
KubeCon
CloudNativeCon
North America 2024

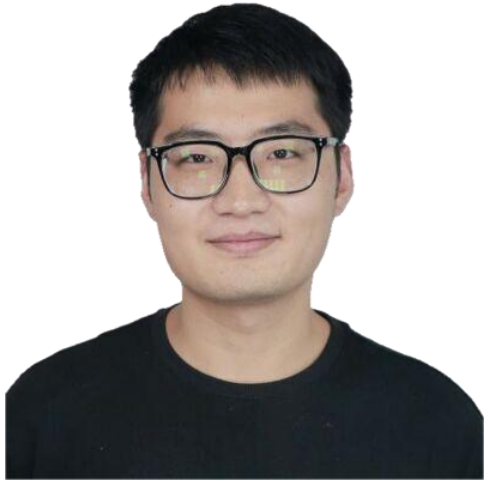

KubeCon | CloudNativeCon
North America 2024

Per-Node Api-Server Proxy: Expand The Cluster's Scale And Stability

Weizhou Lan, Daocloud
Iceber Gu, Daocloud

About Us

Weizhou Lan
From Daocloud
Senior Tech Lead



Icerber Gu（Wei Cai）
From Daocloud
Senior Software Engineer
CNCF Ambassador

Outline

Per-Node APIServer Proxy:
Expand The Cluster's Scale and Stability

1. APIServer is under pressure as the cluster scales up

2. The APIServer Cache Proxy on each node or area

3. Transparently proxy requests based on **ebpf**

What Causes Pressure on APIServer?

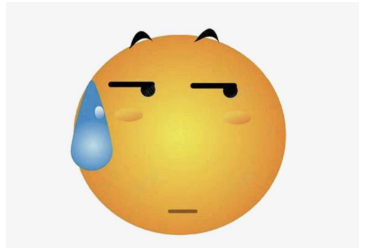
Free
based on OpenAPI

Informer + Declarative
work great

Fragmentation
of the Kubernetes ecosystem

More and More Controllers

Uncontrollable
by Cluster or Ops



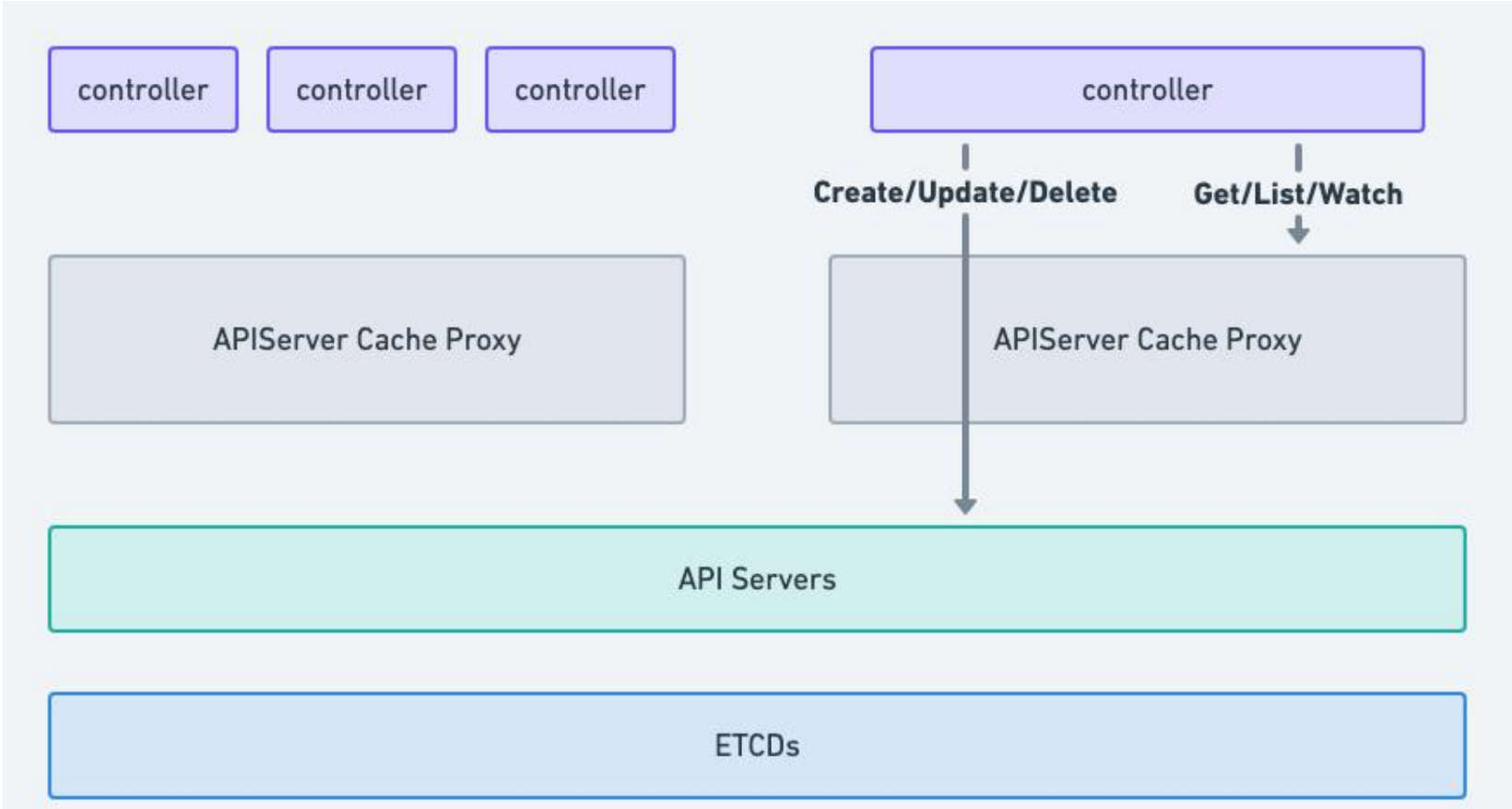
APIServer

Application Controllers and Cluster Scale

As the cluster scales up

- The number of Kubernetes resources and nodes increases, The APIServer will inevitably face more pressure.
- Even common anomalies may cause huge pressure on the APIServer.
- Components from the community may be required to be deployed on every node, further increasing the load on the APIServer.
- Uncontrollable Controllers may also not take into account the stress on the cluster.

Cache and Proxy Your API-Server



Key to Cache: Consistency

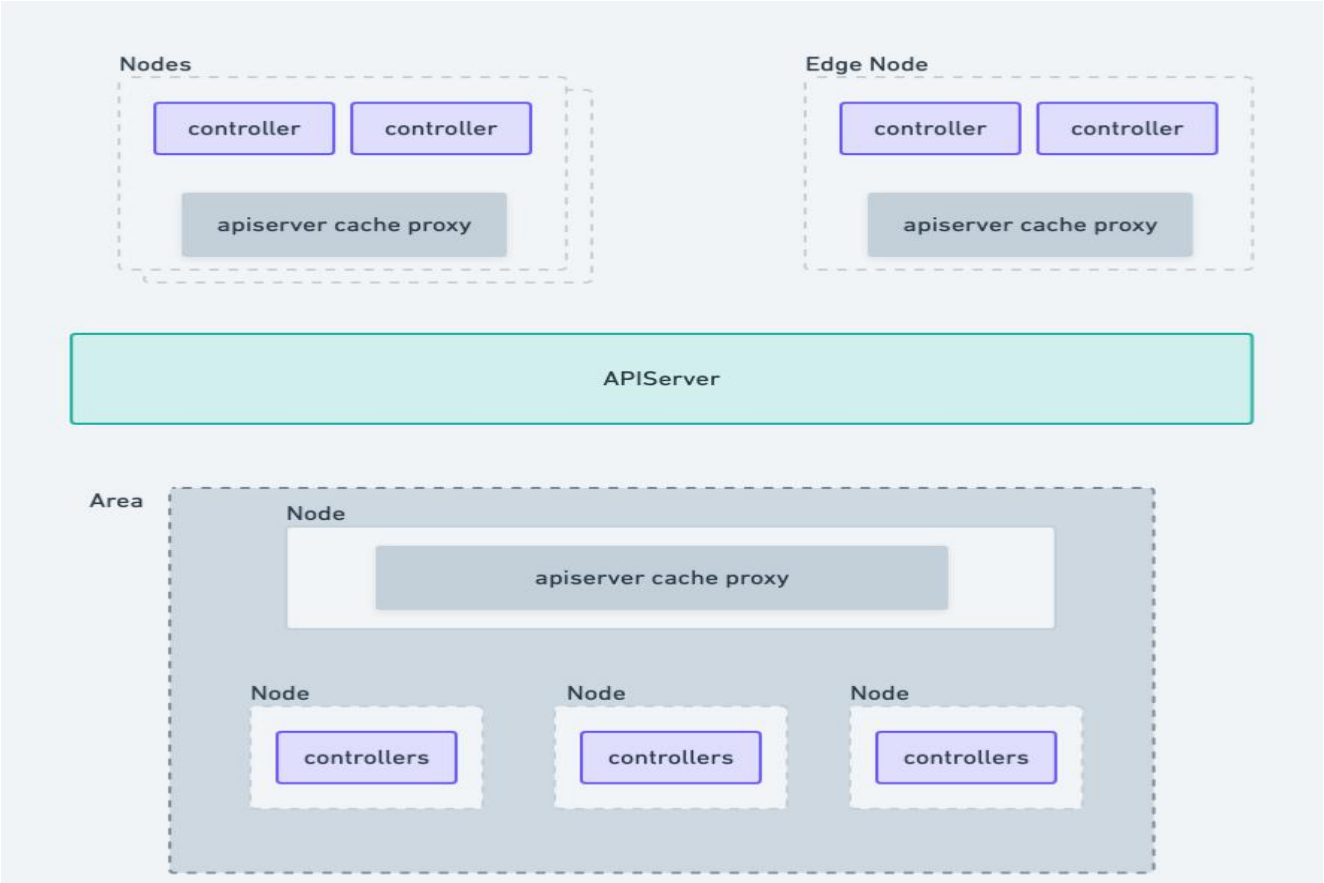
Two Caches in the APIServer Cache Proxy

- a total resource cache built using the informer
- a temporary cache due to write requests from the proxy

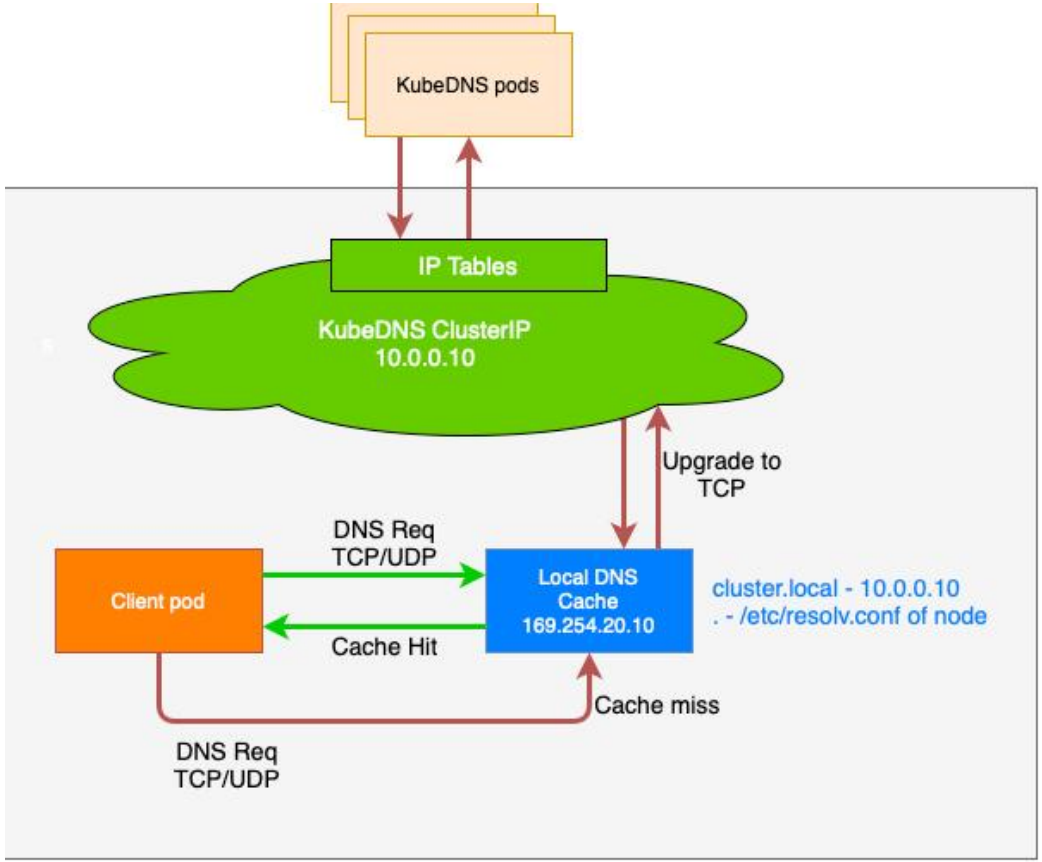
CONSISTENCY

- If you make a write request to a resource through a proxy, the next read request will definitely see the side effects of the previous request.
- Ensuring resource version continuity for List & Watch in case of multiple replicas and concurrency.

Give Your Business Back Its Bandwidth

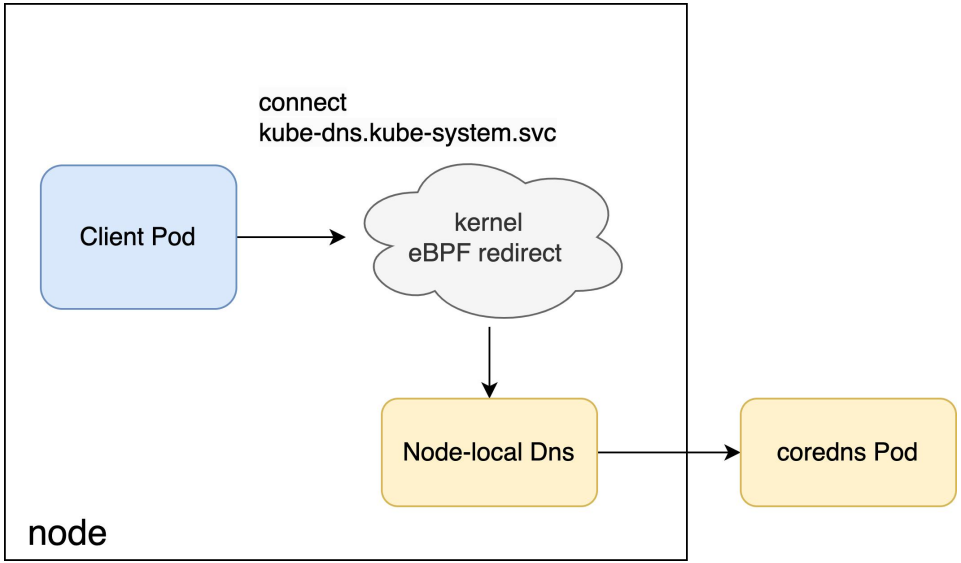


Redirection For Node-local DNS



- **Highly Intrusive**
  Involves the modification of Pod's DNS server
- **Failover Issue**
  Because the Pod's DNS server points to the new address, but when the local DNS Pod fails, DNS requests cannot be switched back to the kube-dns pod

eBPF Redirection For Node-local DNS



Cilium provides the localredirect policy, which perfectly addresses the need for local service redirection.

Additional consideration:

- CNI-agnostic
- A large number of Cilium configuration parameters could overwhelm novices.

```
root@10-20-1-10:~# helm search repo cilium/cilium
NAME            CHART VERSION   APP VERSION     DESCRIPTION
cilium/cilium   1.15.4          1.15.4          eBPF-based Networking, Security, and Observability
root@10-20-1-10:~#
root@10-20-1-10:~# helm show values cilium/cilium | grep -E "^[[:space:]]+[a-zA-Z]+:[[:space:]]+[a-zA-Z]+" | wc -l
276
root@10-20-1-10:~#
```

eBPF has become mainstream

- eBPF has matured and stabilized after years of development
- The mainstream Linux distributions have now adopted high-version kernel
- Outstanding eBPF projects within the fields of networking, security, observability

Features

- High performance
- Strong interaction with user space
- Robust kernel security
- Seamless hot upgrades
- Compile once, run anywhere

Practices

DeepFlow

coroot :~#

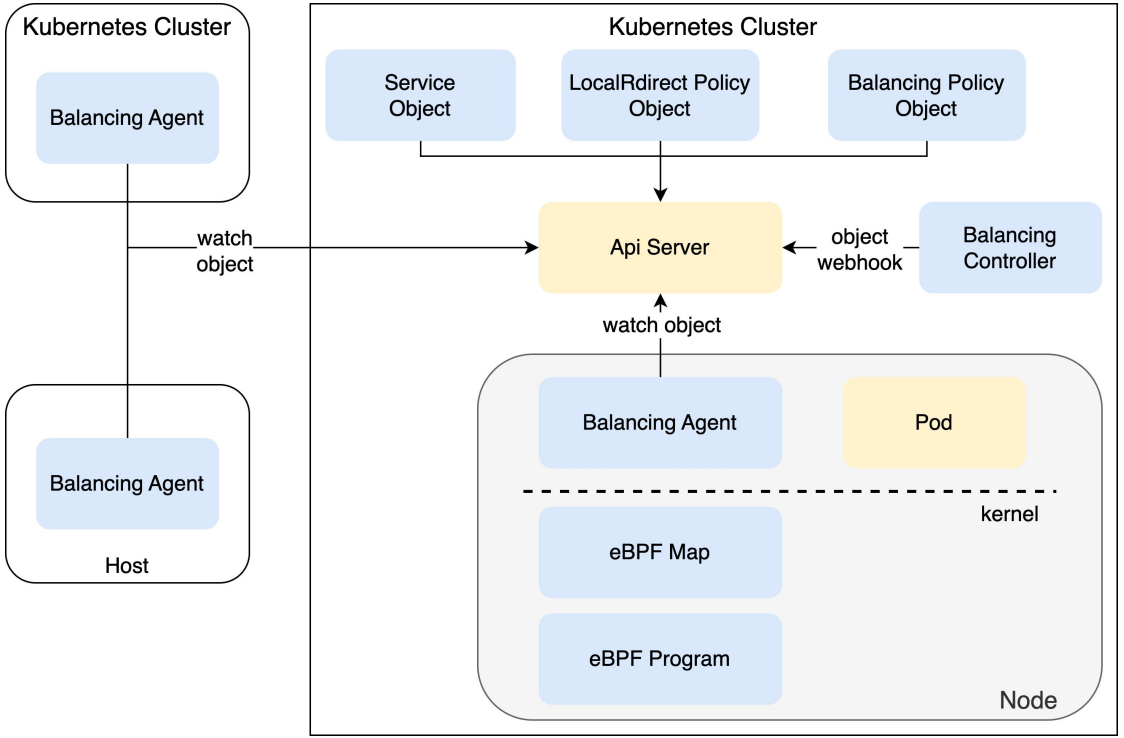
Balancing: Layer-4 eBPF Load Balancing

Balancing
A layer-4 load balancing component implemented with eBPF on Kubernetes

It references projects like cilium, calico, and KPNG. and provides CNI-independent load balancing access capabilities for applications inside and outside the Kubernetes cluster.

- Supports running on the Kubernetes cluster
- Supports running on bare metal
- Offers more than just local redirection functionality

https://github.com/elf-io/balancing



Balancing Features And Use Cases

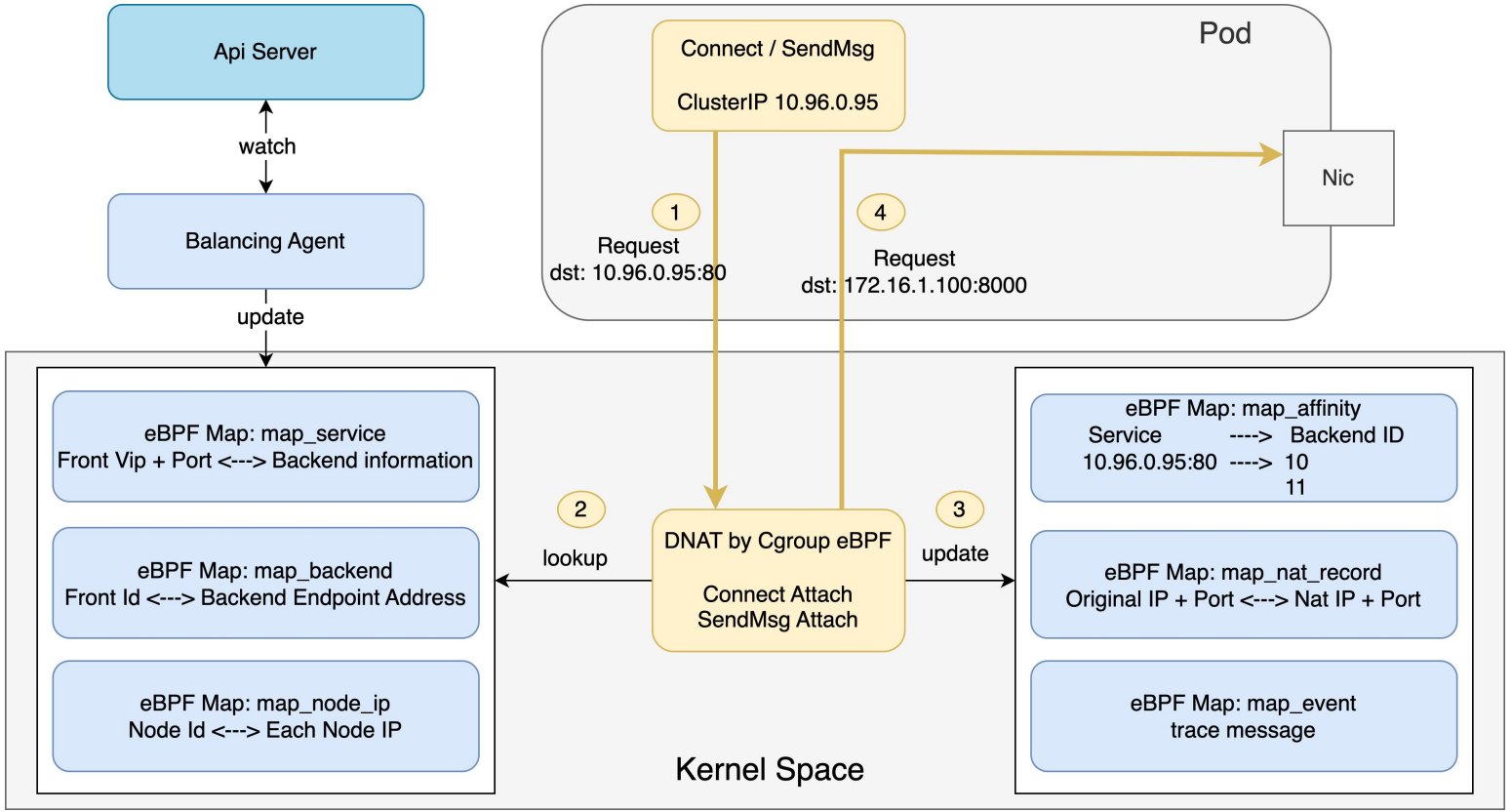
- CNI-agnostic service resolution, serving as “ kube-proxy replacement ” .

- Strengthened Layer-4 local redirection
  Redirect resolution to the service on the same node, and support node labelSelector.
  The typical use case is node-local DNS.

- Global load balancing between intra-cluster and inter-cluster services
  It supports running on kubernetes as pod, and on bare-metal as a container or binary, and enable bidirectional load balancing.
  For examples of use cases, bare metal, KubeVirt, KubeEdge. In future versions, it will also offer solutions for cross-cluster service access.

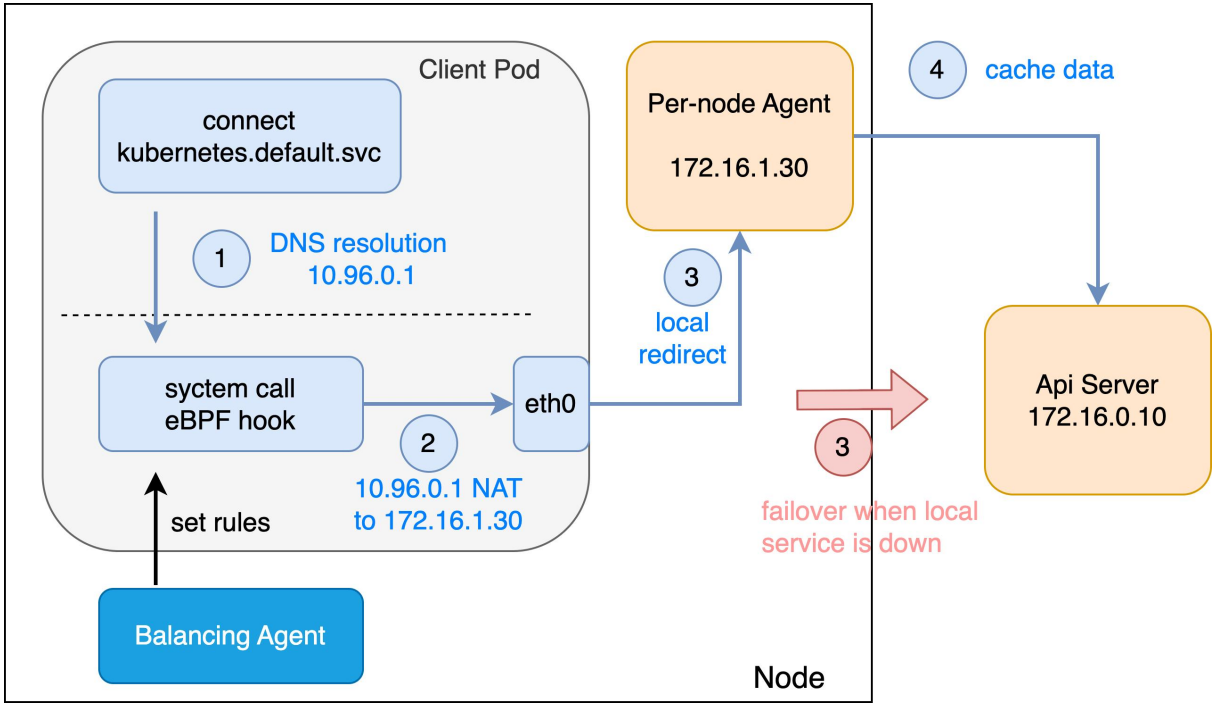
- Customized policy, with flexibility to support customization of frontend and backend IP addresses

Balancing: cGroup eBPF Implements NAT



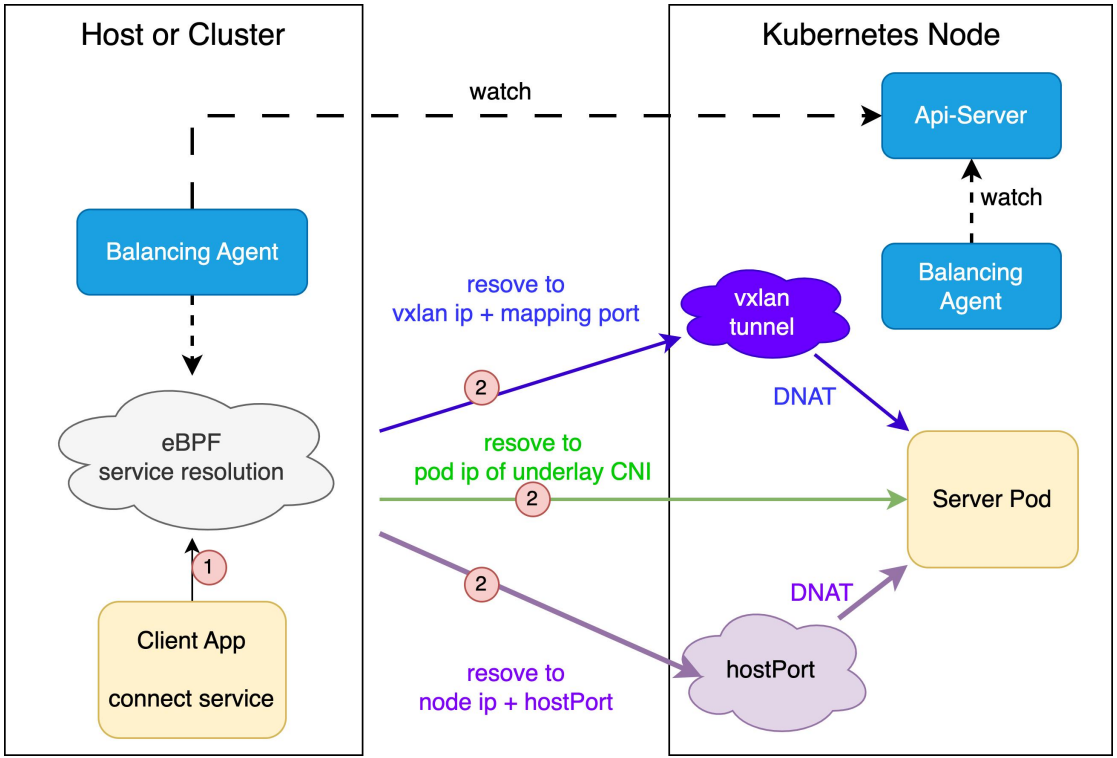
Balancing: Local Redirection

- Transparent Resolution
- High Availability
- CNI-agnostic



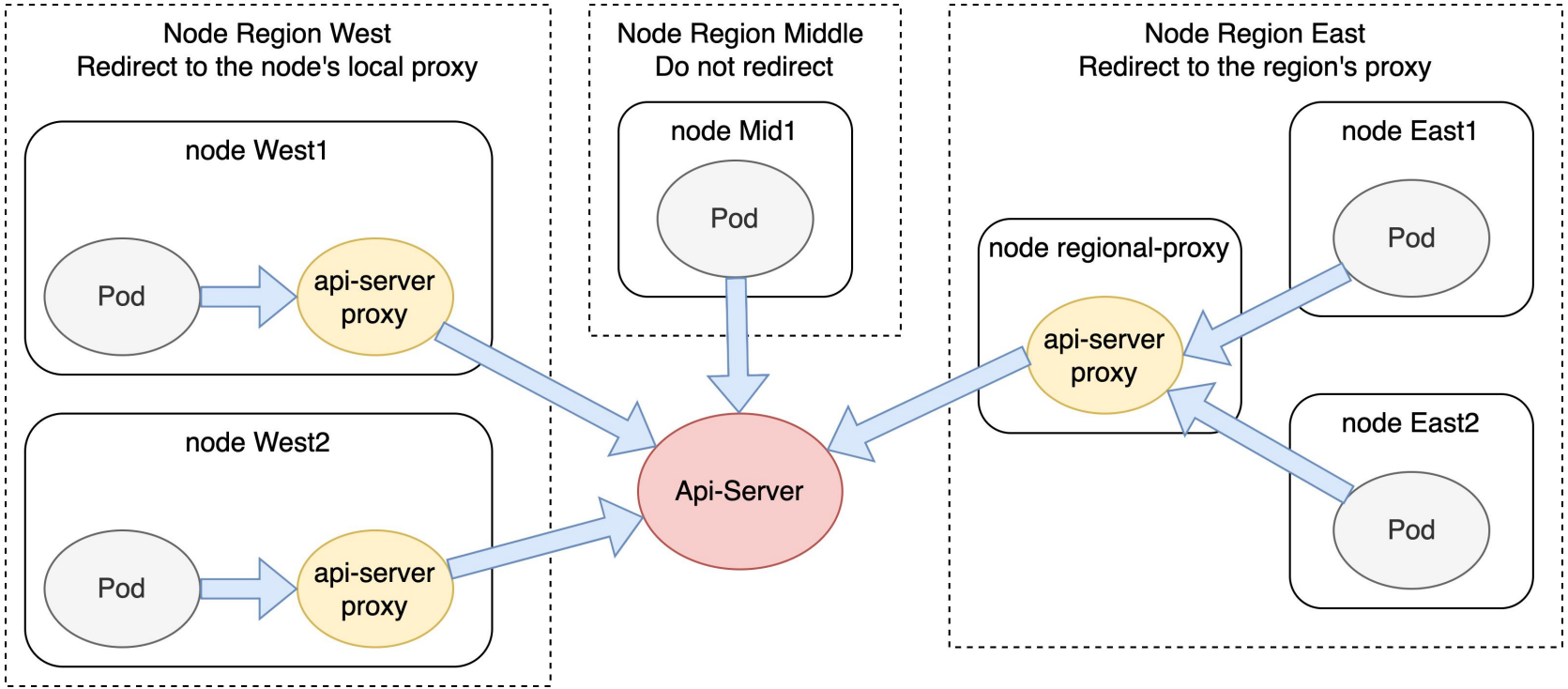

```
apiVersion: balancing.elf.io/v1beta1
kind: LocalRedirectPolicy
metadata:
  name: test
spec:
  frontend:
    serviceMatcher:
      serviceName: kubernetes
      namespace: default
      toPorts:
        - port: "443"
          protocol: TCP
          name: p1
  backend:
    endpointSelector:
      matchLabels:
        app: proxy-redirect
    toPorts:
      - port: "443"
        protocol: TCP
        name: p1
```

Balancing: Global Redirection




```
apiVersion: balancing.elf.io/v1beta1
kind: BalancingPolicy
metadata:
  name: test-service-podendpoint
spec:
  config:
    enableOutCluster: true
    nodeLabelSelector:
      matchLabels:
        region: balancing
  frontend:
    serviceMatcher:
      serviceName: http-server
      namespace: default
      toPorts:
        - port: "8080"
          protocol: TCP
          name: p1
  backend:
    serviceEndpoint:
      endpointSelector:
        matchLabels:
          app: http-redirect
      redirectMode: podEndpoint
      toPorts:
        - port: "80"
          protocol: TCP
          name: p1
```

Balancing: Multiple Strategies For Redirection



Infinite Possibilities for Proxy Requests

- Join untrusted nodes to the cluster
- The Truman Show in Kubernetes: Virtual environments at the apiserver level

Thanks

Presentation
Feedback

Api-Server Proxy

Balancing
eBPF Loadbalancing