State of Container Networking

Where are we at and where are we going?

Frederick F. Kautz IV / Red Hat

@ffkiv

THE LINUX FOUNDATION

THE LINUX FOUNDATION

OPEN SOURCE SUMMIT

Container Networking

THE LINUX FOUNDATION

Containers

- Virtualization Methodology
  - OS Kernel allows for multiple isolated user space
  - Isolation by features such as cgroups and namespaces
- Cgroups provides ability to
  - limit, account and isolate resource usage of process groups
  - prioritize resources and control that includes freeze, checkpoint and restarts
- Namespace
  - partitions key kernel structure to create environment that include process, network, IPC, mount points, hostname and user

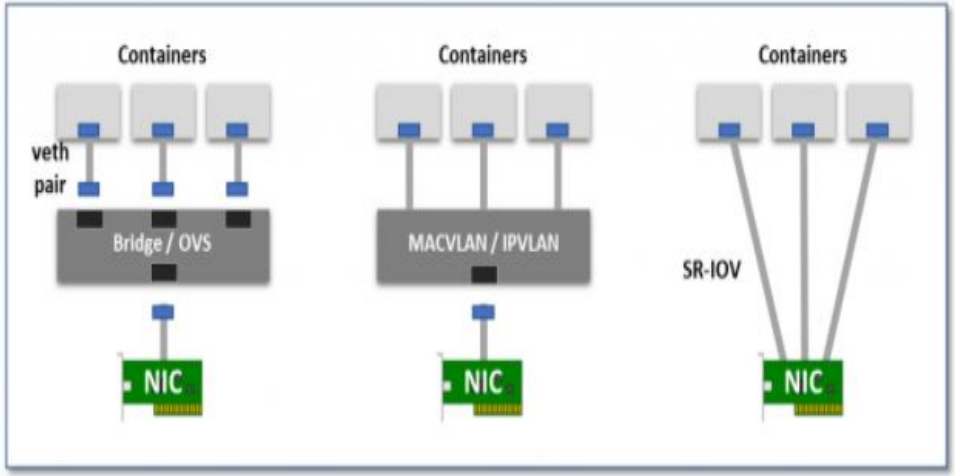
Container Virtualization Benefits

- Density - More containers than VMs in a single host
- Speed - Starting up a container can take less than a second
- Low overhead Management - Lower weight orchestration
- Portability - Encapsulating an application and its configuration simplifies the migration process
- Options - Variety of different Open Source standards

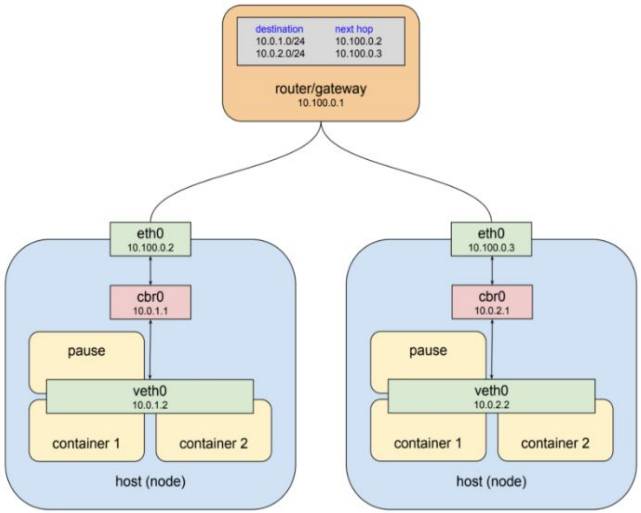
Container Management Landscape

Container Networking Types

- None
- Bridge
- Host
- Overlay
- Underlay



Container Networking Model



Container Networking Landscape

Google Compute Engine

weavenet

Major Types of SDNs

- Dataplane OSI Level
  - L2 - MAC Address, L2 Switches
  - L3 - IP Address, IP Routers
  - L4 - TCP/UDP Ports, Load Balancers
  - L7 - Application aware, Service Mesh
- Control plane
  - Static
  - Centralized
  - Distributed
- Management plane
  - CNI
  - libnetwork

Contiv

- Multi host policy based networking
- Multiple backend drivers:
  - L2 (VLAN)
  - L3 (BGP)
  - Overlay (VXLAN)
  - Cisco SDN (ACI)
- Policy support

Flannel

- Allocates a separate subnet per host
- Overlay network between each host
- K8S API or etcd for configuration management
- No policy, pair with calico

Weave

- Uses standard port numbers for containers
- Container IP discovered from DNS query on container name
- Two connection modes :
  - Sleeve mode
    - UDP channel to traverse IP packets from containers
  - Fastdp mode
    - VXLAN based solution

Calico

- Policy focused
- Pure layer 3 approach
- Also implements BGP for routing, for scaling
- Option to use stateless IPinIP overlay

Contrail

- Policy support
- Gateway services
- SNAT
- ECMP load balancing in services
- Ingress load balancing

OpenDaylight

- Openstack Kuryr Integration
- POD L2 connectivity same node
- POD L3 connectivity multi-node
- Service connectivity WIP

OVN

- Creates logical switches and routers
- Reference architecture for OVS based container networking solutions
- Lightweight control plane with essential features
- Geneve based

Data Planes

VPP

- Packets processed in batch through nodes in a Directed Graph
- Routing decisions in Userspace
- Attempts to eliminate cache misses
- Supports DPDK

OpenVSwitch

- Dataplane for many production quality SDN solutions
- Packets processed rules in tables
- Routing decisions in Kernel
- Configurable through OVSDB and OpenFlow
- Extensive set of built in features
- Supports DPDK


THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
Recent Advances
THE LINUX FOUNDATION

eBPF

- eBPF support in kernel expanded
  - bpfilter
- Historically used netfilter hook join-points
- XDP added eBPF before memory allocation for received packets
- Cilium added eBPF data interception before linux allocation occurs

Shared Memory

- VPP based shared memory driver
- Userspace solution
- App can be modified to use libmemif directly
- LD_PRELOAD to redirect socket

Service Mesh

- Layer 7 Load Balancer & Service Discovery
- Focus primarily on MicroServices
- Load Balancing
- Failure recovery
- Graceful function degradation
- Distinct from SDN

K8s on Edge

- Starting to see real world deployments
  - Chic-Fil-A running 2000 K8s clusters, one for each store
- K8s-based edge data centers
  - Vapor.io
- SDNs expanding beyond network virtualizer to support edge

5G CNFs on K8s

- New term, Cloud Native Network Functions (CNFs)
- VNF to CNF not straightforward
- May have virtualized components
  - Why? Kernel modules!
- Moved from talking about CNFs to building infrastructure
- Despite new efforts, CNFs still have many open questions

Network Service Mesh

- Cloud-Native Controller
- Matchmaking for cross-connects
  - Pod -> Pod
  - Dataplane -> Dataplane
  - Pod -> Device
- Allows non-IP payloads
- Choose your favorite dataplane
- Implements SFC
- Doesn't require changes K8s

Multus CNI

- Coordinates multiple CNI plugins why may be backed by multiple SDNs
- Executes multiple CNI plugins for a single pod
- End result may be multiple interfaces

IPv6

- K8S supports IPv6-only clusters
  - No mixed IPv6 + IPv4 cluster
- Segment Routing (IETF)
- Work here continues to progress

Working Groups

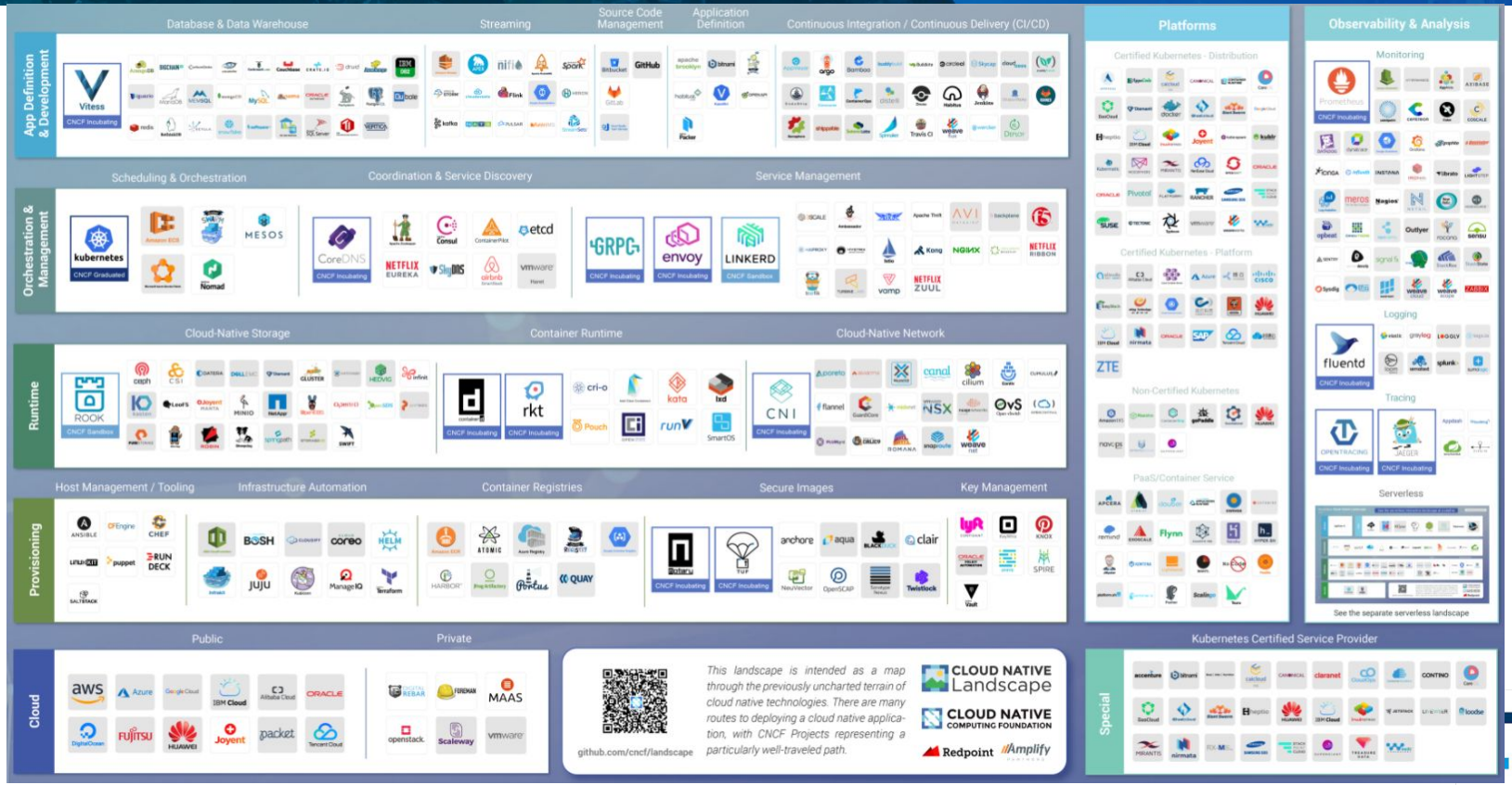
- Several working groups:
- CNCF CNI
- Network SIG
- Network Plumbing Working Group, sub group of K8s sig-network
- Network Service Mesh
- Istio Working Group

- CNCF driving CNF definition
- ONAP
- OpenDaylight COE Project

Current work

- Increase diversity in container networking
  - SR-IOV
  - Memif
  - eBPF bypass
- IPV6-only Deployments
- Multi-Endpoint (includes Multi-Interface)
- More eBPF support
- Telco involvement
- VPP

Cloud Native Landscape



THE LINUX FOUNDATION

Future Trends

- Kubernetes in Telco
- Kubernetes in more advanced enterprise (beyond network virt)
- Edge Containers (Better interop between SDN + Schedulers)
- Smarter Multi-Site Interop (Cloud <~> Cloud / Cloud <~> OnPrem)
- Service Mesh + SDN Interop
- Network Service Mesh + SDN Interop
- Openstack Services Managed by Kubernetes
- More kernel bypass with eBPF
- SDN use P4 advanced use cases (PISA chips become common)

References

- google-cloud/understanding-kubernetes-networking
- Container-landscape
- Hackers-guide-kubernetes-networking
- ligato/container-networking-overview