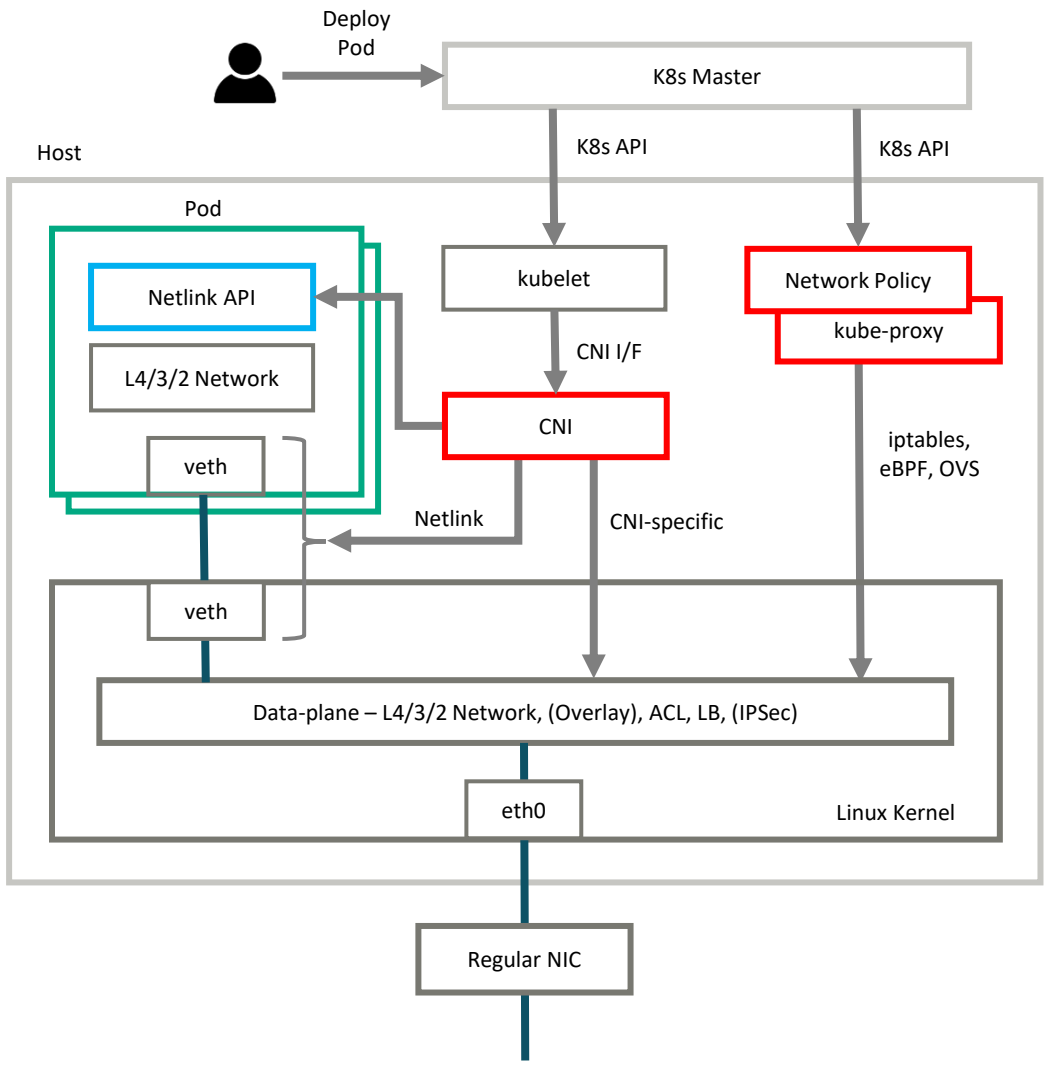
K8S CNI FOR XPU OFFLOADING

OPI Community Discussion

Toshi Kani, HPE

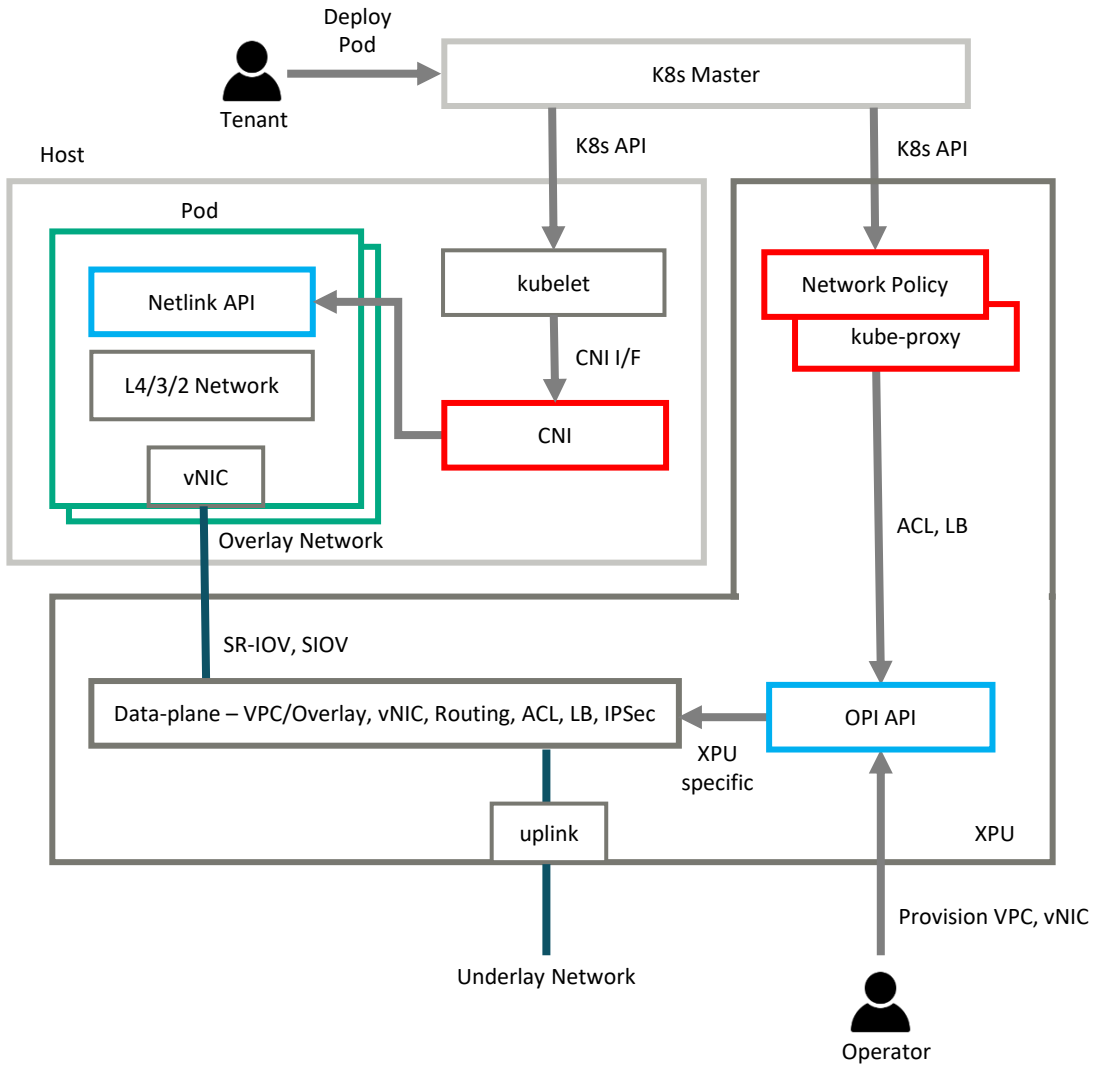
Feb 2023

K8S CNI WITH REGULAR NIC



- K8s CNI sets up container network
  - Provision container network data-plane inside Linux kernel
    - The data-plane design & API vary in CNI types
  - Create a virtual ethernet (veth pair) and attach it to Pod
  - Setup veth via netlink API from the Pod network namespace
- Network Policy monitors K8s API and sets up the data-plane for K8s NetworkPolicy as stateful firewall (ACL)
  - Calico Felix has plugin I/F to support multiple data-plane types, such as iptables, Windows, VPP
- kube-proxy monitors K8s API and sets up the data-plane for K8s Service as stateful load-balancing
  - K8s default kube-proxy supports iptables & ipvs. Some CNI packages replace it for using other methods, such as eBPF
- Linux kernel as the container network data-plane leads to:
  - Consuming host CPU cycles
  - Limited network performance
  - Security attack surface on the host

HIGH-LEVEL DESIGN



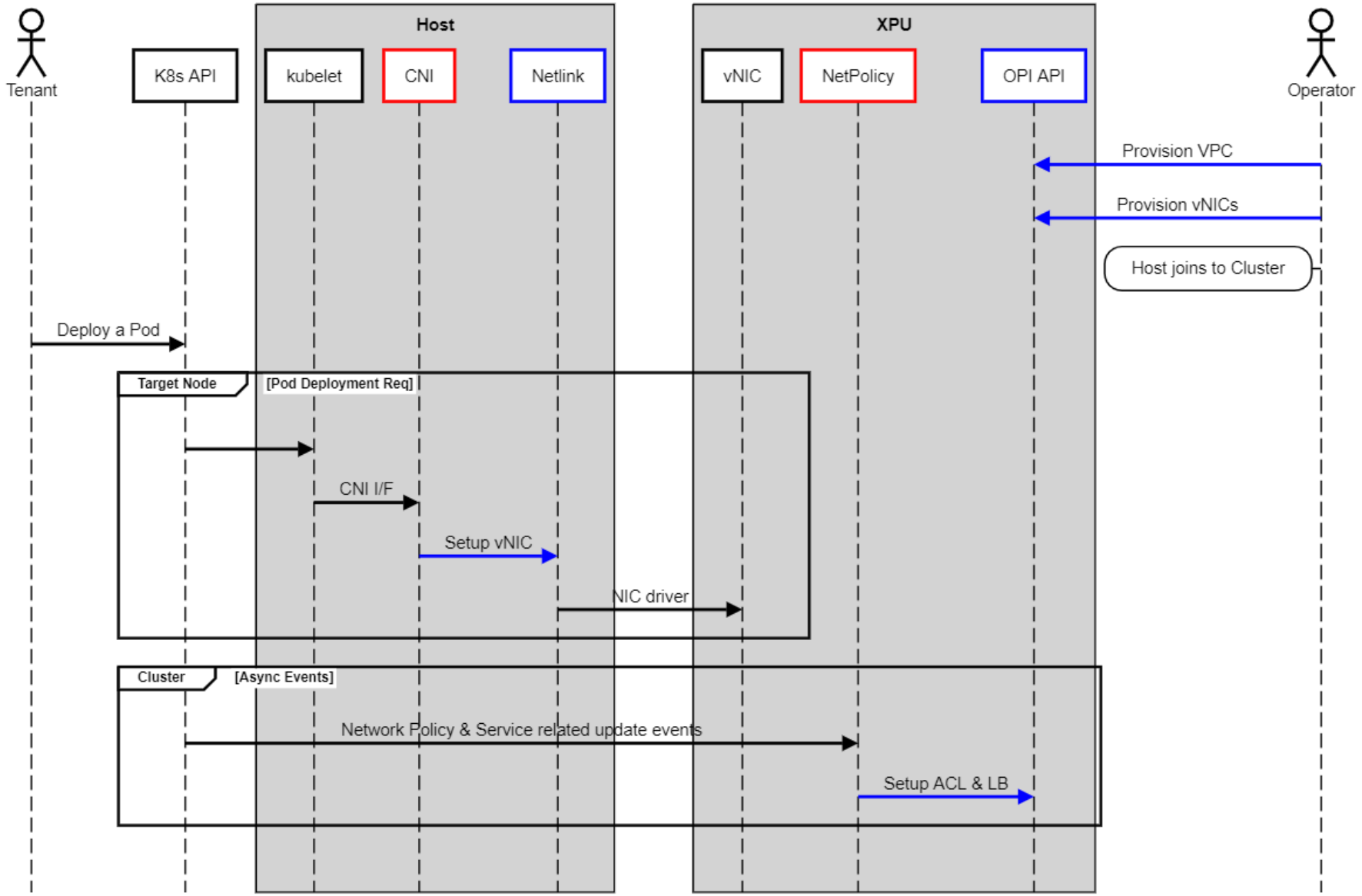
- The container network data-plane is offloaded to XPU
  - vNIC (as SR-IOV VF or SIOV) is directly attached to Pod
- XPU provides security isolation between tenant and operator
  - Tenant overlay network (VPC) isolates vNIC from the operator underlay network
- K8s CNI package calls OPI API and netlink API
  - OPI API – XPU API that sets up the container network data-plane inside XPU (See the next page)
  - Netlink API – Linux API that sets up a container NIC interfaces, e.g., link state, IP, MTU, routing. No API change
- K8s CNI package has the following main modules
  - K8s CNI plugin – Attach vNIC to Pod and set it up from the Pod network namespace
  - Network Policy – Monitor K8s API and call OPI API for K8s NetworkPolicy (ACL) offloading
  - Kube-proxy – Monitor K8s API and call OPI API for K8s Service (LB) offloading

OPI API FUNCTIONS FOR K8S CNI XPU OFFLOADING

- Provision VPC
  - Setup overlay network (e.g., VXLAN), tunnel endpoint, underlay routing, etc.
  - Setup IPSec for underlay network (e.g., VXLAN over IPSec) – Optional feature
- Provision vNIC
  - Allocate vNIC as an SR-IOV (or SIOV) NIC VF, which is an L2 NIC interface in the overlay network
    - Static provisioning may require some offloading setup, e.g., traffic shaping, at Pod deployment as a separate step
- Setup Stateful Firewall (ACL)
  - Setup ingress & egress stateful L3/4 firewall, similar functionality as AWS Security Groups (SG)
    - Policy rule with a group target, e.g., iptables IPSet and AWS SG, is highly desirable
- Setup Stateful Load-Balancing (LB)
  - Setup stateful L3/4 load-balancing with DNAT & SNAT

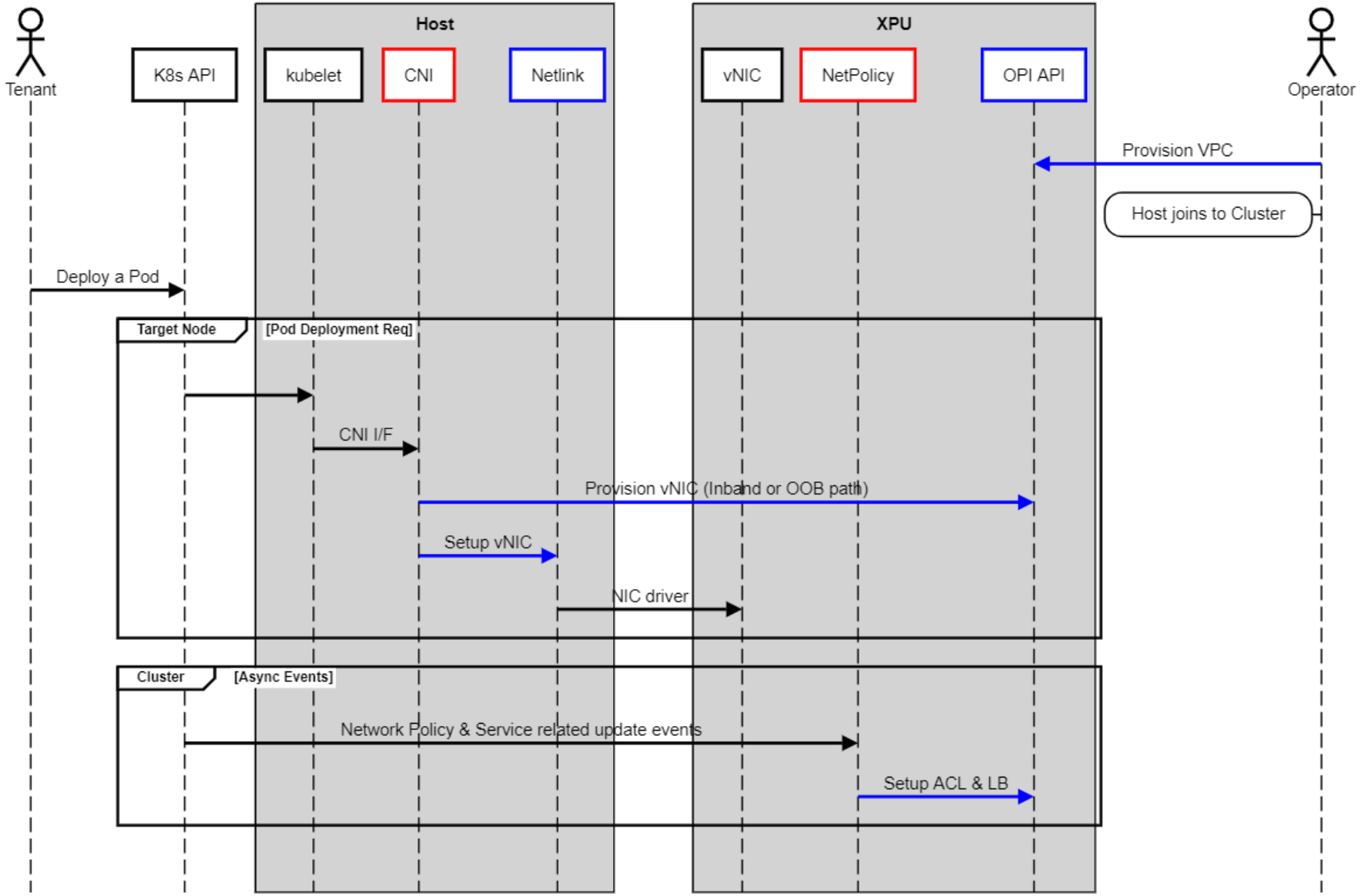
Note: OPI API interface definition is intentionally left for separate discussions

STATIC VNIC PROVISIONING FLOW



Note: Kube-proxy is omitted for brevity.
The sequence is shared with NetPolicy

DYNAMIC VNIC PROVISIONING FLOW



Note: Impl may vary in detail., e.g., alt. provision vNIC in the background to replenish minimal assignable vNICs