IstioCon

IstioCon

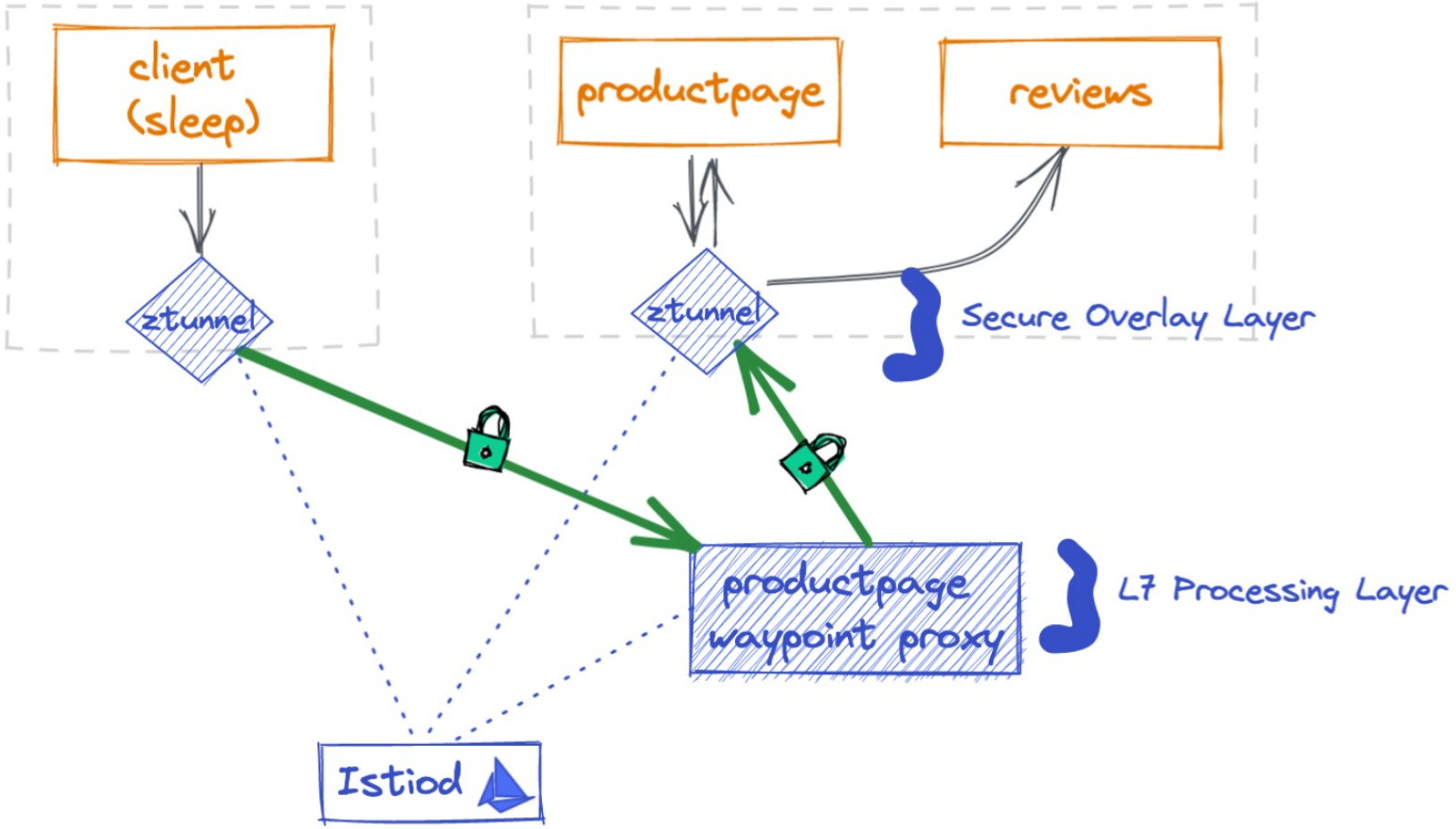
Building an Efficient Service Mesh: Merbridge's Innovations in eBPF Implementation and Istio Ambient

Kebe Liu

What is Ambinent Mesh?

IstioCon

Ambient is a new data plane proxy mode that, through node proxy + Waypoint mode, can reduce resource usage and lower the difficulty of use compared to the Sidecar mode.

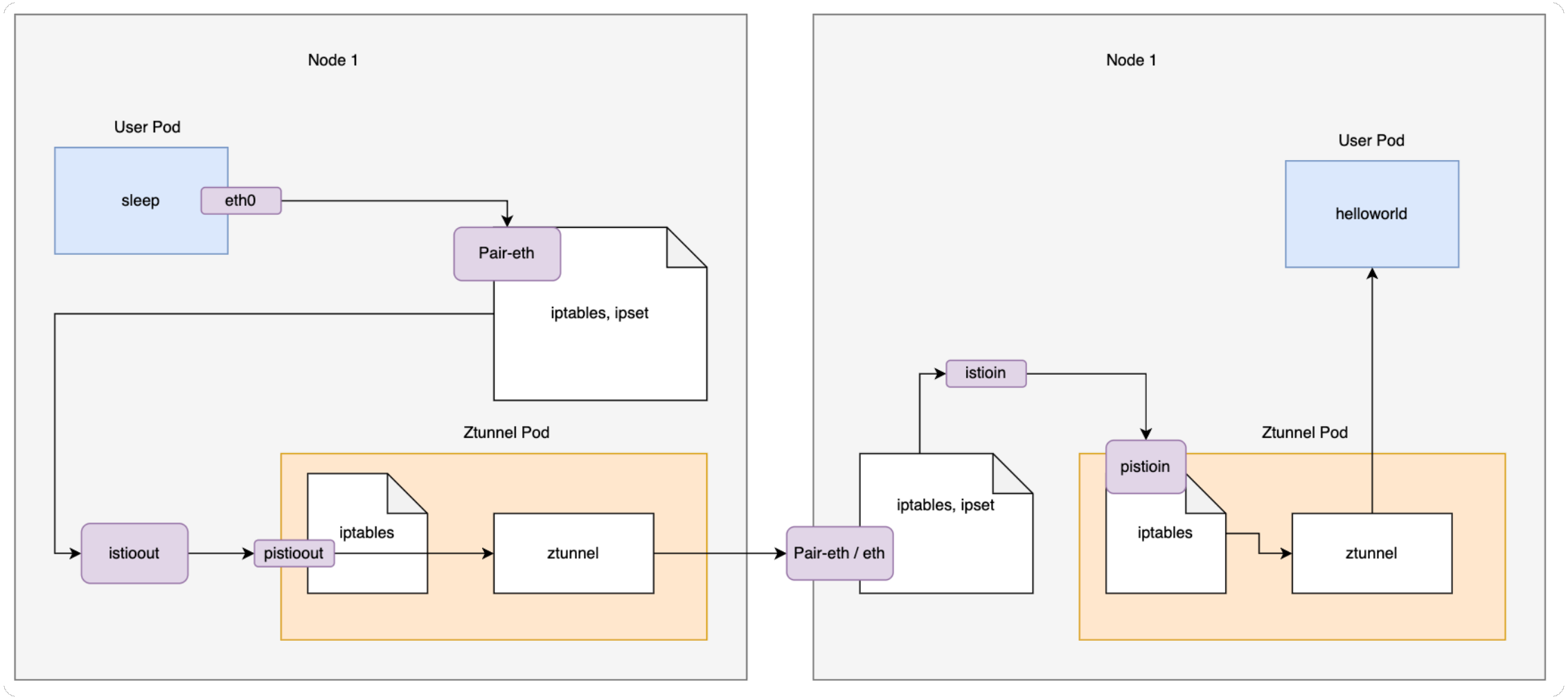


From: https://istio.io/latest/blog/2022/introducing-ambient-mesh/

Ambient mode traffic forwarding path

IstioCon

kubectl exec deploy/sleep – curl helloworld:5000/hello Traffic path



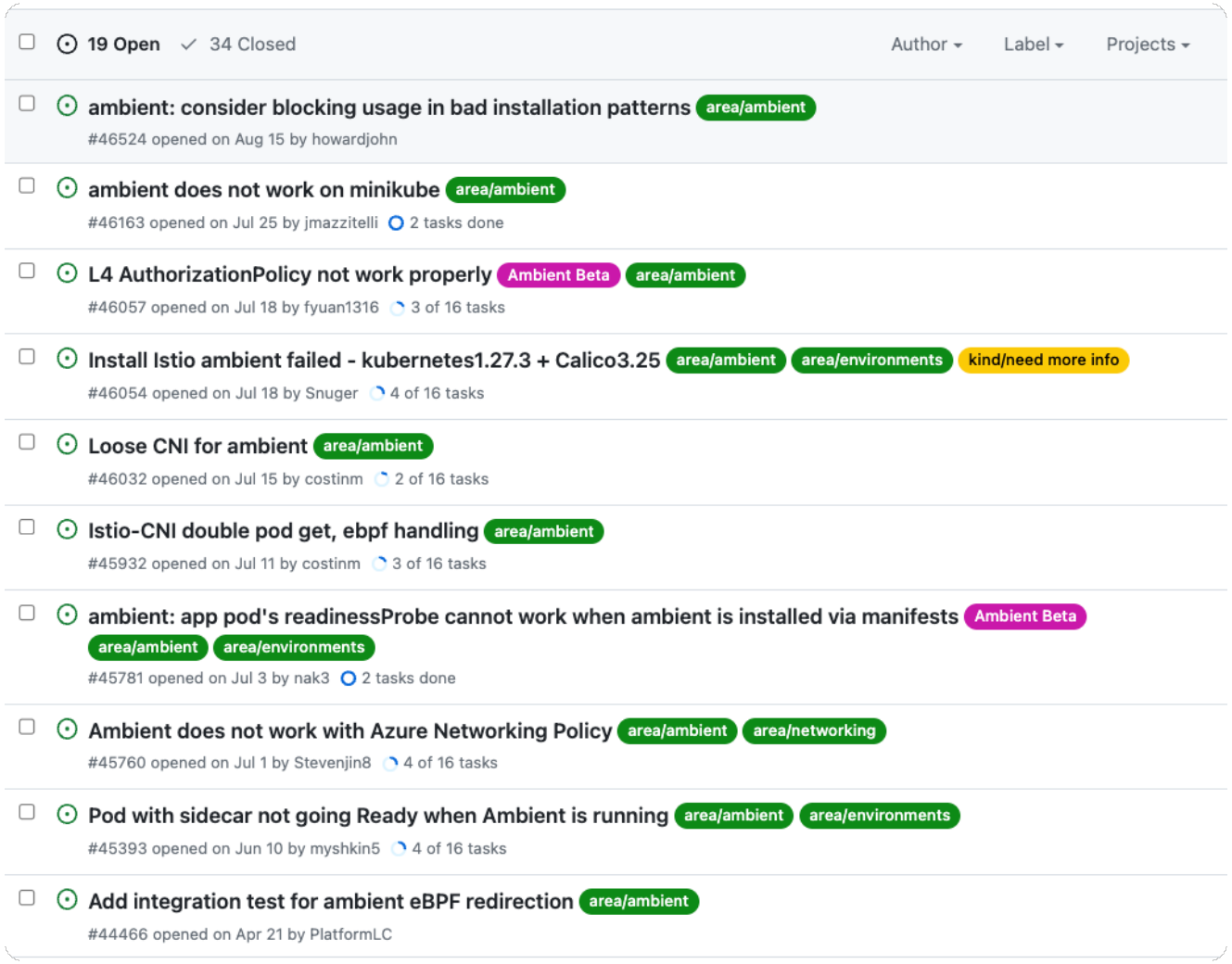
Ambient mode challenges

IstioCon

Because the Ambient mode uses iptables by default to control and forward traffic on the host, and many CNIs also manage container networks through policy routing or iptables, it can easily cause policy conflicts and become unavailable.

Therefore, the Ambient mode cannot currently adapt to all CNIs.

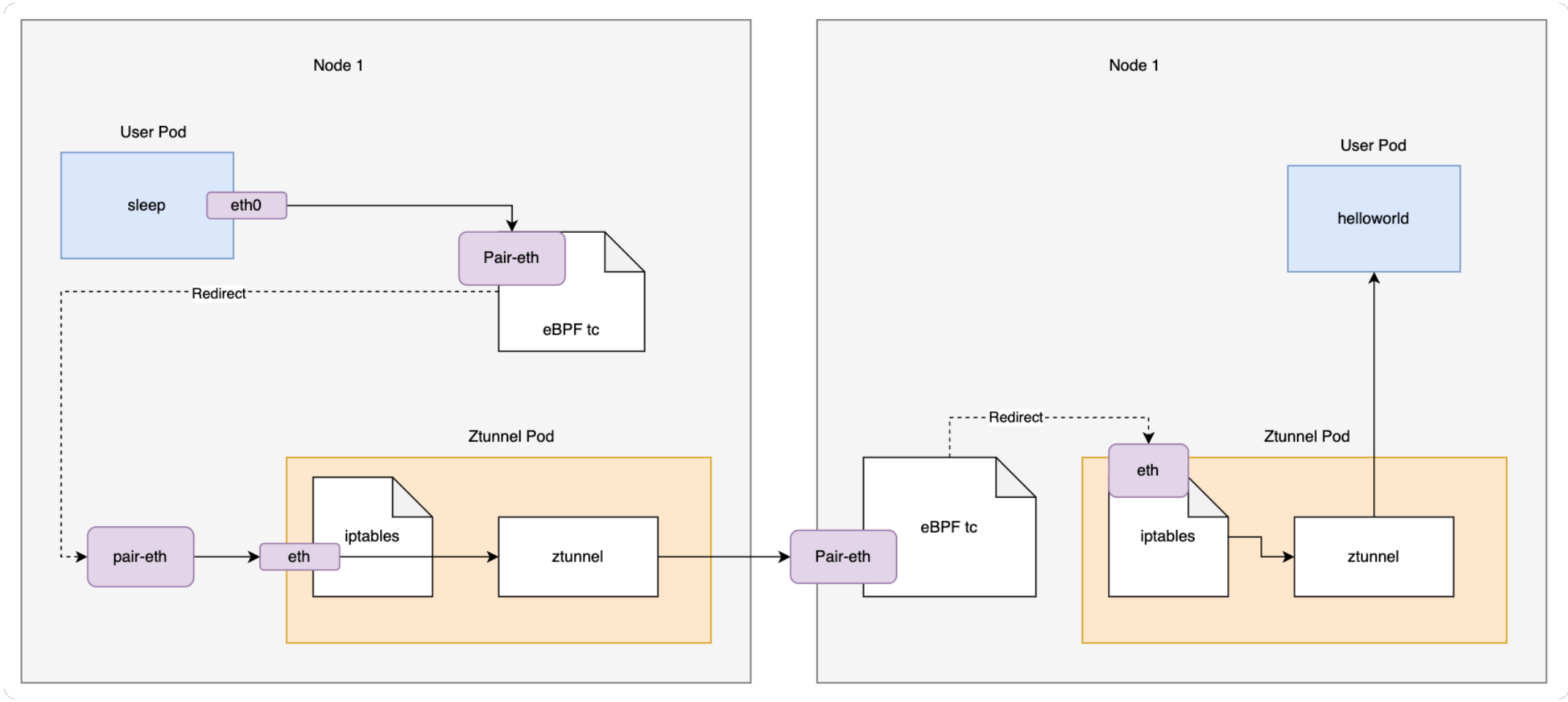
Based on this, the community has added support for eBPF to bypass iptables for traffic forwarding.



Using eBPF and TC to Optimize Traffic Paths

IstioCon

kubectl exec deploy/sleep – curl helloworld:5000/hello Traffic path with eBPF / TC



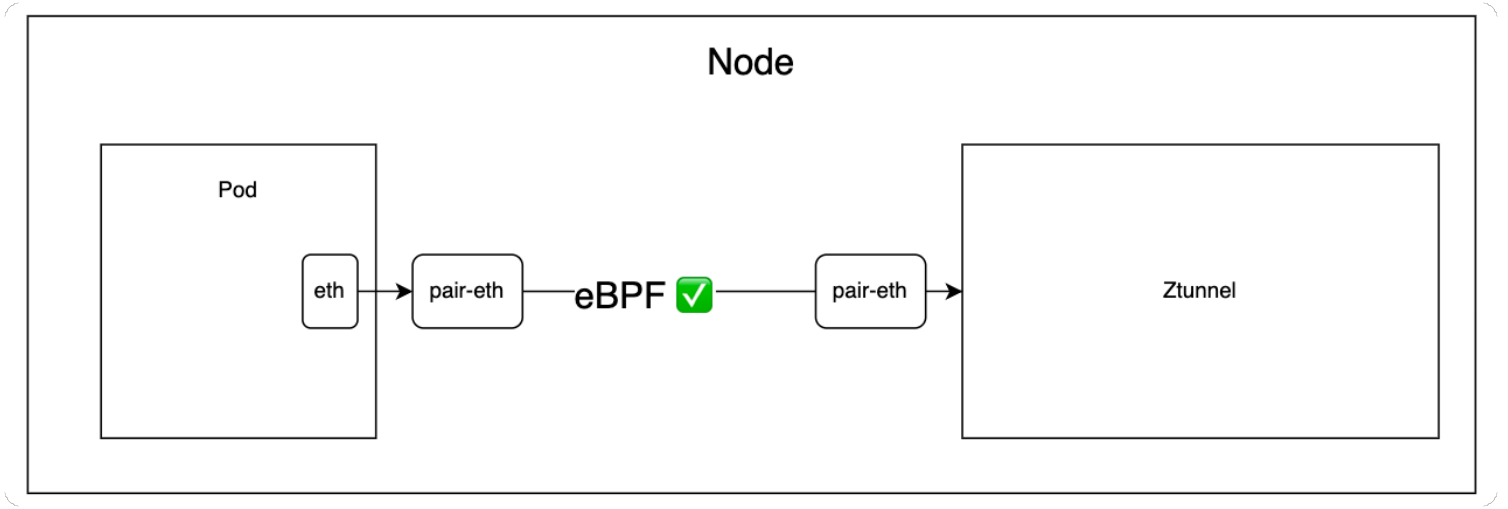
Problems with Ambient eBPF Mode

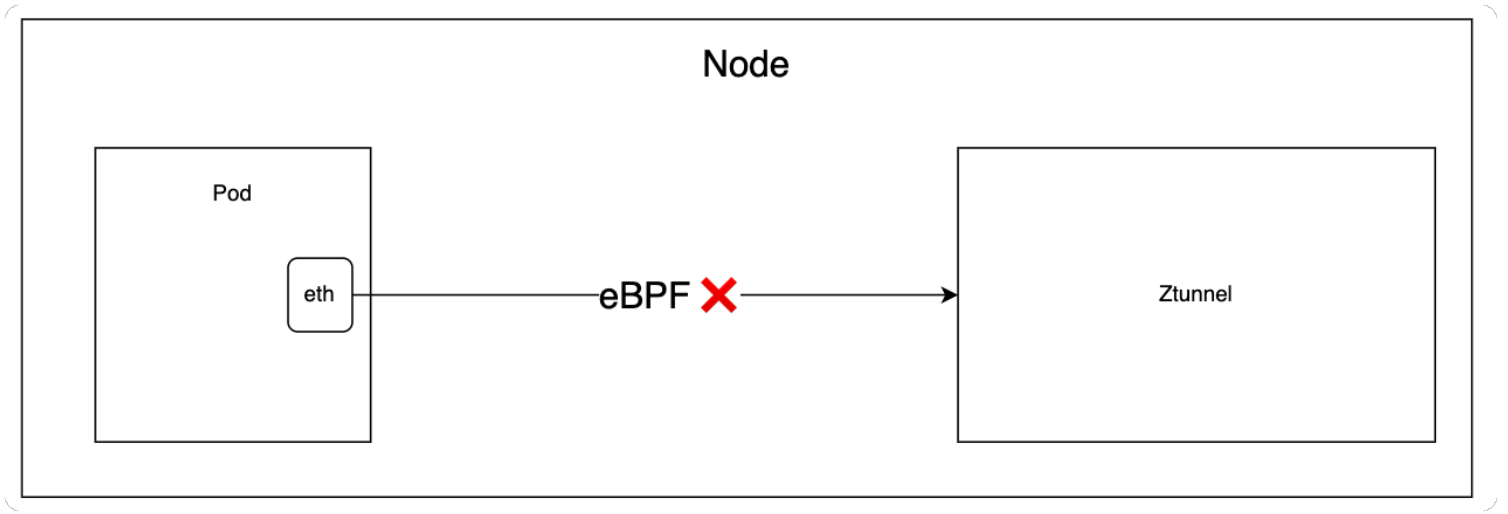
IstioCon

Ambient's eBPF mode, by applying eBPF programs on the pair network interface of the Pod, forwards traffic to the pair network interface of ztunnel, thereby allowing traffic to enter ztunnel and replacing the functionality of iptables. This approach can alleviate the difficulty of supporting CNI.

However, this mode relies on CNIs that use the pair network interface mode. For some CNIs implemented in macvlan mode, since there is no pair network interface on the host, this mode cannot be used.

(PS: The core reason is that tc redirect cannot cross network namespaces.)

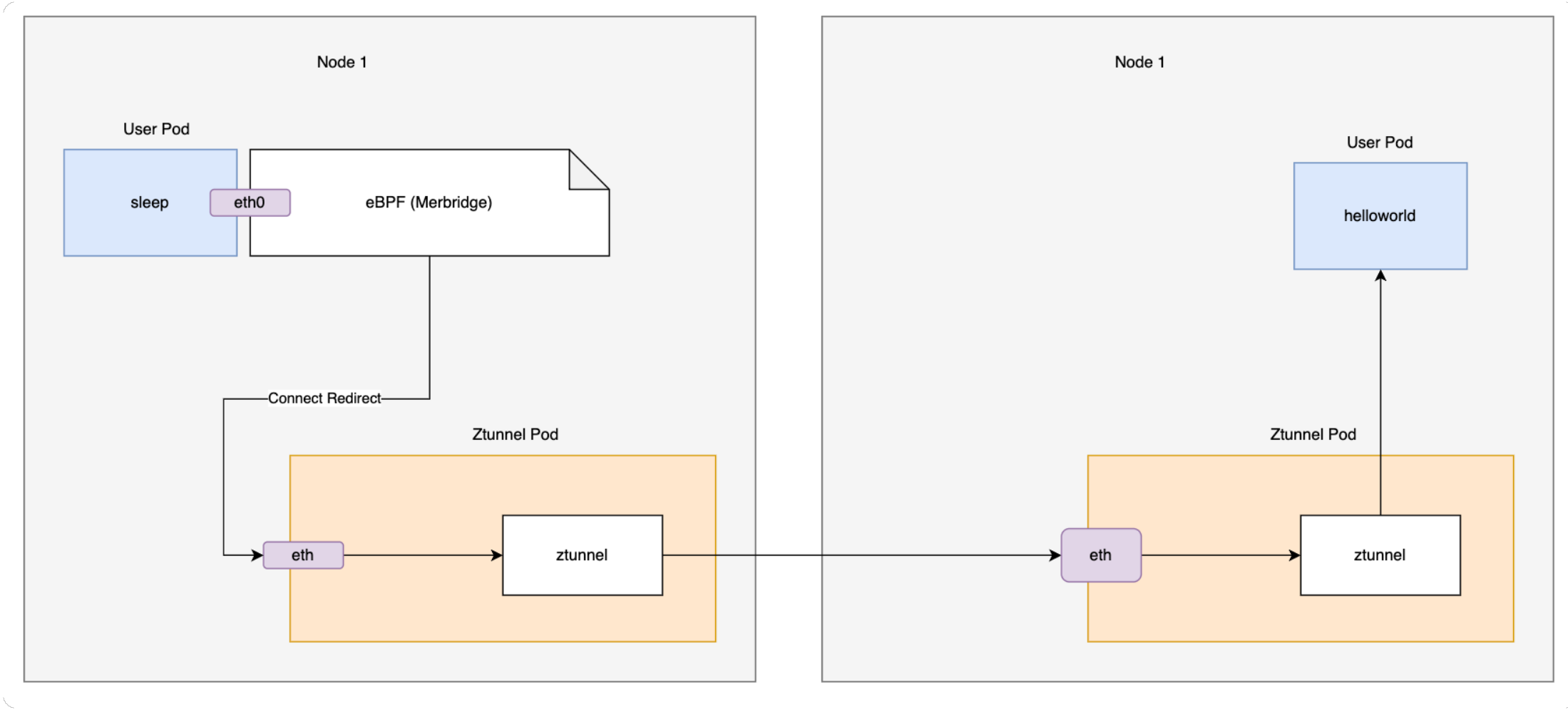




Using Merbridge to Enhance Ambient

IstioCon

kubectl exec deploy/sleep – curl helloworld:5000/hello Traffic path with Merbridge

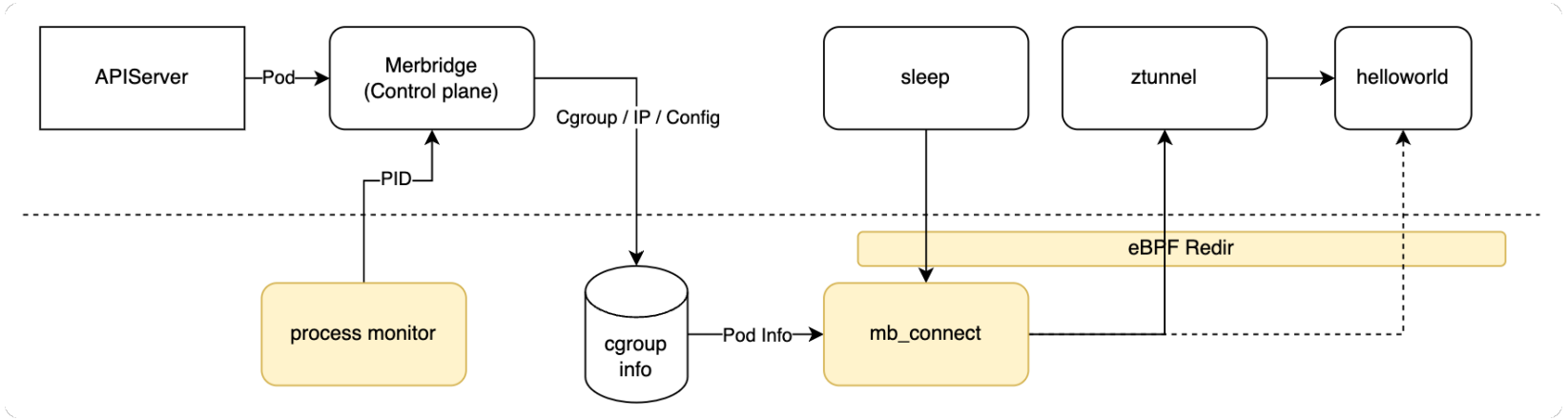


How to handle traffic forwarding?

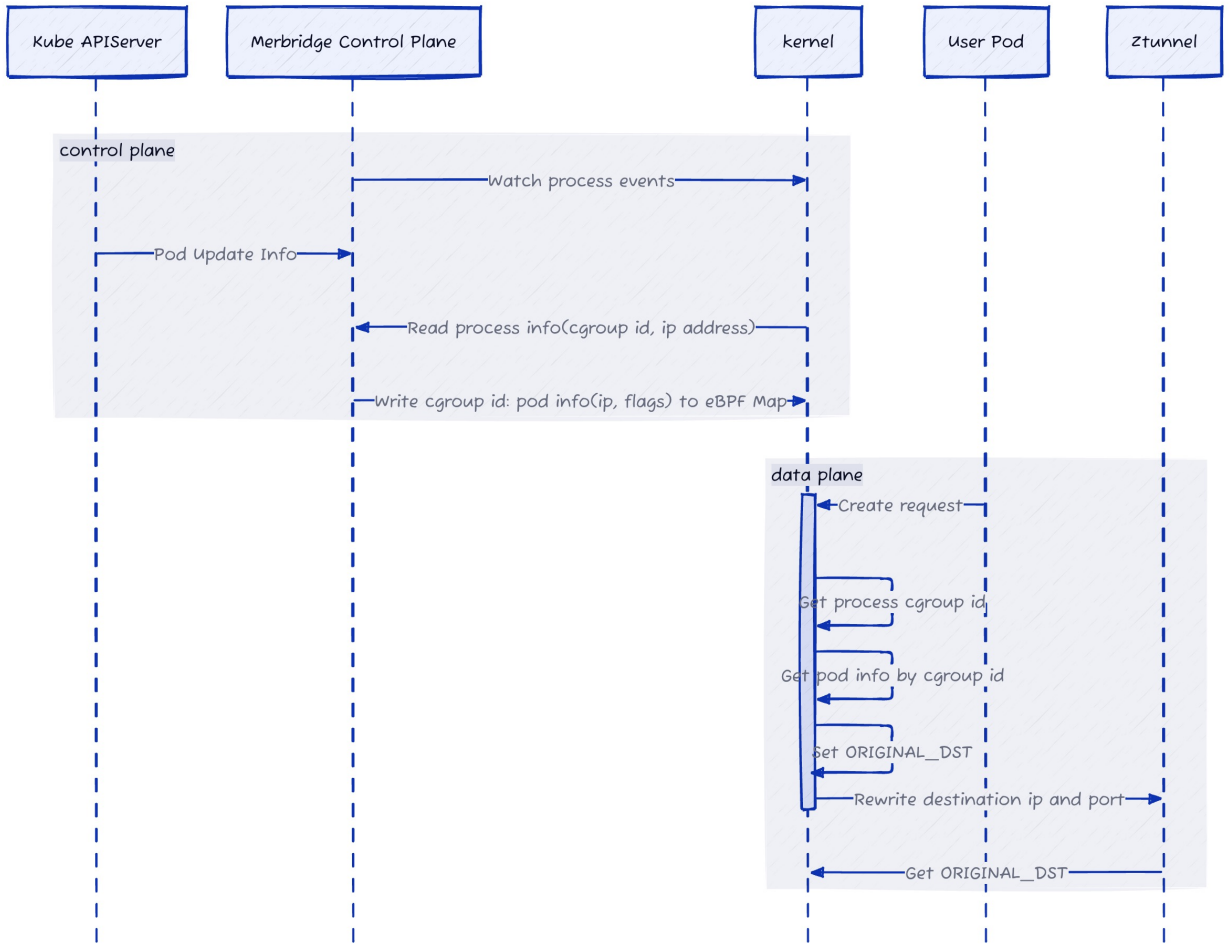
IstioCon

In Ambient mode, by matching whether the source IP of the traffic is in Ambient mode on the host, it is determined whether to forward the traffic to Ztunnel.

However, the eBPF program of Merbridge will directly determine whether the traffic needs to be forwarded to ztunnel when the process in the Pod initiates a request (before it is processed by the kernel protocol stack and the source IP is not bound yet).



How to handle traffic forwarding?



Some known issues

IstioCon

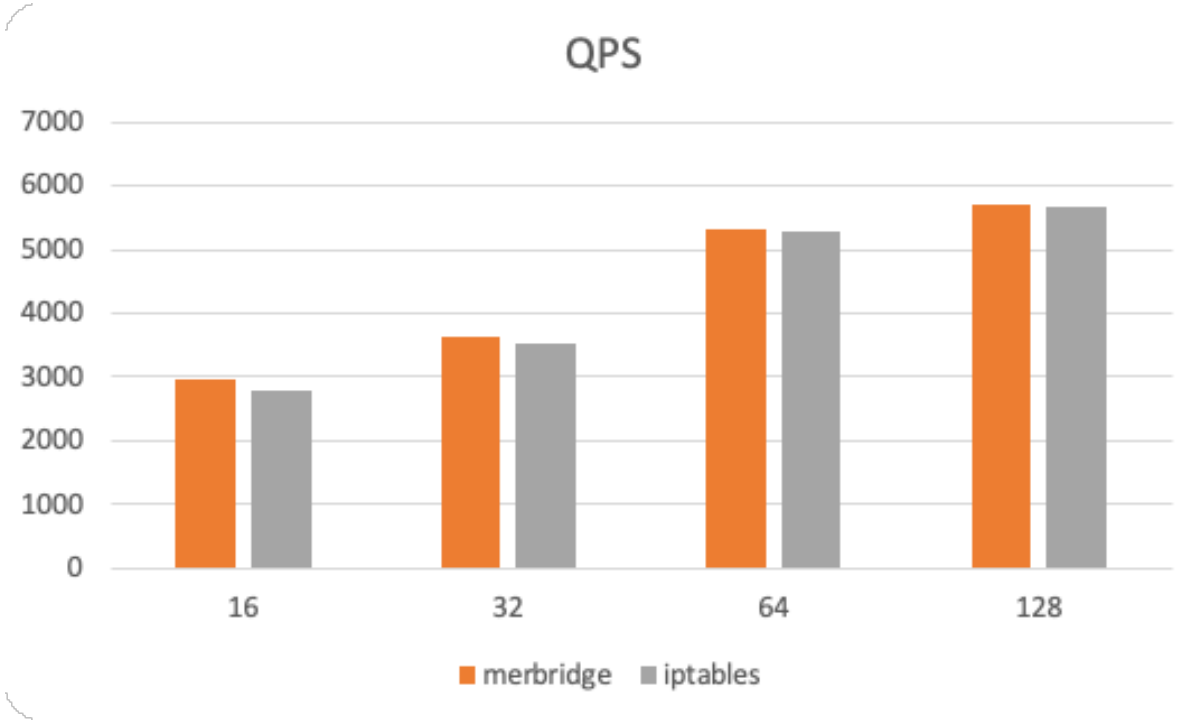
- ORIGINAL_SRC
  Ztunnel will bind the source address of its traffic sent to other Pods to the IP address of the requesting Pod (rather than ztunnel itself). However, Merbridge does not rely on iptables, so it will be modified when ztunnel binds the source address.

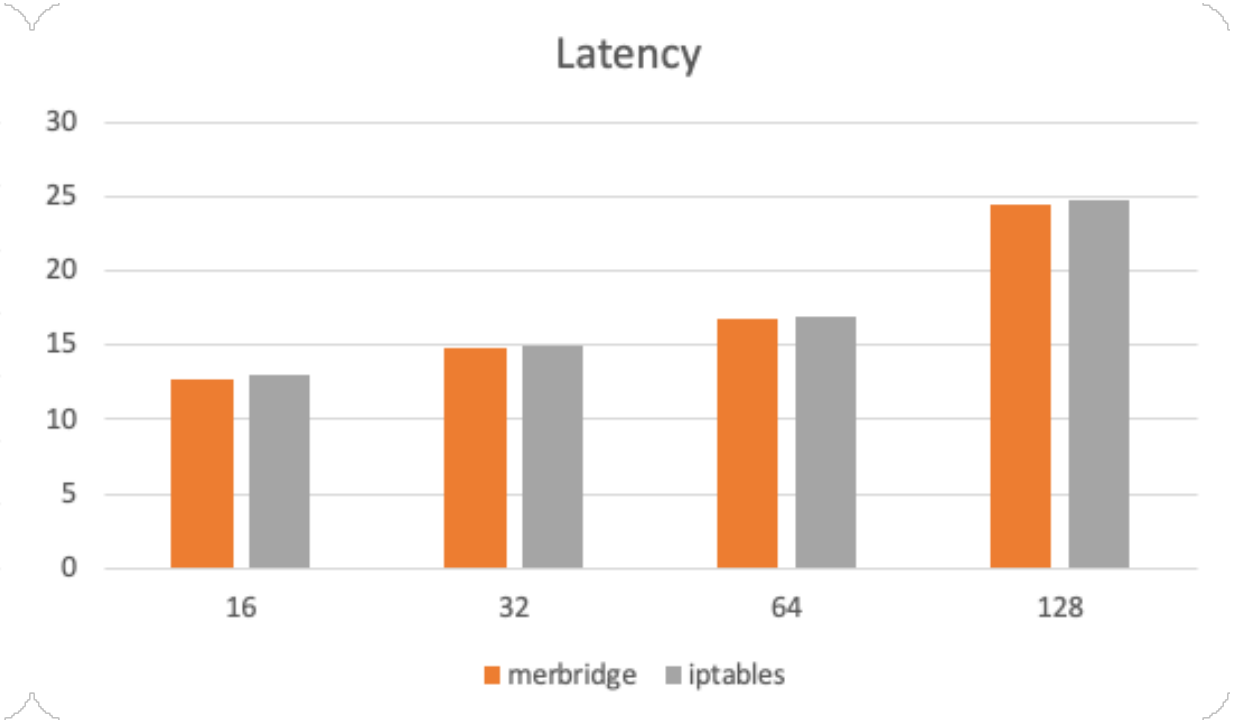
```
if ((cg_info.detected_flags & ZTUNNEL_FLAG) &&
    (cg_info.flags & ZTUNNEL_FLAG)) {
    // ztunnel
    __u32 *ztunnel_ip = get_ztunnel_ip();
    if (!ztunnel_ip) {
        debugf("can not get ztunnel pod ip in bind");
        return 1;
    }
    // ztunnel will bind the source pod ip to upstream,
    // we will rollback this operation because we not support TPROXY mode.
    ctx->user_ip4 = ztunnel_ip[3];
    debugf("successfully rewrite ztunnel bind");
}
return 1;
```

- msg_redirect
  Because ztunnel uses zero copy between downstream and upstream to accelerate the transmission of layer 4 data, in this situation, bpf_msg_redirect cannot be used directly. This will result in the acceleration of bpf_msg_redirect being unavailable in some paths.

Performance

IstioCon





Community

IstioCon

Github：https://github.com/merbridge/merbridge

Slack: https://join.slack.com/t/merbridge/shared_invite/zt-11uc3z0w7-DMyv42eQ6s5YUxO5mZ5hwQ