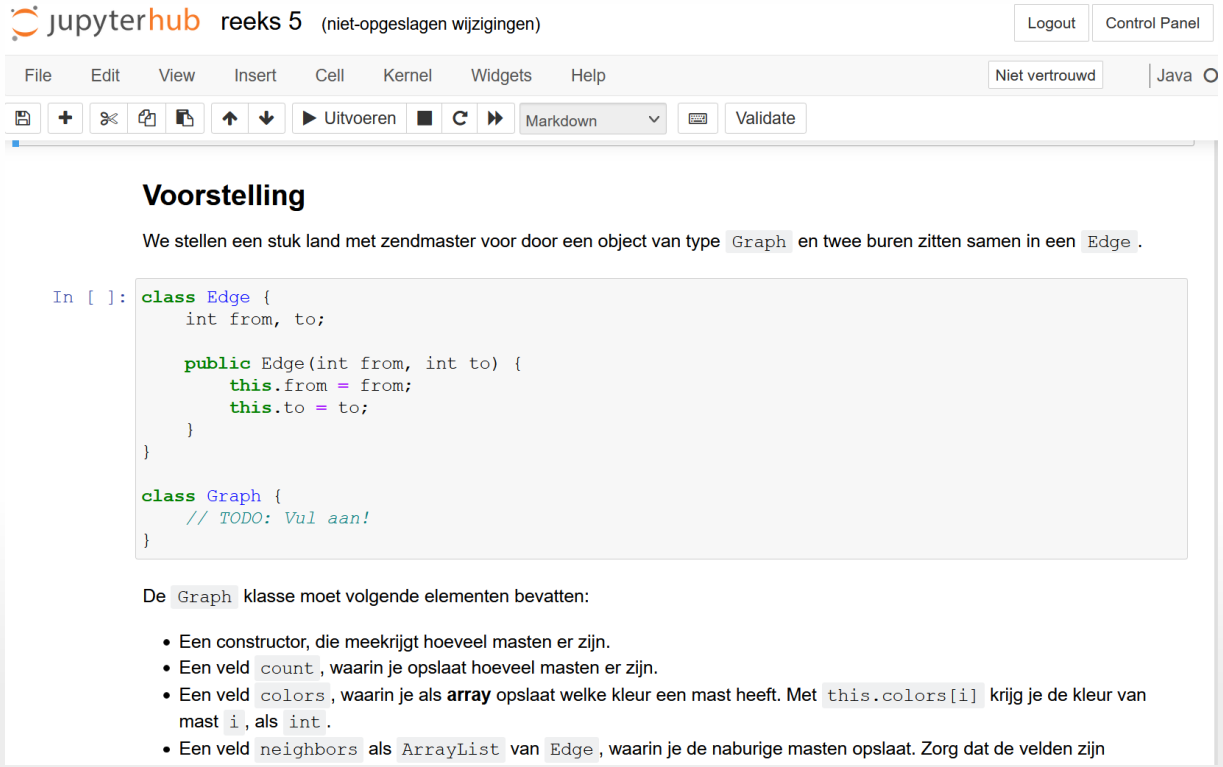
AGENDA

- **Recap from previous user committee meeting & news**
- **Multi-cloud networking via Consul** (20 min)
- **OpenStack Role-Based Access Control Policies on networking & NFV** (20 min)
- **Other news, future work, Q&A**

OTHER NEWS, FUTURE WORK, Q&A

USE-CASE: JUPYTERHUB

- ***JupyterHub*** SaaS used for the programming courses of 1st ba. ir. students
  - Each student receives an individual **web-based** Jupyter programming environment
  - Proves to be a great choice in these remote teaching/working times
  - Deployed on the OpenCloudEdge Kubernetes cluster



Pilot project in 2020-2021 academic year

- **positive evaluation**
- Extension planned for additional courses and students
- Additional server ordered & will be added to the cluster

OTHER NEWS, FUTURE WORK, Q&A

VLAAMS SUPERCOMPUTER CENTRUM (VSC)

Vlaams Supercomputer Centrum ***Tier-1 cloud***

- **OpenStack**
- https://www.vscentrum.be/cloud
- OpenStack hands-on tutorial provided (43 pages, English)

- Requested access to evaluate multi-cloud between on-premise & remote OpenStack clusters

OTHER NEWS, FUTURE WORK, Q&A

PLANS FOR THE UPCOMING MONTHS

- Deploying, Scheduling & Networking of **multi cloud**
  - Our on-premise OpenStack & Kubernetes cluster
  - AWS, (Azure), (Vlaams Supercomputer Centrum OpenStack cloud)
  - Terraform and Consul cloud tools

- **IoT**-based **Edge** computing
  - IEEE 802.15.4 IoT project with edge nodes (Raspberry pi) for computing & networking
    - *Wireless sensor network* ***Testbed as a Service***
  - Lightweight *containerized* (Kubernetes) or *serverless* (e.g. openFaaS) computing

OTHER NEWS, FUTURE WORK, Q&A

WORKSHOP SURVEY AND Q&A

- **Survey** on topic interests for upcoming **workshop** session
  - **https://survey.opencloudedge.be** (google forms)
  - Multiple-choice: OpenStack, Kubernetes, Terraform, Consul
  - Other workshop requests? Let us know!

- **Q&A**