
KubeCon
CloudNativeCon
North America 2023


KubeCon | CloudNativeCon

North America 2023

Learn How to Build an eBPF CNI Plugin from Scratch

Adam Sayah, Solo.io

Speaker

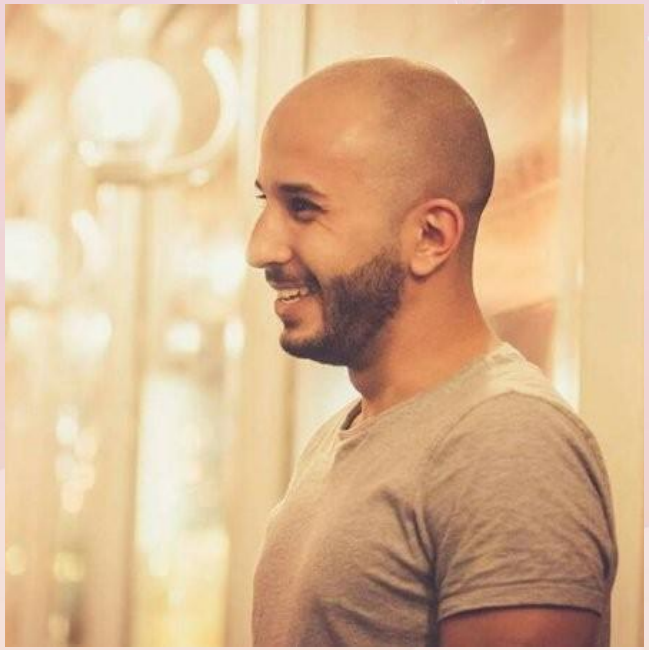
KubeCon | CloudNativeCon
North America 2023

Adam Sayah

Solo.io

_asayah

linkedin.com/in/adamsayah

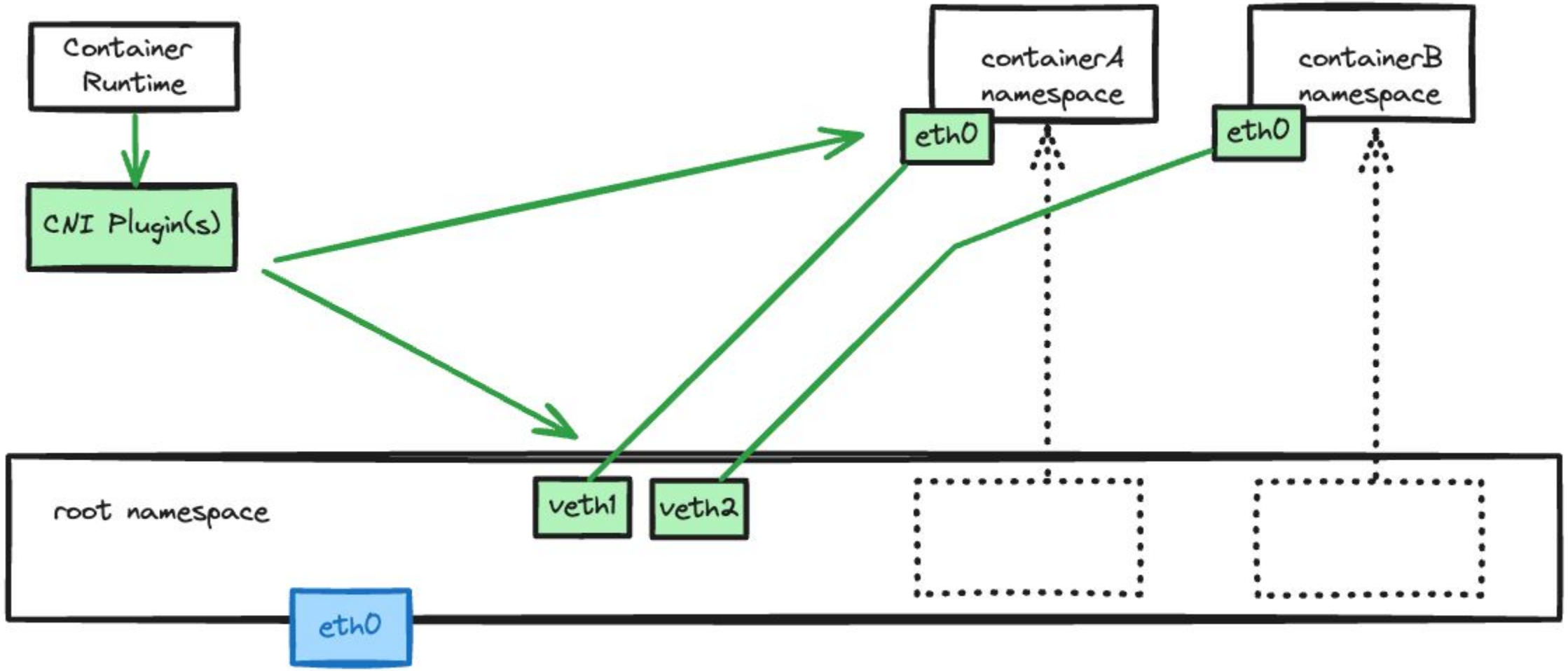


Agenda

- What is a CNI ?
- Exercise 1: Create a CNI
- Exercise 2: eBPF Basics
- Exercise 3 : eBPF Maps / Monitoring
- Exercice 4: eBPF for Security.
- Conclusion


What is a CNI ?

What is a CNI?

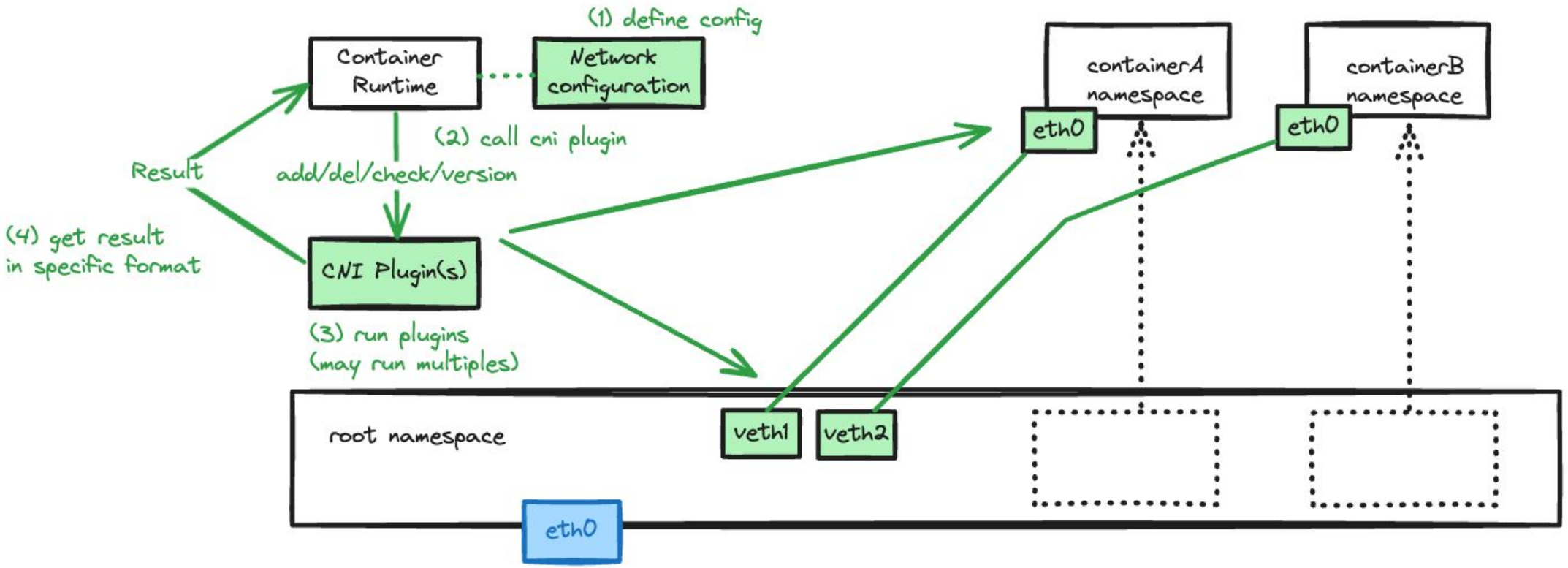

Container Runtime
CNI Plugin(s)
containerA namespace
eth0
containerB namespace
eth0
root namespace
veth1
veth2
eth0

https://www.cni.dev

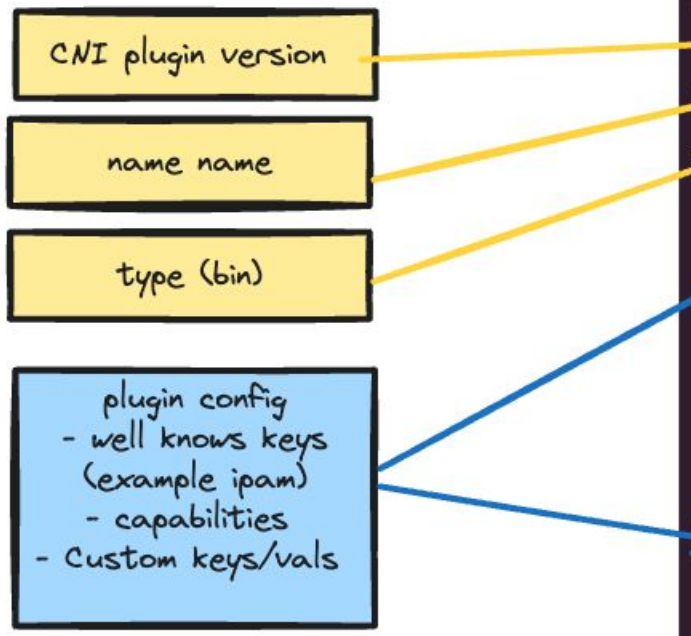
CNI

- CNI Specification
- Implementations
- Lib

CNI - Spec


(1) define config
Container Runtime
Network configuration
(2) call cni plugin
Result
add/del/check/version
(4) get result in specific format
CNI Plugin(s)
(3) run plugins (may run multiples)
containerA namespace
containerB namespace
eth0
eth0
root namespace
veth1
veth2
eth0

CNI - Configuration

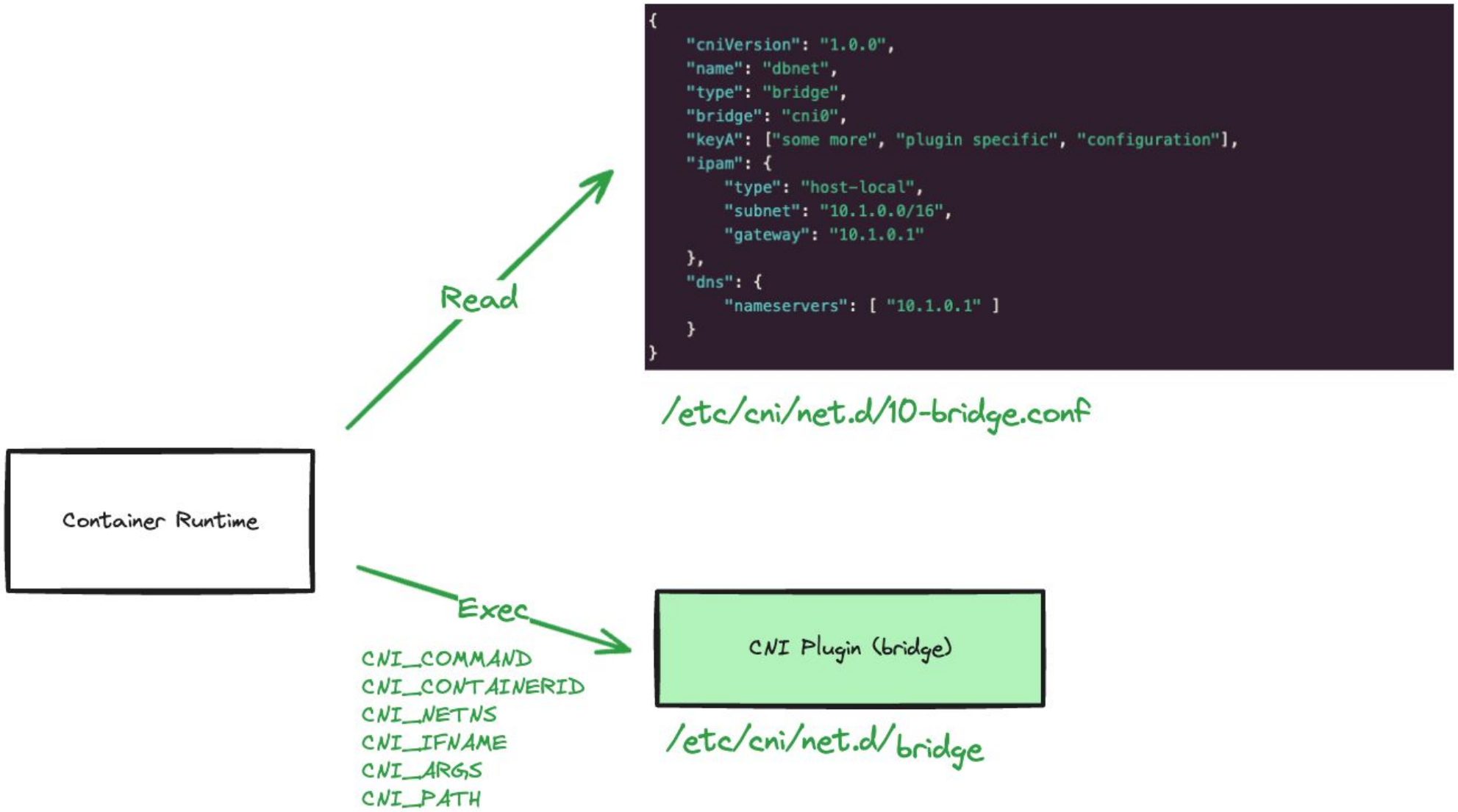



```
{
    "cniVersion": "1.0.0",
    "name": "dbnet",
    "type": "bridge",
    "bridge": "cni0",
    "keyA": ["some more", "plugin specific", "configuration"],
    "ipam": {
        "type": "host-local",
        "subnet": "10.1.0.0/16",
        "gateway": "10.1.0.1"
    },
    "dns": {
        "nameservers": [ "10.1.0.1" ]
    }
}
```

CNI - Result


```
{
    "cniVersion": "1.0.0",
    "name": "dbnet",
    "type": "bridge",
    "bridge": "cni0",
    "keyA": ["some more", "plugin specific", "configuration"],
    "ipam": {
        "type": "host-local",
        "subnet": "10.1.0.0/16",
        "gateway": "10.1.0.1"
    },
    "dns": {
        "nameservers": [ "10.1.0.1" ]
    }
}
```

CNI - Execution

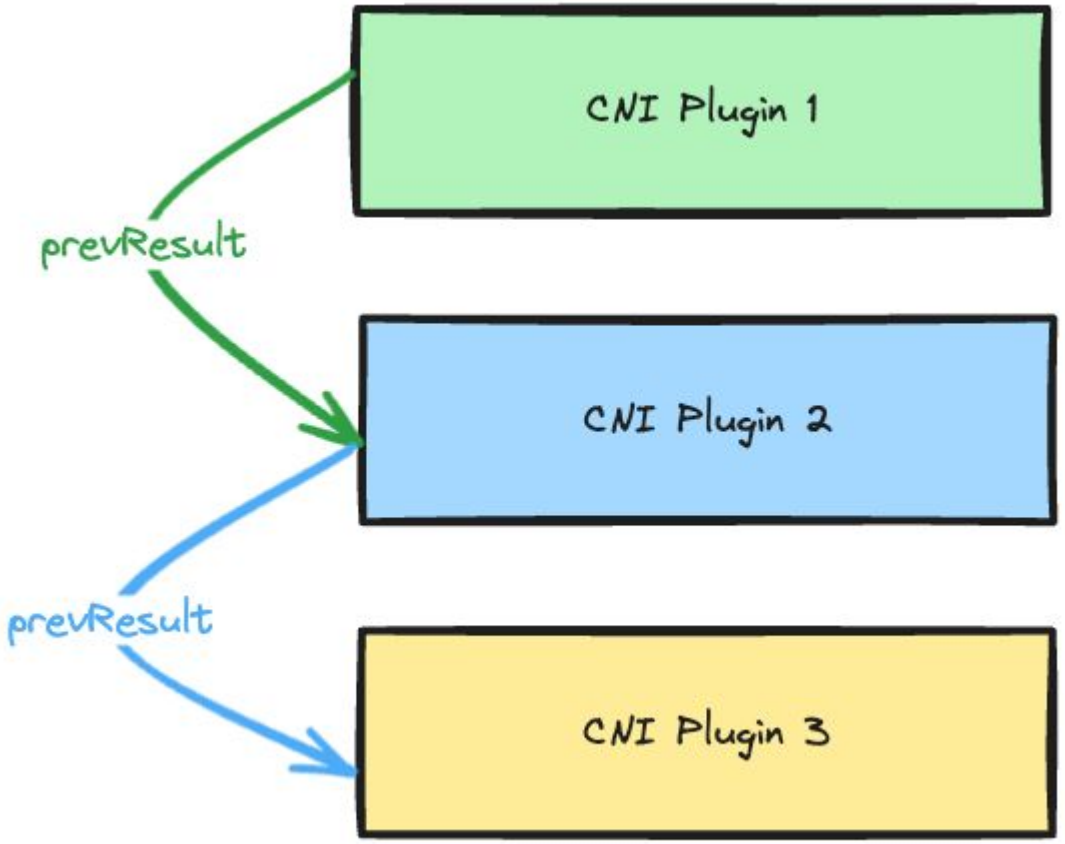


CNI - Parameters

- `CNI_COMMAND`: indicates the desired operation; `ADD`, `DEL`, `CHECK`, `GC`, or `VERSION`.
- `CNI_CONTAINERID`: Container ID. A unique plaintext identifier for a container, allocated by the runtime.
- `CNI_NETNS`: A reference to the container's "isolation domain". If using network namespaces, then a path to the network namespace (e.g. `/run/netns/[nsname]`)
- `CNI_IFNAME`: Name of the interface to create inside the container
- `CNI_ARGS`: Extra arguments passed in by the user at invocation time. Alphanumeric key-value pairs separated by semicolons; for example, "FOO=BAR;ABC=123"
- `CNI_PATH`: List of paths to search for CNI plugin executables.

Source: https://github.com/containernetworking/cni/blob/main/SPEC.md#parameters

CNI - Multiple Plugins

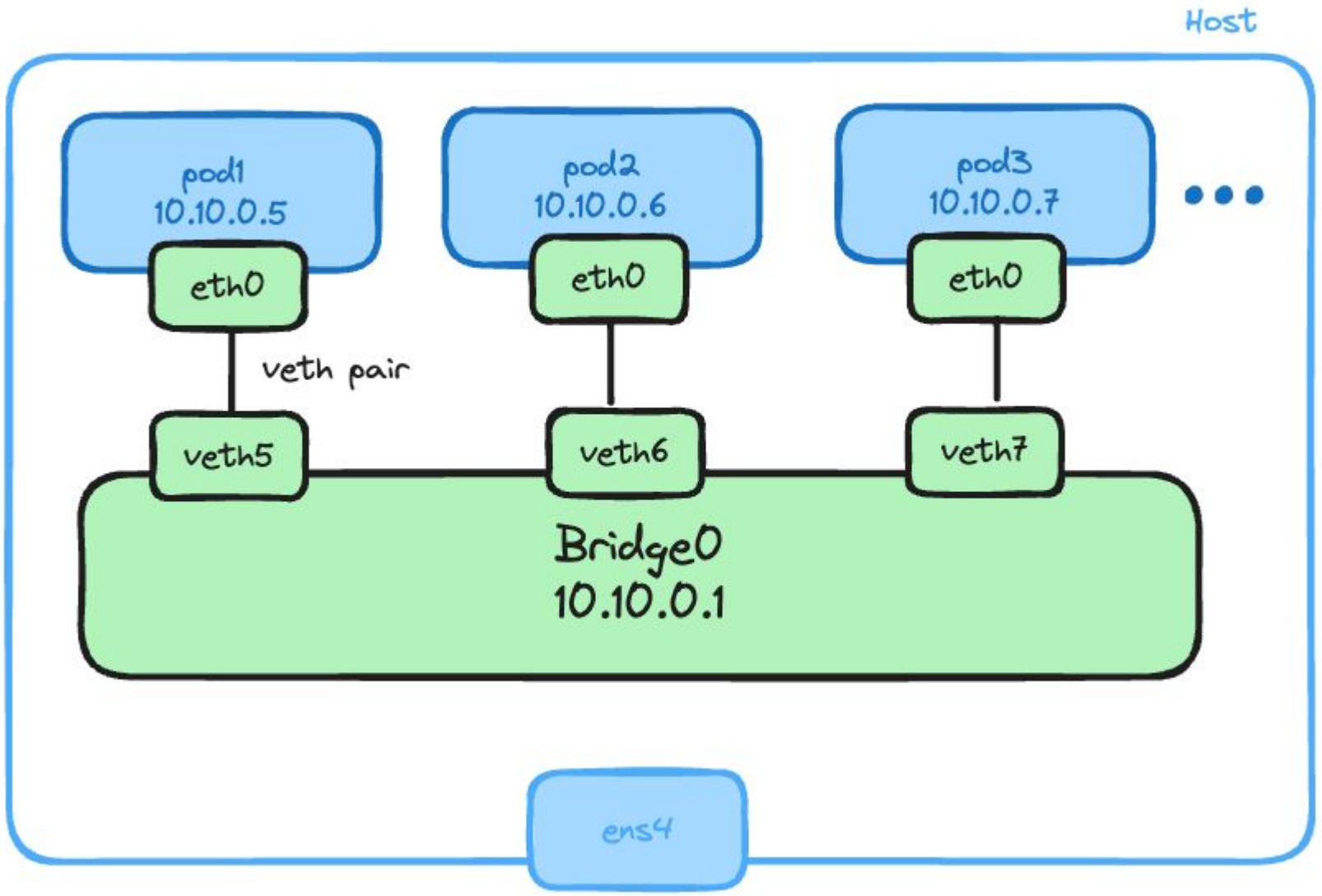



KubeCon | CloudNativeCon

North America 2023

Your First CNI Plugin

Exercise 1 - Bridge Plugin



Lab Environment

bit.ly/ebpfcni


KubeCon | CloudNativeCon

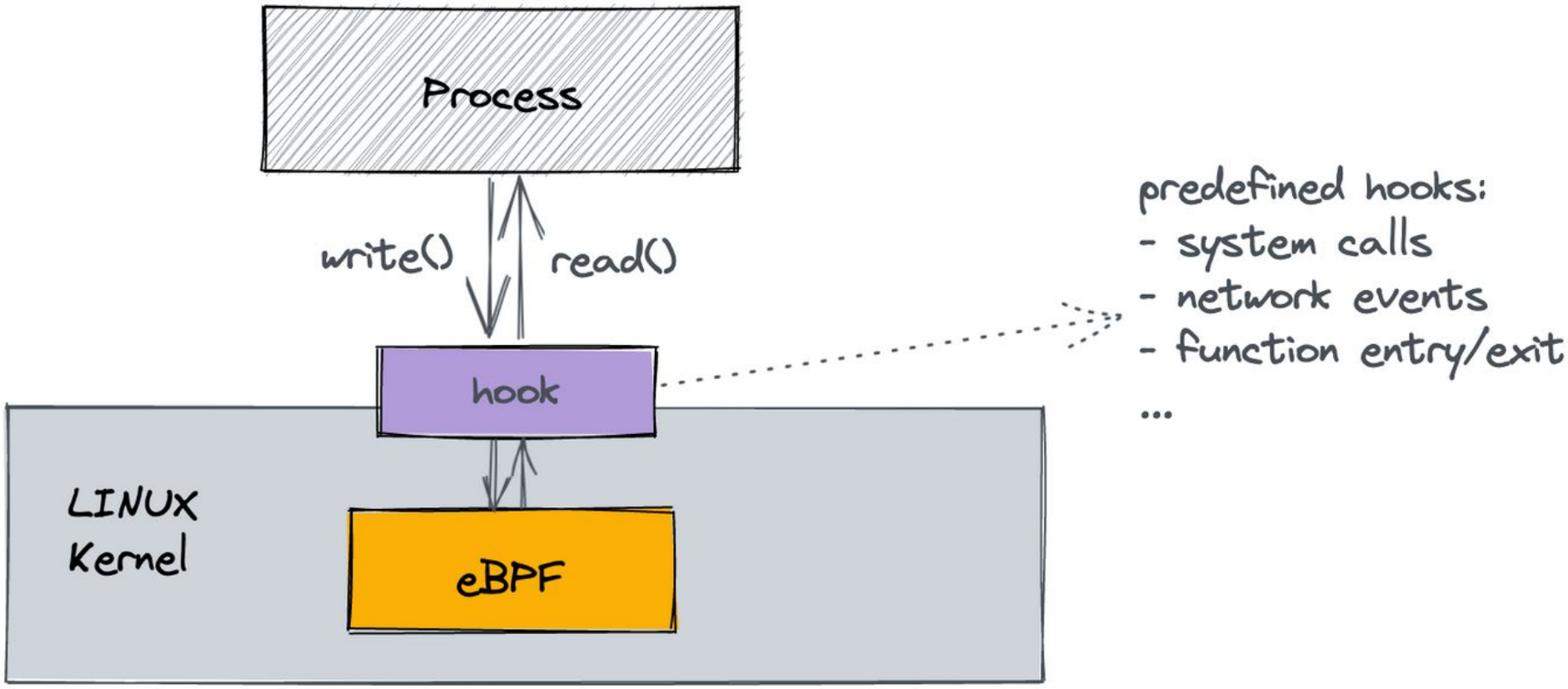
North America 2023

eBPF

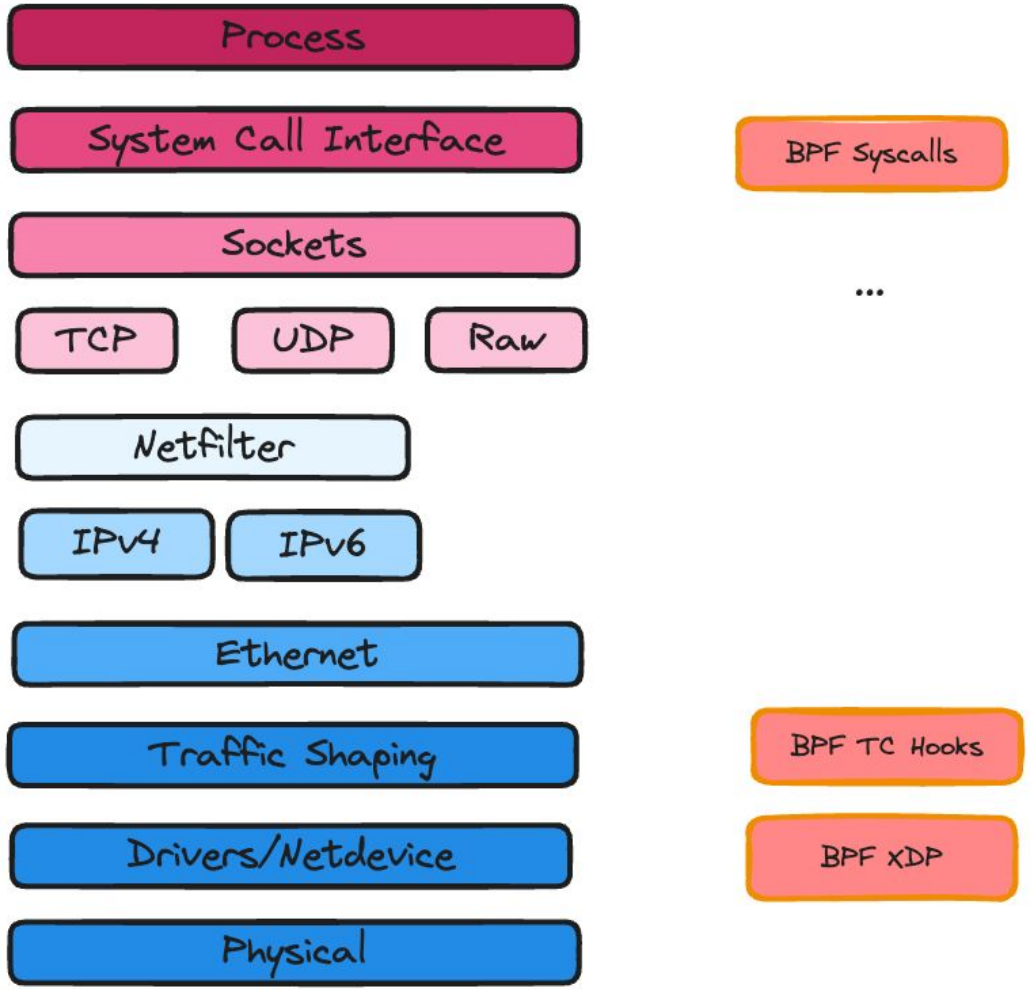
What is eBPF

- Linux technology which enables users to run custom programs “sandboxed” in the kernel
- **e**xtended **B**erkeley **P**acket **F**ilter, evolution of “classic BPF” – think tcpdump
  - ‘eBPF’ & ‘BPF’ will be used interchangeably
- Event-based – programs are attached to “hook points” that are triggered by certain events
  - E.g. ‘kprobe’ type programs are attached to kernel functions and are then executed when that function is called
- BPF programs are verified to be “safe” – won’t crash the kernel, guaranteed to return (no infinite loops), can only access specific sections of memory, etc.

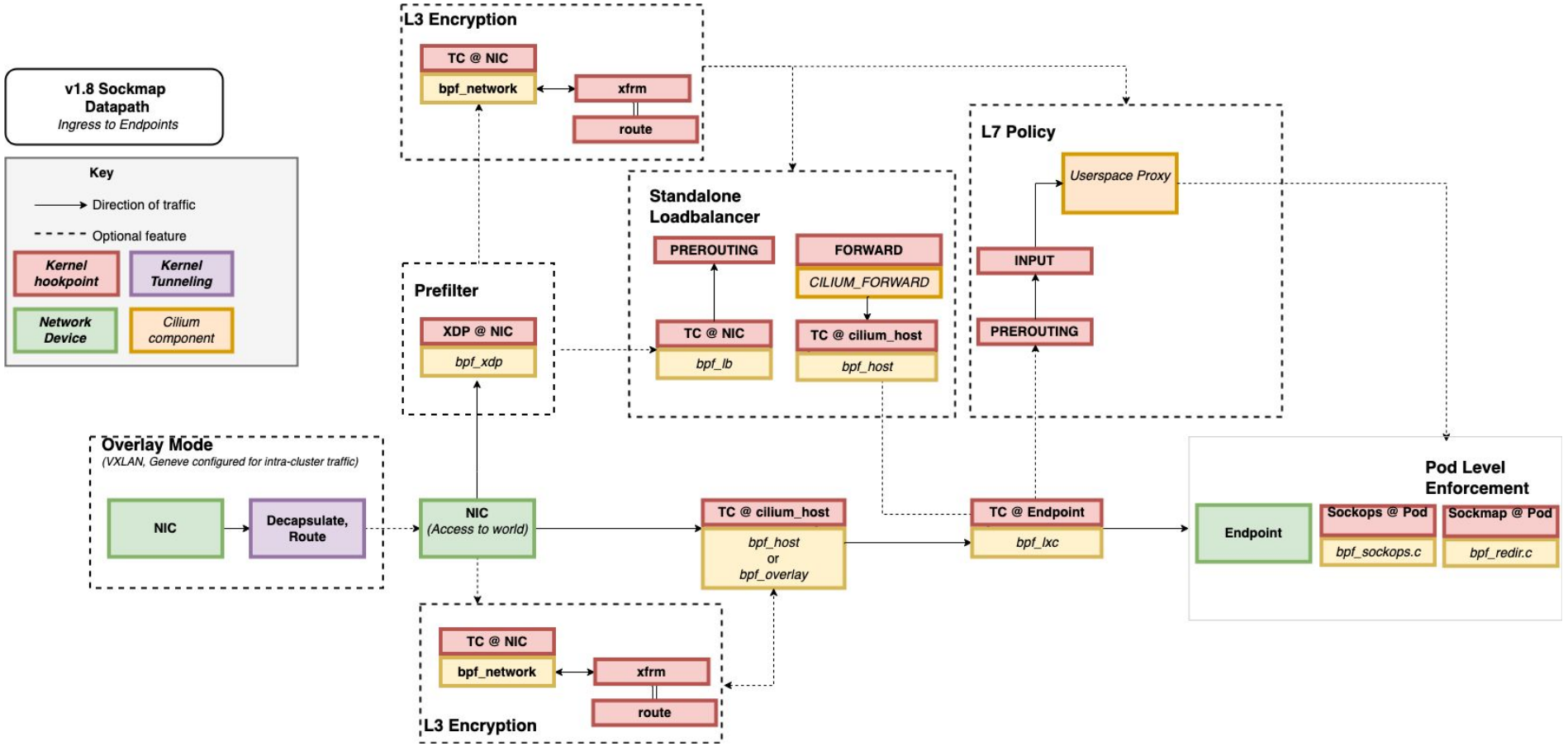
eBPF


Process
write()
read()
hook
LINUX
Kernel
eBPF
predefined hooks:
- system calls
- network events
- function entry/exit
...

xBPF Hooks

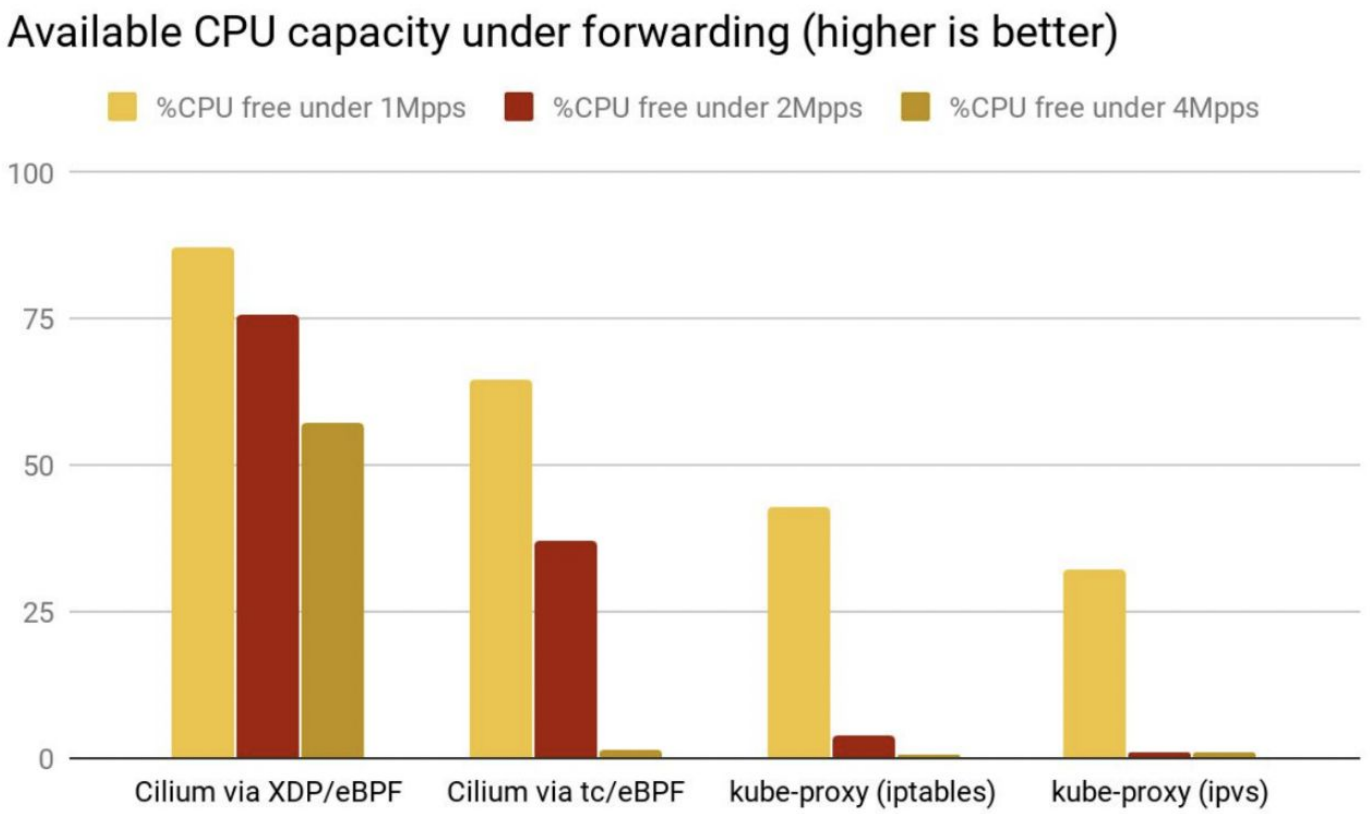

Process
System Call Interface
Sockets
TCP
UDP
Raw
Netfilter
IPv4
IPv6
Ethernet
Traffic Shaping
Drivers/Netdevice
Physical
BPF Syscalls
...
BPF TC Hooks
BPF XDP

eBPF in Cilium



source: https://docs.cilium.io/en/stable/concepts/ebpf/lifeofapacket/#ingress-to-endpoint

eBPF in Cilium



source: https://cilium.io/blog/2020/06/22/cilium-18/


KubeCon | CloudNativeCon

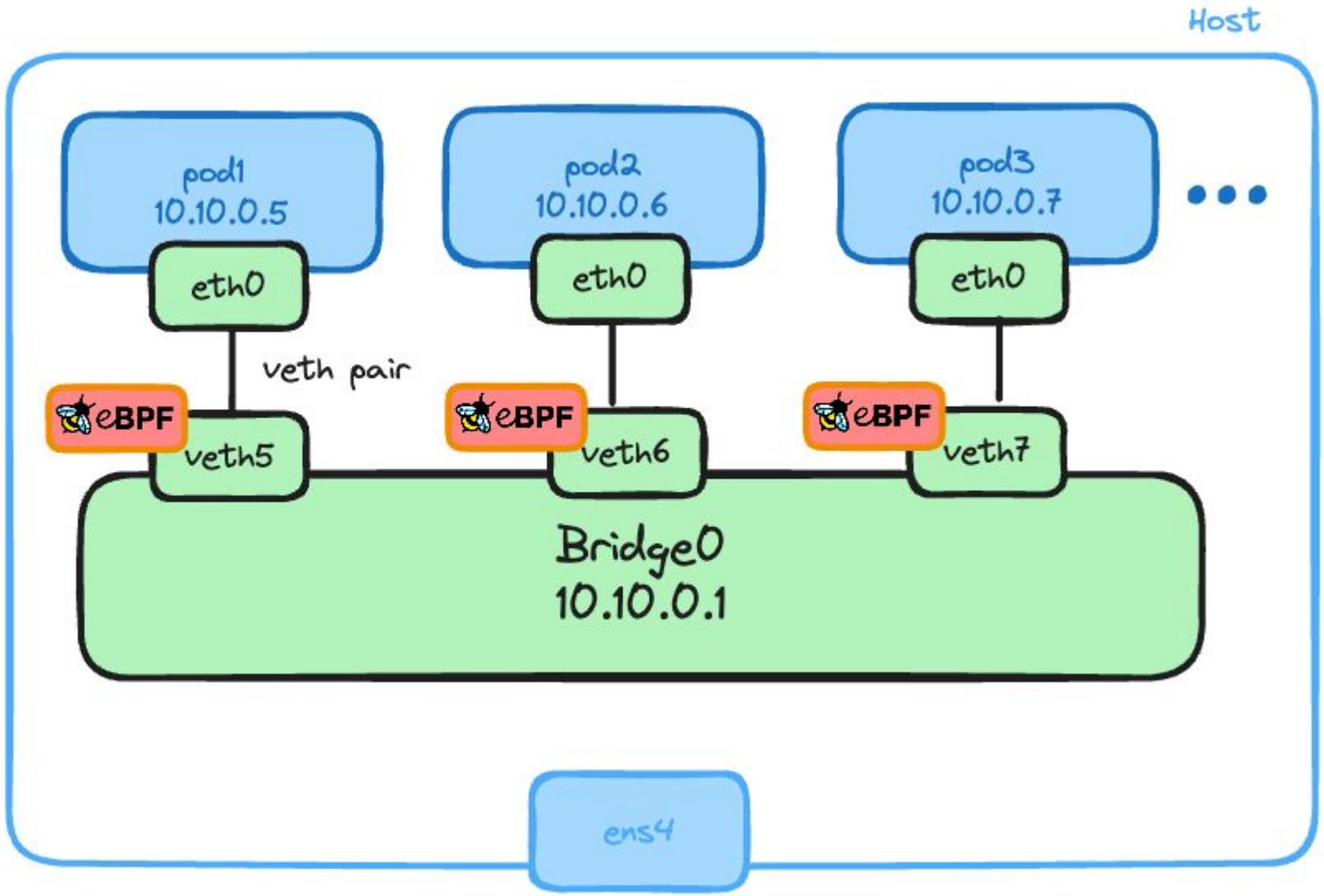
North America 2023

Your First eBPF program

Exercice 2 - eBPF with our CNI plugin

eBPF

Monitor Traffic
Security Policies
Routing



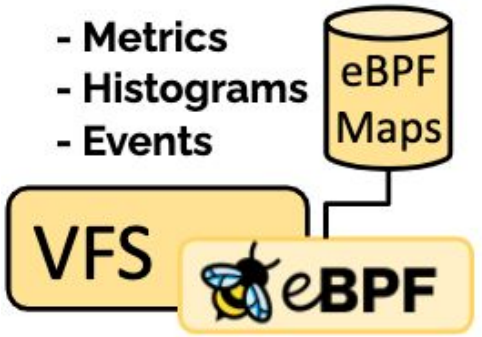

KubeCon | CloudNativeCon

North America 2023

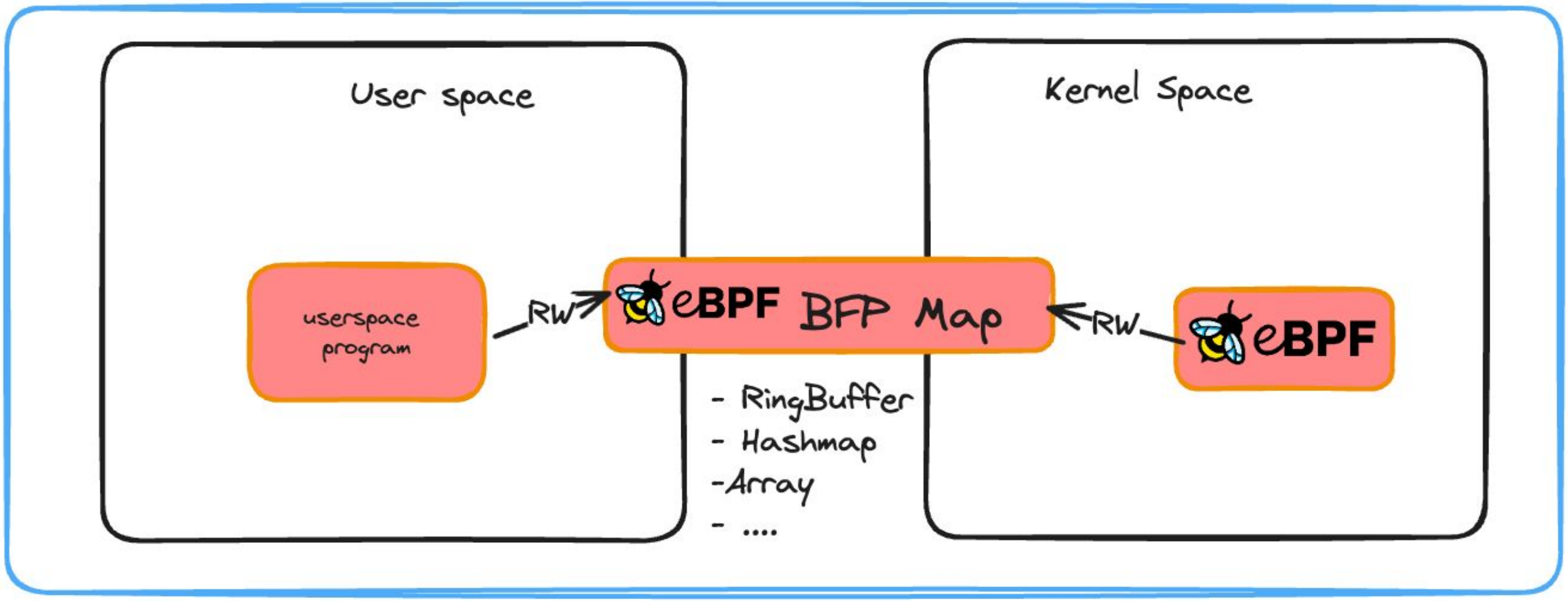
eBPF for Monitoring

Why eBPF for Observability

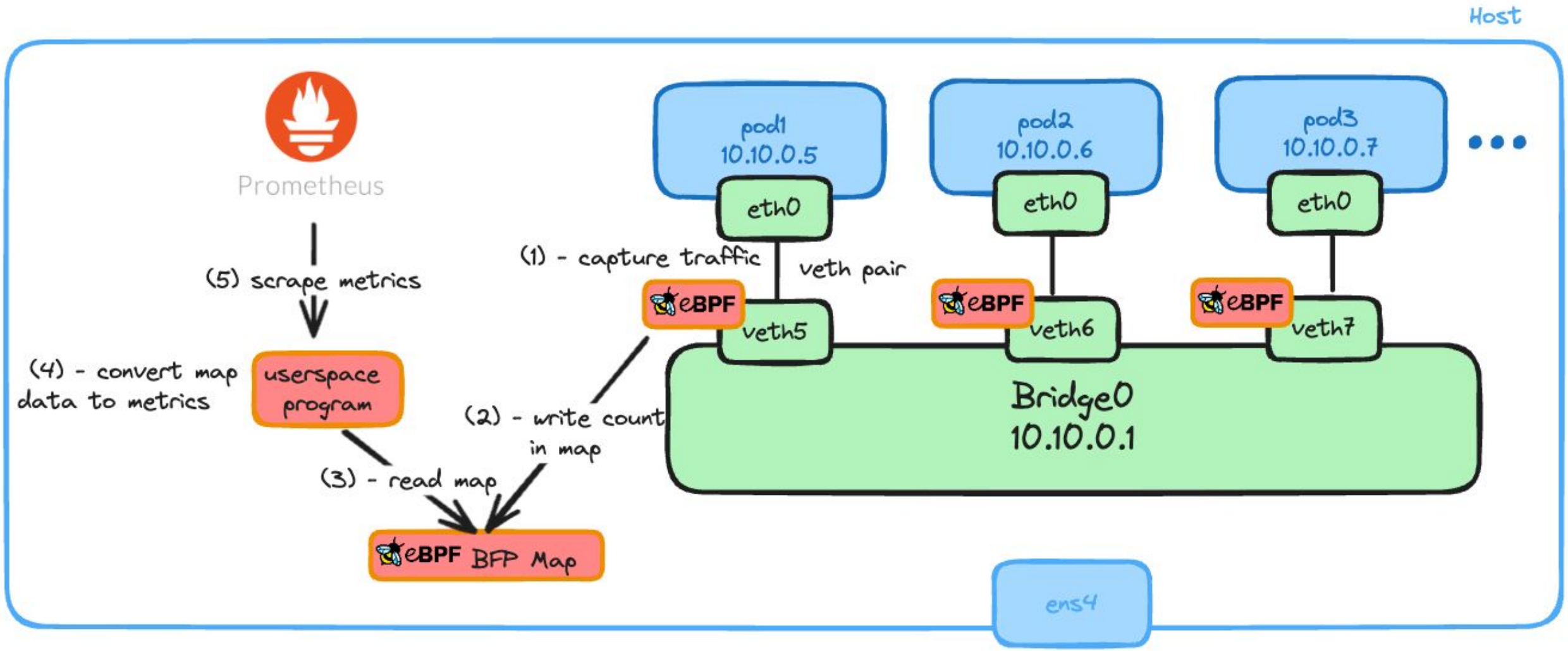
- BPF programs can be attached to almost any kernel function, making it possible to extract data or metrics for observing the state of your systems
- Specific hookpoints (e.g. ‘kprobes’ and ‘tracepoints’) and infrastructure exist to ena BPF-based observability, performance monitoring, and tracing
- Due to eBPF’s efficiency, you can process raw events as they happen in the kernel, enabling visibility that isn’t possible with non-eBPF based solutions



eBPF Maps


User space
Kernel Space
userspace program
RW
eBPF BFP Map
RW
eBPF
- RingBuffer
- Hashmap
-Array
- ....

Exercise 3 - Monitoring



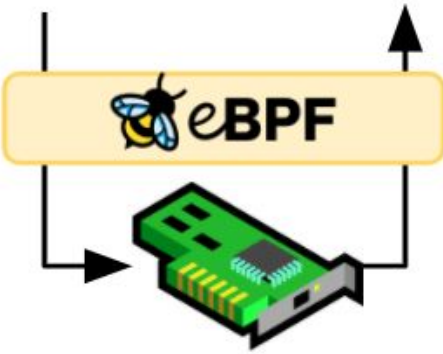

KubeCon | CloudNativeCon

North America 2023

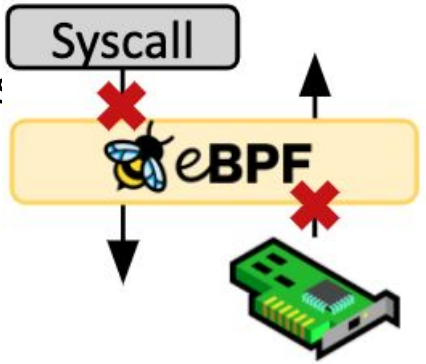
eBPF for Security

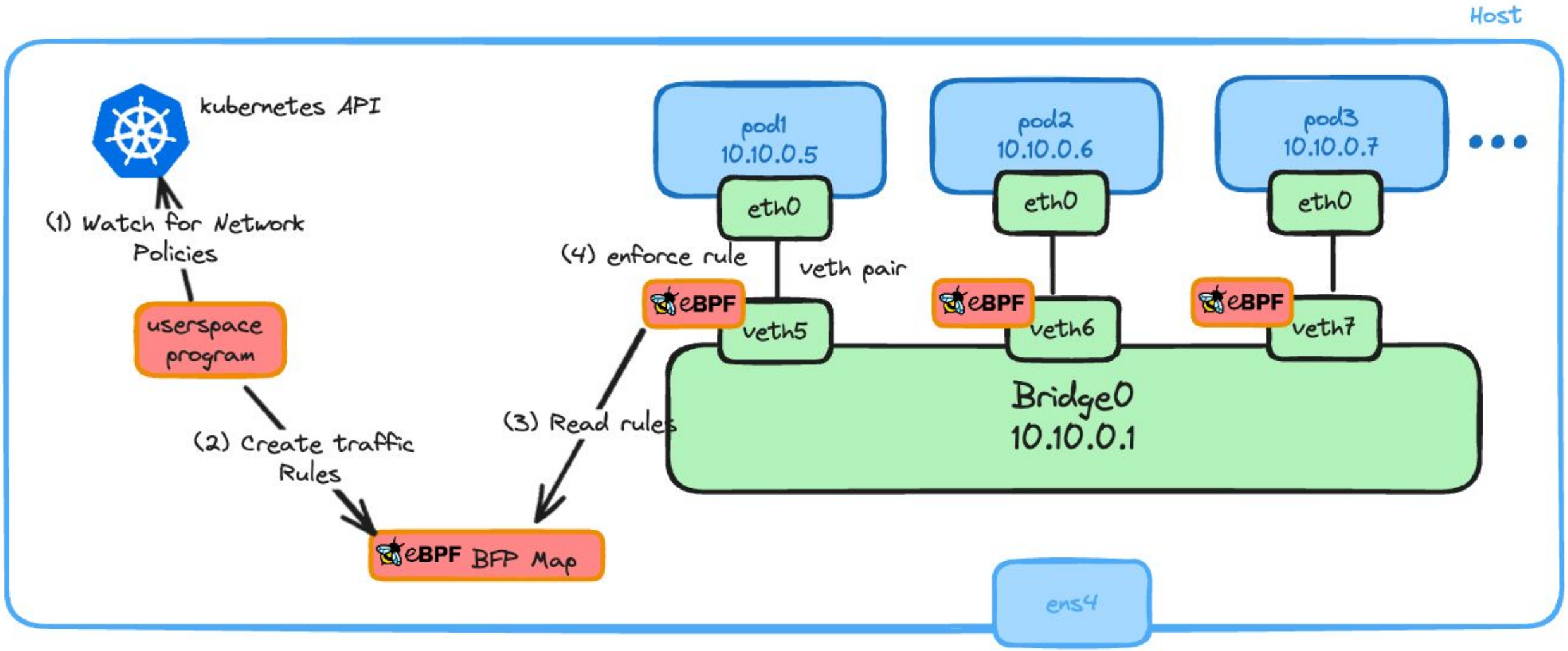
Why eBPF for Pod Networking

- **Networking**
  - As classic BPF was designed for efficiently filtering packets (e.g. tcpdump), eBPF naturally lends itself well to numerous applications in the networking space
  - eBPF enables programmatic processing of packets at several locations in the Linux networking stack



- **Security**
  - Since eBPF programs can be attached to low-level system events, it is a great fit for codifying security policies or monitoring sensitive operations
  - Additionally, as previously mentioned, since eBPF programs can be attached to various network operations, it is a logical fit for enforcing network security




Host
kubernetes API
(1) Watch for Network Policies
userspace program
(2) Create traffic Rules
eBPF BFP Map
(3) Read rules
(4) enforce rule
pod1
10.10.0.5
pod2
10.10.0.6
pod3
10.10.0.7
eth0
eth0
eth0
veth pair
eBPF
eBPF
eBPF
veth5
veth6
veth7
Bridge0
10.10.0.1
ens4

Conclusion

Links

- https://github.com/asayah/ebpf-cni-from-scratch-kubecon-na-2023

- https://github.com/libbpf/libbpf-bootstrap
- https://github.com/cloudflare/ebpf_exporter
- https://github.com/cilium/ebpf
- https://github.com/containernetworking/plugins/

Please scan the QR Code above
to leave feedback on this session