RESILIENCE
REALIZED

KubeCon | CloudNativeCon
North America 2021

Storage and Networking: Rook-Ceph on Multus

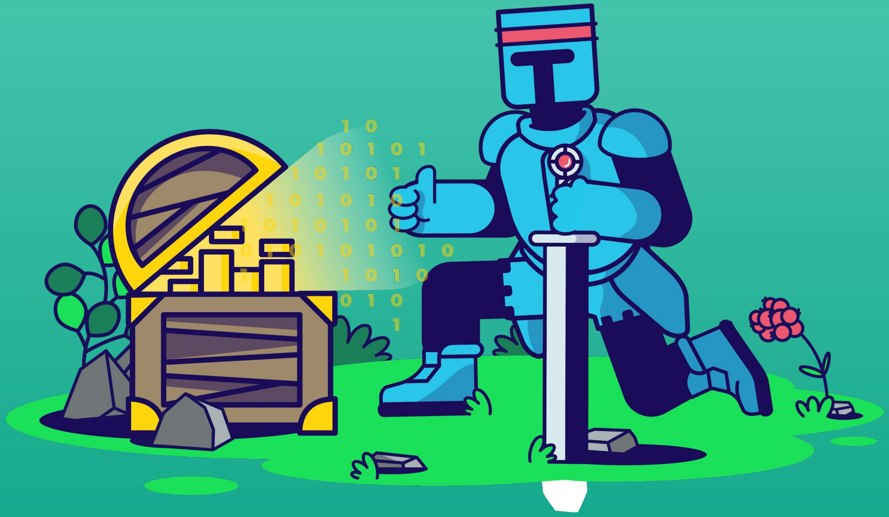
Sébastien Han
Rohan Gupta
13 Oct 2021

KUBERNETES STORAGE CHALLENGES

Storage challenges

- Kubernetes is a platform to manage distributed apps
    - Traditionally stateless
- Reliance on external storage (outside Kubernetes)
    - Not portable if the environment changes
    - Deployment burden
    - Day 2 operations (add capacity etc) - who is managing the storage?
- Reliance on cloud provider managed services
    - Vendor lock-in

ROOK



What is Rook-Ceph?

- Open Source
- Storage Operator for Kubernetes
- Automates Management of Ceph
    - Deployment
    - Configuration
    - Upgrading
- CNCF Graduated project (Oct 2020)
- Storage is then provided from the Kubernetes cluster
- Offers homogeneous experience regardless of the platform

CEPH

What is Ceph?

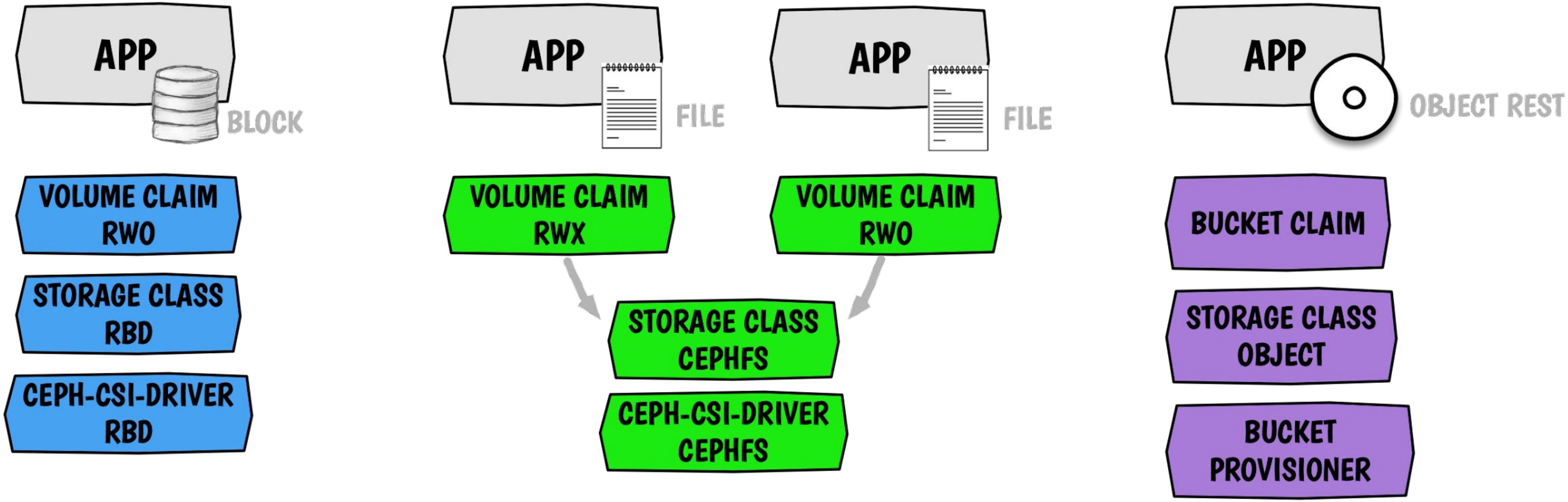
- Open Source
- Distributed storage software-defined solution
  - Block (Kernel module / QEMU plugin / NBD)
  - Shared File System (Native driver / FUSE)
  - Object Storage (Amazon S3 compliant)
- Support snapshot/clone/geo-replication for all storage interfaces
- Robust and battle tested for +10 years

ROOK-CEPH ARCHITECTURE

Architectural Layers

- Rook:
  - The operator owns the management of Ceph
- Ceph-CSI:
  - CSI driver dynamically provisions and connects client pods to the storage
- Ceph-COSI: coming soon! Just like CSI but for Object Bucket Claims
- Ceph:
  - **Data** layer

Dynamic provisioning

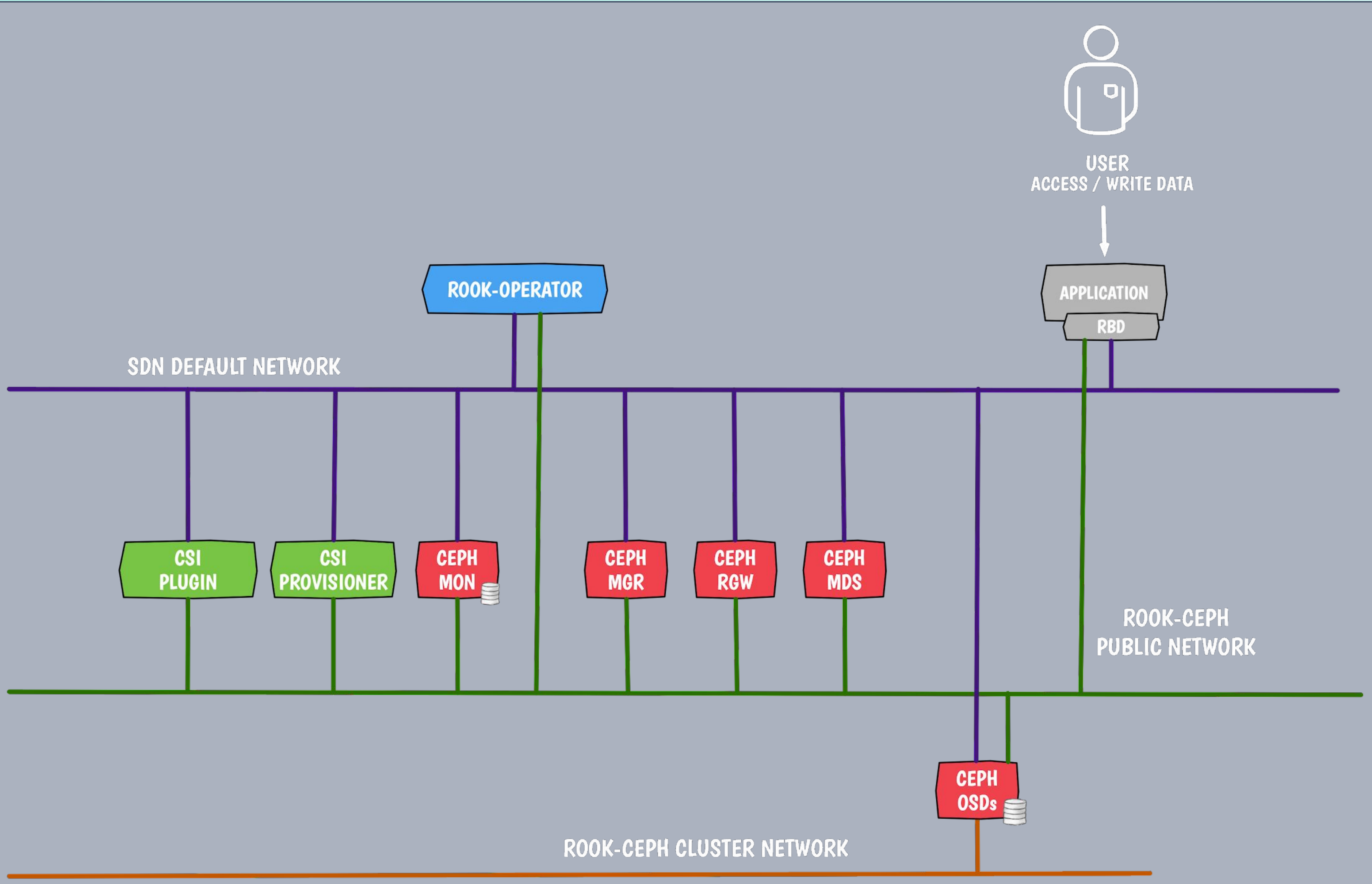


NETWORKING IN CEPH

Network capability

- Ceph supports two networks:
    - public (client-side)
    - cluster replication
- Ceph functions with a public network only, but you may see significant performance improvement with a second “cluster” network

Network topology



NETWORKING WITH MULTUS IN ROOK-CEPH

Networking models

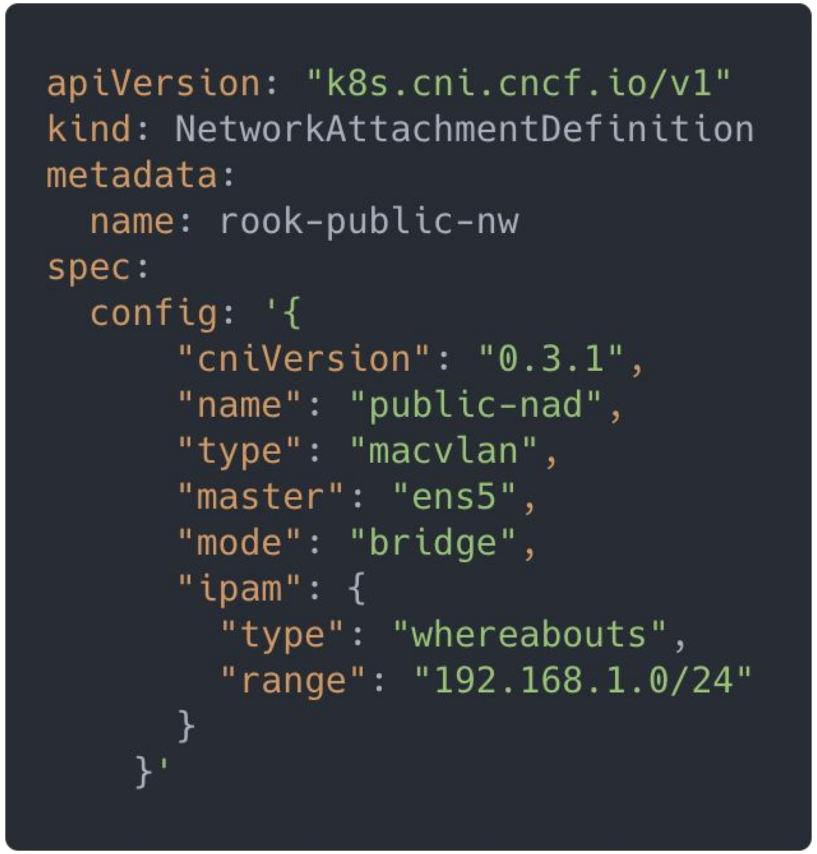
- Rook supports the following networking methods:
  - Traditional pod networking - single network interface - default SDN
  - Host networking - runs on host network namespace and uses host IP. All host’s network stack is visible
  - Multus - Rook supports addition of public and cluster network for Ceph

IPAM - choose wisely

All of our testing and recommandations go with the *'whereabout'* IP Address Management:

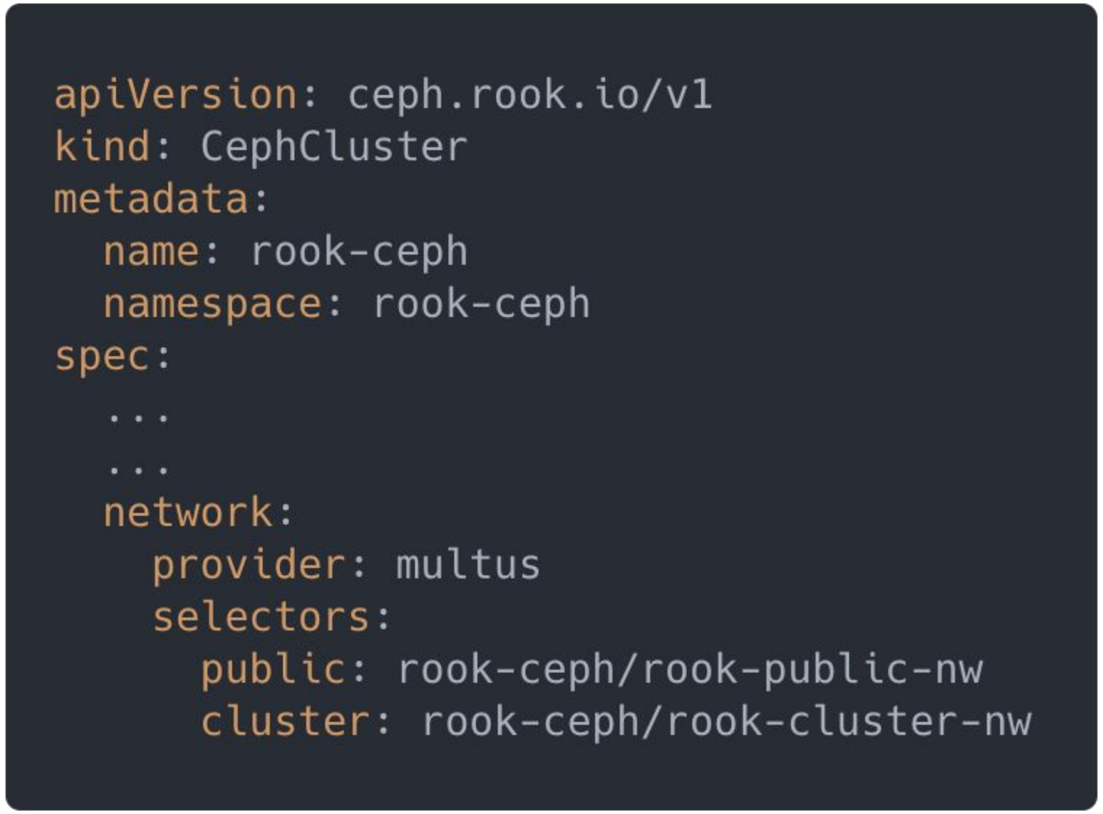
- Cluster-wide IP assigning support
- IPv4 and IPv6 support (not dual-stack)
- No DHCP involved!

Network Attachment Definition



```
apiVersion: "k8s.cni.cncf.io/v1"
kind: NetworkAttachmentDefinition
metadata:
  name: rook-public-nw
spec:
  config: '{
      "cniVersion": "0.3.1",
      "name": "public-nad",
      "type": "macvlan",
      "master": "ens5",
      "mode": "bridge",
      "ipam": {
        "type": "whereabouts",
        "range": "192.168.1.0/24"
      }
    }'
```

ROOK-CEPH CRD



```yaml
apiVersion: ceph.rook.io/v1
kind: CephCluster
metadata:
  name: rook-ceph
  namespace: rook-ceph
spec:
  ...
  ...
  network:
    provider: multus
    selectors:
      public: rook-ceph/rook-public-nw
      cluster: rook-ceph/rook-cluster-nw
```

PERFORMANCE

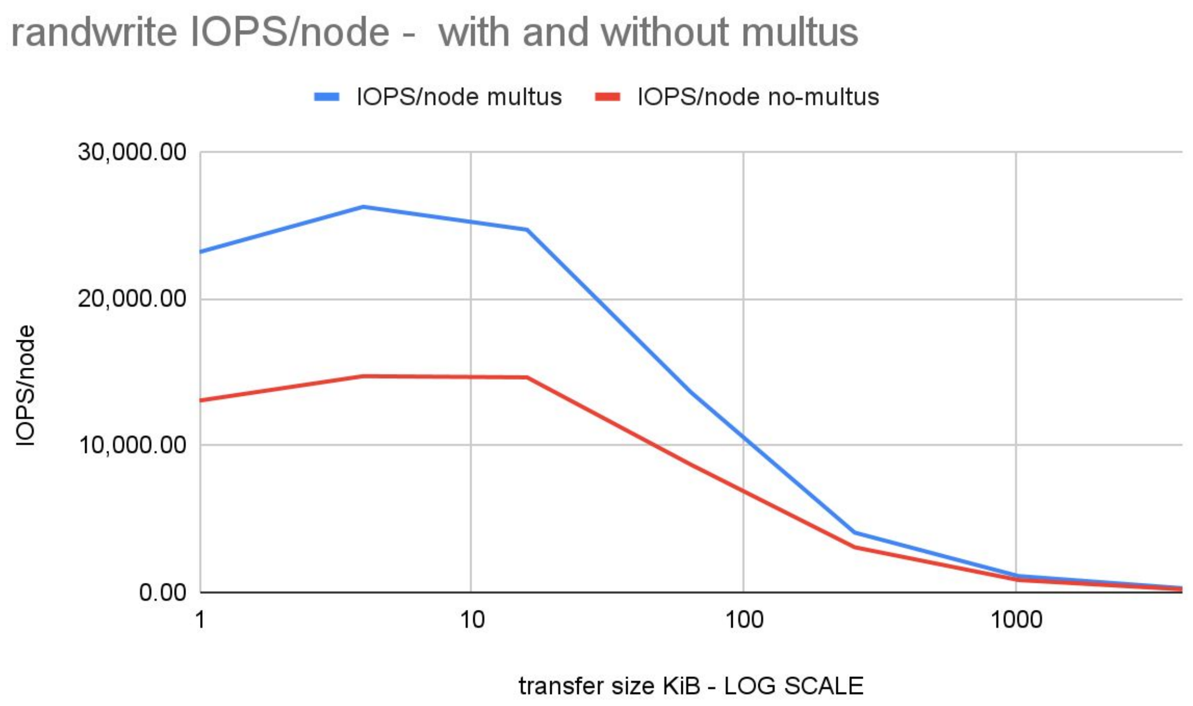
What’s in the box?

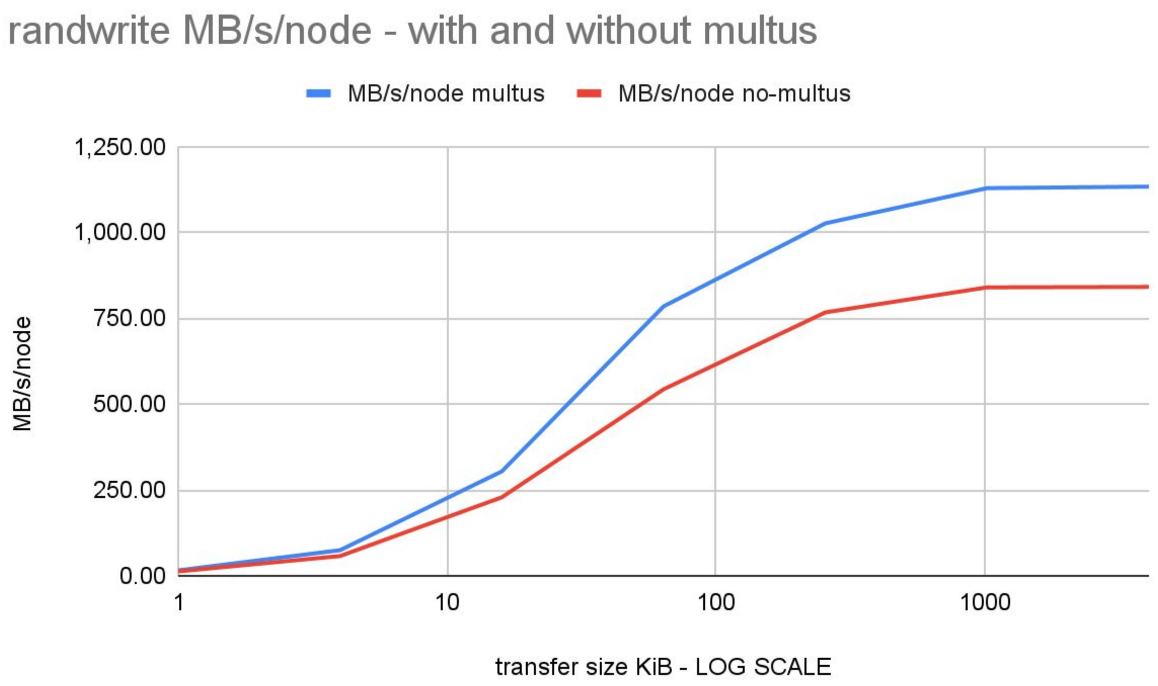
The hardware used is not relevant. We are simply focusing on comparing with and without multiple network interfaces.

Networks:

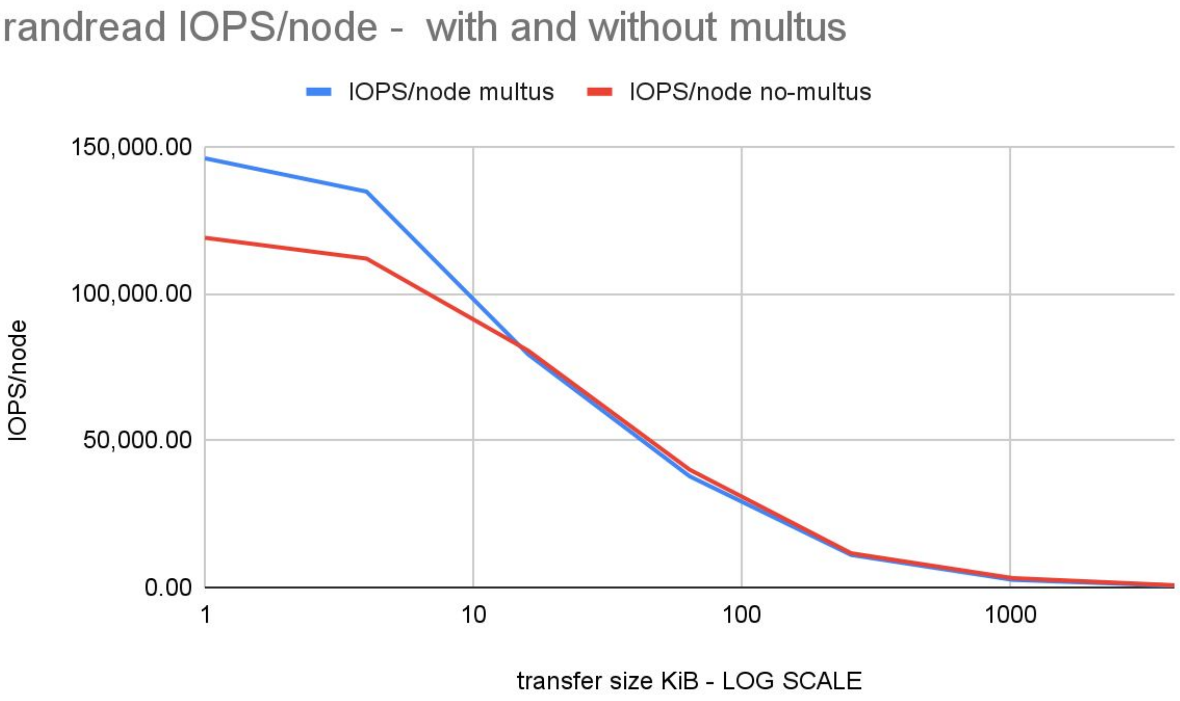
1. Default SDN network
2. Ceph public network (multus)
3. Ceph private network (multus)

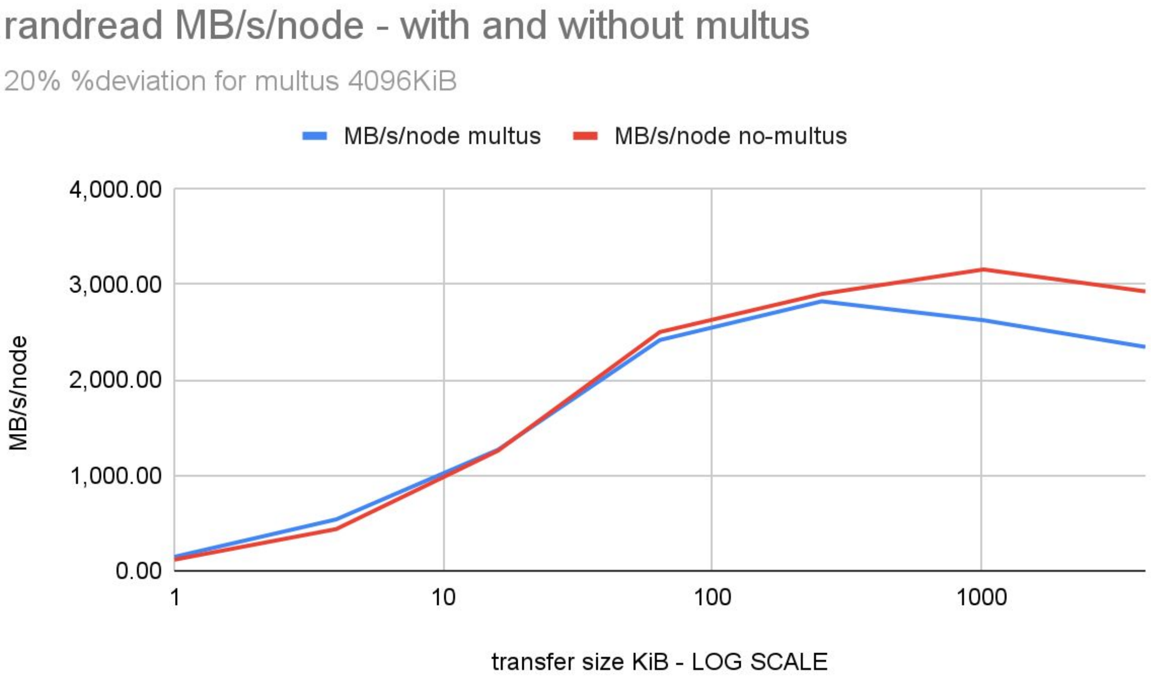
Random WRITE - IOPS and Bandwidth





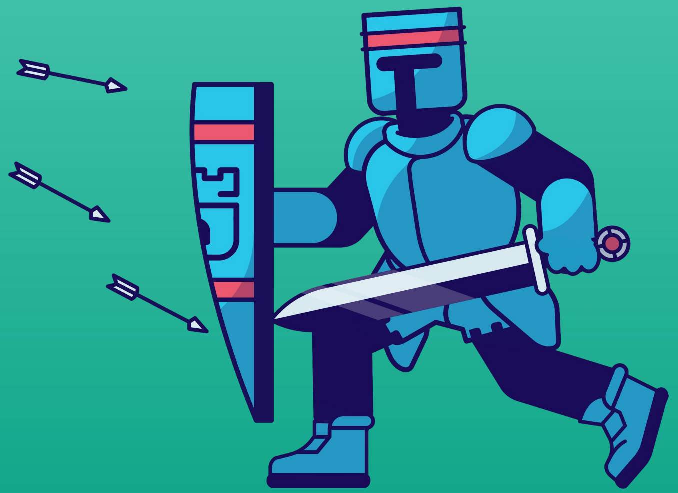
Random READ - IOPS and Bandwidth





DEMO TIME

Cluster walkthrough



KEY TAKEAWAYS

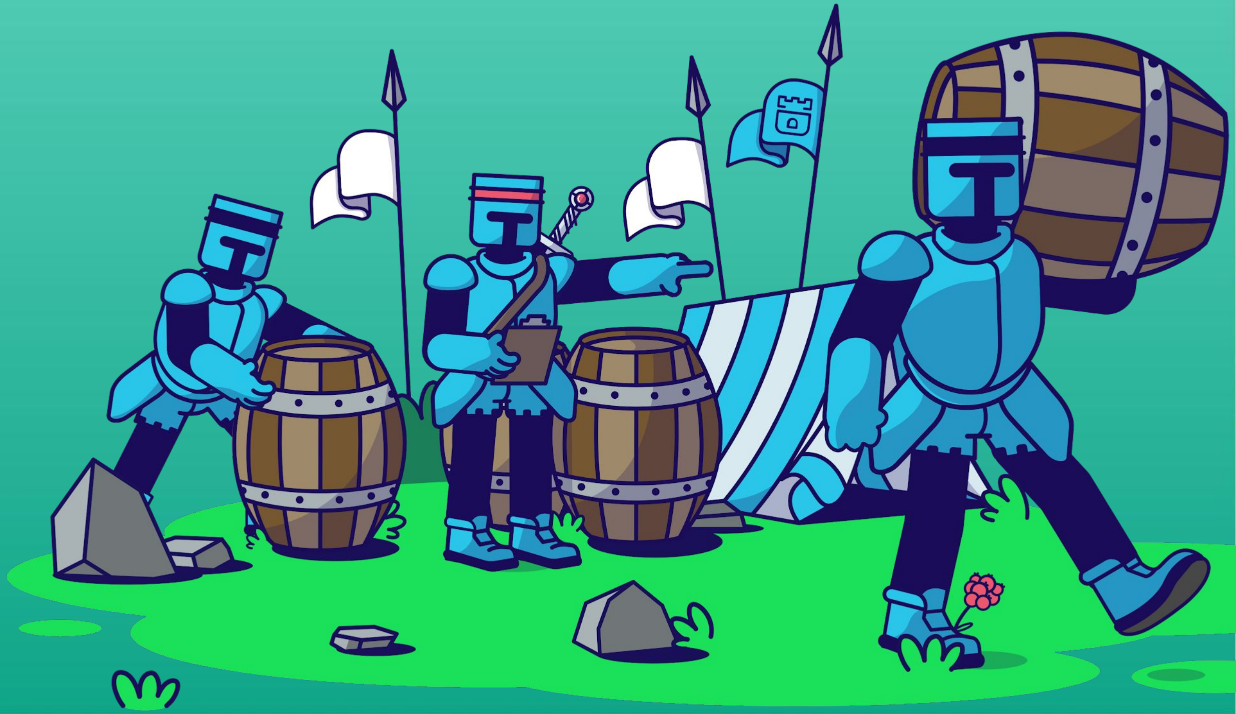
Wrap it up

- Separating Ceph networks is possible with Multus
- Whereabout IPAM is preferred
- Performance improvement than just using a single interface
- Available since Rook v1.7

Thanks!

https://rook.io/

rohgupta@redhat.com
seb@redhat.com



Links

- https://rook.io
- https://github.com/rook/rook
- https://docs.ceph.com/en/latest/rados/configuration/network-config-ref/
- https://github.com/k8snetworkplumbingwg/multus-cni
- https://github.com/k8snetworkplumbingwg/whereabouts