

Despliegue de una federación *cloud*

Instanciación, presentación de recursos, gestión de la pertenencia y monitorización y recuperación ante desastres

Lucas Cauhé Viñao

Escuela de Ingeniería y Arquitectura
Universidad de Zaragoza

2025-12-03

- 1 Introducción
- 2 Estado del arte
- 3 Análisis y diseño
- 4 Servicios de la federación
- 5 Validación y pruebas
- 6 Conclusiones

1 Introducción

- **Motivación**

- Gestión local de infraestructura pero servicios compartidos
- Planes de desarrollo UE para la soberanía de la información.

• Motivación

- Gestión local de infraestructura pero servicios compartidos
- Planes de desarrollo UE para la soberanía de la información.

• Objetivos

- Definir, implementar y validar modelo de federación.
- Dar una solución para la recuperación ante desastres.
- Definir la organización de los usuarios en federación.

2 Estado del arte

Infraestructura

- Cloud On-Premise,
OpenNebula
- Almacenamiento, Ceph

Infraestructura

- Cloud On-Premise,
OpenNebula
- Almacenamiento, Ceph

Servicios de la federación

- Gestor de políticas, XACML
- Modelo de federación, NIST
- Diseño de políticas, Gaia X

Infraestructura

- Cloud On-Premise,
OpenNebula
- Almacenamiento, Ceph

Servicios de la federación

- Gestor de políticas, XACML
- Modelo de federación, NIST
- Diseño de políticas, Gaia X

Despliegue

- Puppet
- Ansible
- OpenTofu

3 Análisis y diseño



Federación

- Políticas de amplio alcance
- Gestor de políticas y sistema de monitorización
- Catálogo de recursos



Federación

- Políticas de amplio alcance
- Gestor de políticas y sistema de monitorización
- Catálogo de recursos

Infraestructura

- Tolerante a fallos y alta disponibilidad
- Despliegue controlado
- Confiable y segura



Federación

- Políticas de amplio alcance
- Gestor de políticas y sistema de monitorización
- Catálogo de recursos

Infraestructura

- Tolerante a fallos y alta disponibilidad
- Despliegue controlado
- Confiable y segura

Recuperación ante desastres

- Disponibilidad del dato
- Almacenamiento de objetos
- Integración con OpenNebula



Infraestructura

- Cloud On-Premise
- Confianza entre entidades
- Adaptable a cada escenario

Infraestructura

- Cloud On-Premise
- Confianza entre entidades
- Adaptable a cada escenario

Arquitectura

- Cloud Privado
- Estructura modular y libre implementación

Infraestructura

- Cloud On-Premise
- Confianza entre entidades
- Adaptable a cada escenario

Arquitectura

- Cloud Privado
- Estructura modular y libre implementación

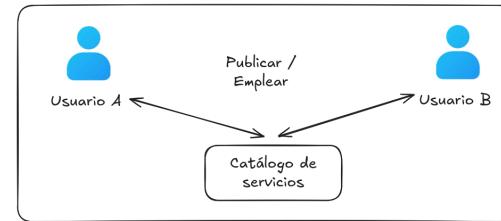
Recuperación ante desastres

- Madurez de entorno real
- Contenido en OpenNebula

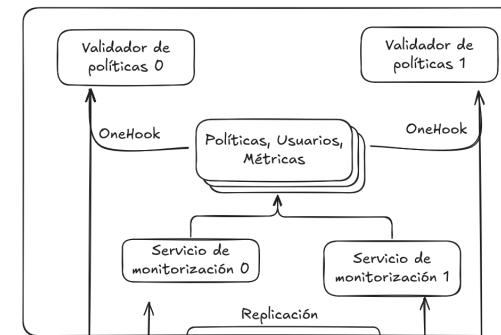
Arquitectura

- Abstracción en 3 planos.
- Servicios de catálogo, control de accesos, monitorización y backup.
- Gestión extendida de usuarios.

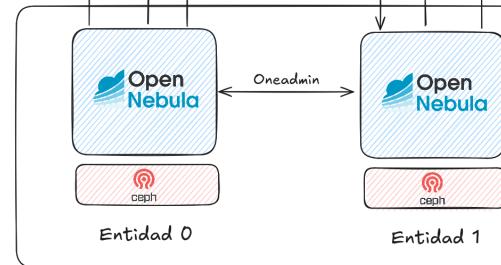
Plano
de uso



Plano
de
gestión

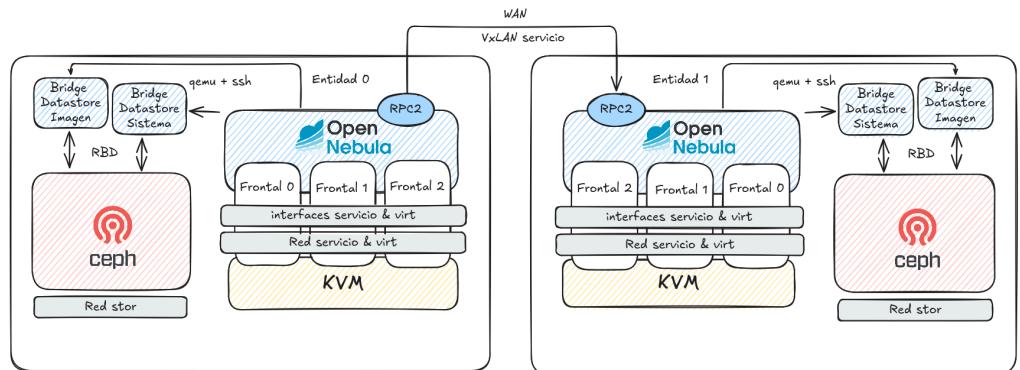


Plano
de
confianza



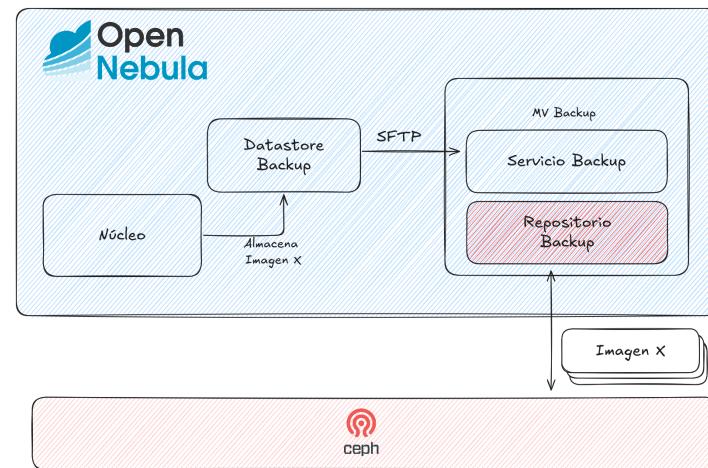
Infraestructura

- Red y almacenamiento aislado en cada entidad.
- OpenNebula y Ceph.
- Replicación segura.



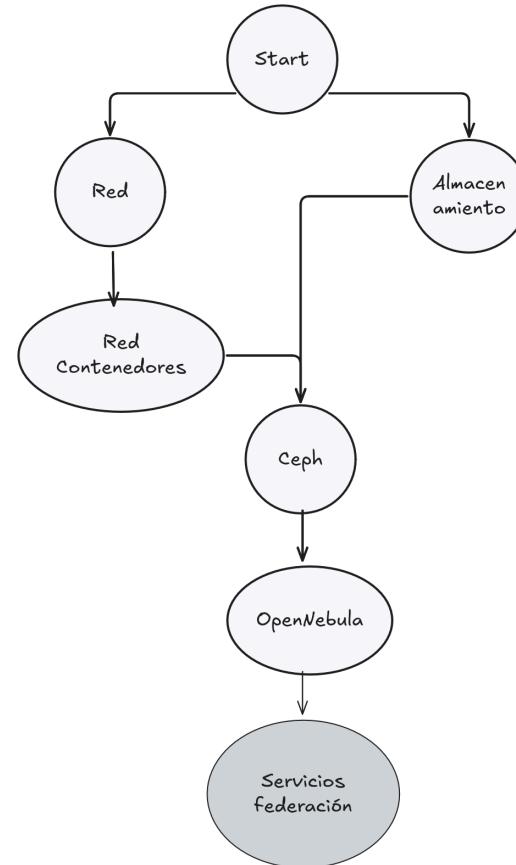
Recuperación ante desastres

- Contenido en OpenNebula.
- Datastore de *backup*.
- Estrategia basada en *snapshots* y *CoW*.



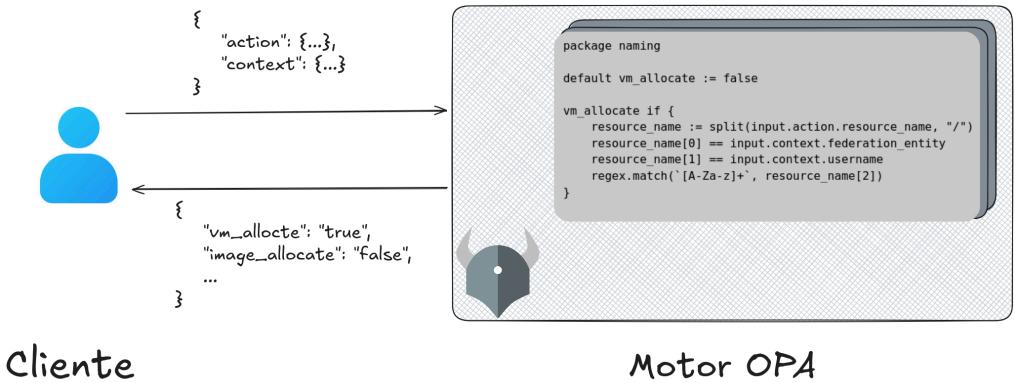
Despliegue

- Despliegue distribuido de *OpenNebula*.
- Despliegue semi-automático.
- Definición de relaciones de despliegue entre recursos.

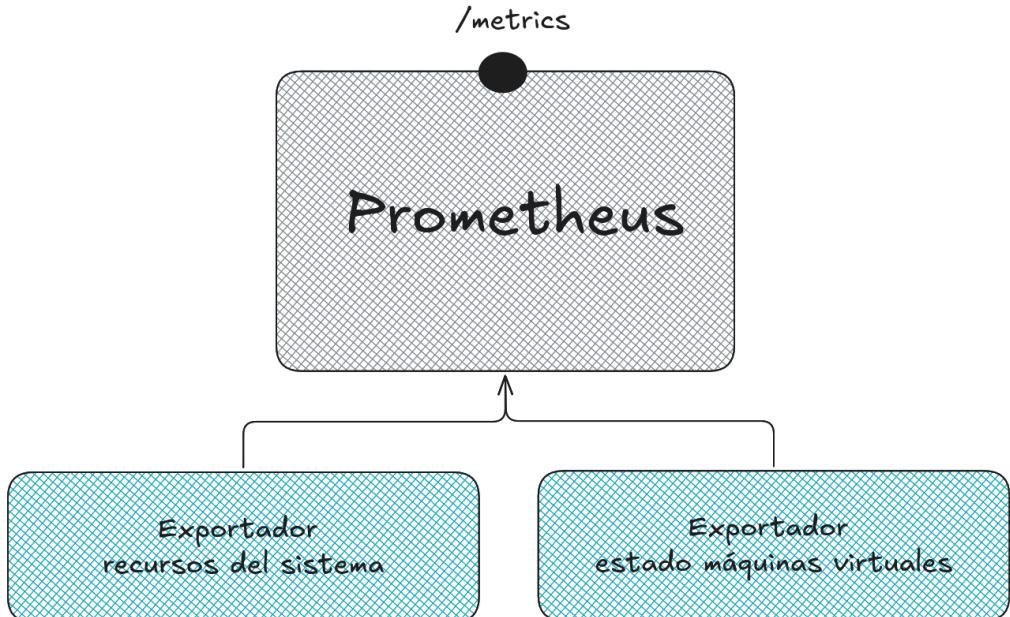


4 Servicios de la federación

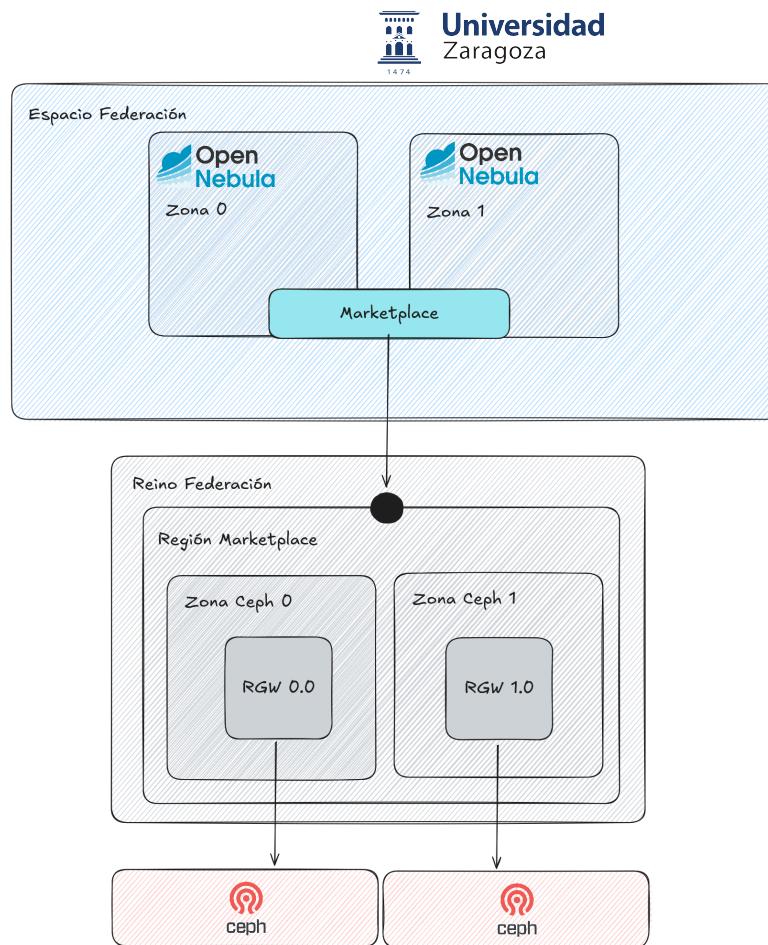
- Motor de control de accesos OPA
- Políticas expresadas en *Rego*
- Políticas de nomenclatura y sobrecarga



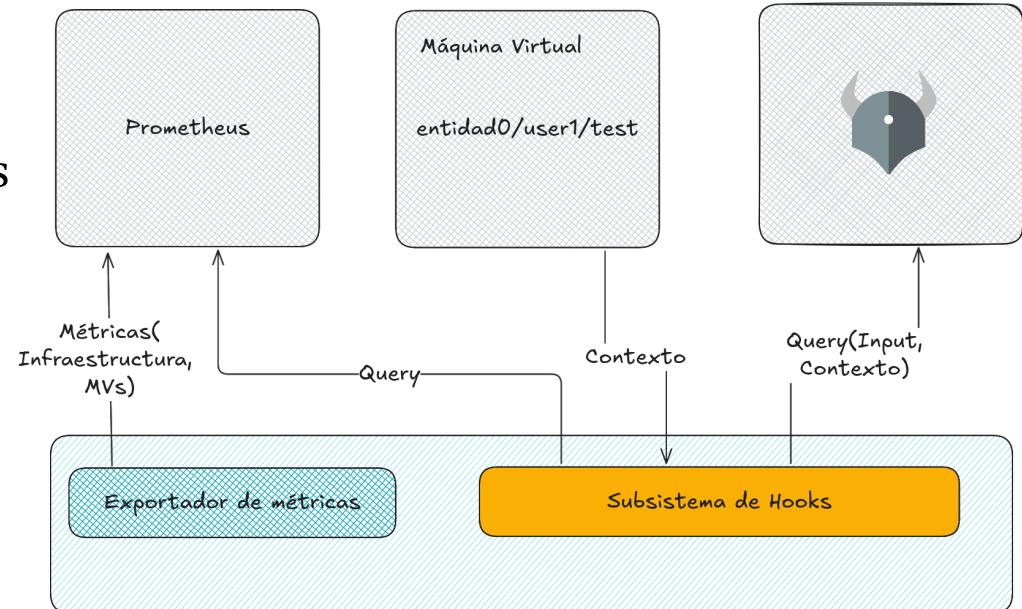
- Monitorización de infraestructura y estado de la federación
- Servicio de monitorización Prometheus
- Cumplimiento de políticas



- Marketplace OpenNebula
- Arquitectura "Multi-site" de Ceph
- Peticiones balanceadas



- Modelo de ejecución programada de eventos (*Hook Execution Manager*).
- Ejecución de *scripts Ruby*.
- Obtención de métricas y validación de políticas.



- Usuarios y permisos
- Almacenamiento y red
- Plan de despliegue

```
resource "opennebula_virtual_machine" "opa-10" {  
    provider = opennebula.policies-10  
  
    name      = "one-10/policies_10/opa"  
    group     = "policies"  
    permissions = "740" # use and manage by zone user and only  
    use by policies group  
  
    context = {  
        NETWORK      = "YES"  
        HOSTNAME     = "$NAME"  
        USERNAME     = "oneadmin"  
        PASSWORD     = local.zone1.nebula.password  
        SSH_PUBLIC_KEY = local.zone1.nebula.ssh_keys  
        START_SCRIPT = <<-EOT  
        # Instalación, montaje de Cephfs y puesta en marcha  
        EOT  
    }  
    disk {  
        image_id = opennebula_image.services-10.id  
        target   = "hda"  
        driver   = "qcow2"  
    }  
    nic {  
        model      = "virtio"  
        network_id = opennebula_virtual_network.services-10.id  
    }  
}
```

5 Validación y pruebas

- **Validación**
- **Pruebas de funcionamiento**
- **Prueba de sobrecarga**
- **Prueba de *backup***

6 Conclusiones

- **Objetivos alcanzados**

- Diseño, implementación y validación del prototipo de federación.
- Backup como recuperación ante desastres
- Gestión extendida de usuarios
- Sistema validado y probado

- **Objetivos alcanzados**

- Diseño, implementación y validación del prototipo de federación.
- Backup como recuperación ante desastres
- Gestión extendida de usuarios
- Sistema validado y probado

- **Conclusión Personal**

- Exposición a un problema real
- Gran aprendizaje

Despliegue de una federación *cloud*

Instanciación, presentación de recursos, gestión de la pertenencia y monitorización y recuperación ante desastres

Lucas Cauhé Viñao

Escuela de Ingeniería y Arquitectura
Universidad de Zaragoza

2025-12-03