ServiceMeshCon

November 17, 2020 - Virtual

L7mp: A Multiprotocol Service Mesh for Legacy Applications

ServiceMeshCon

Gábor Rétvári <gabor.retvari@ericsson.com>

Felicián Németh <nemethf@tmit.bme.hu>

MŰEGYETEM 1782

ERICSSON

whoami(1)

ServiceMeshCon

Gábor Rétvári, Sr. Research Fellow @ BME & Ericsson Research

- PhD in Electrical Engineering
- doing research on the intersection of telco & cloud-native

gabor.retvari@gmail.com

@littleredspam

rg0now

Rise of a telco service mesh

ServiceMeshCon

Legacy apps are typically SW monoliths

Service mesh is key to disaggregation: connectivity, routing control, security, resiliency, and observability in cluster networking

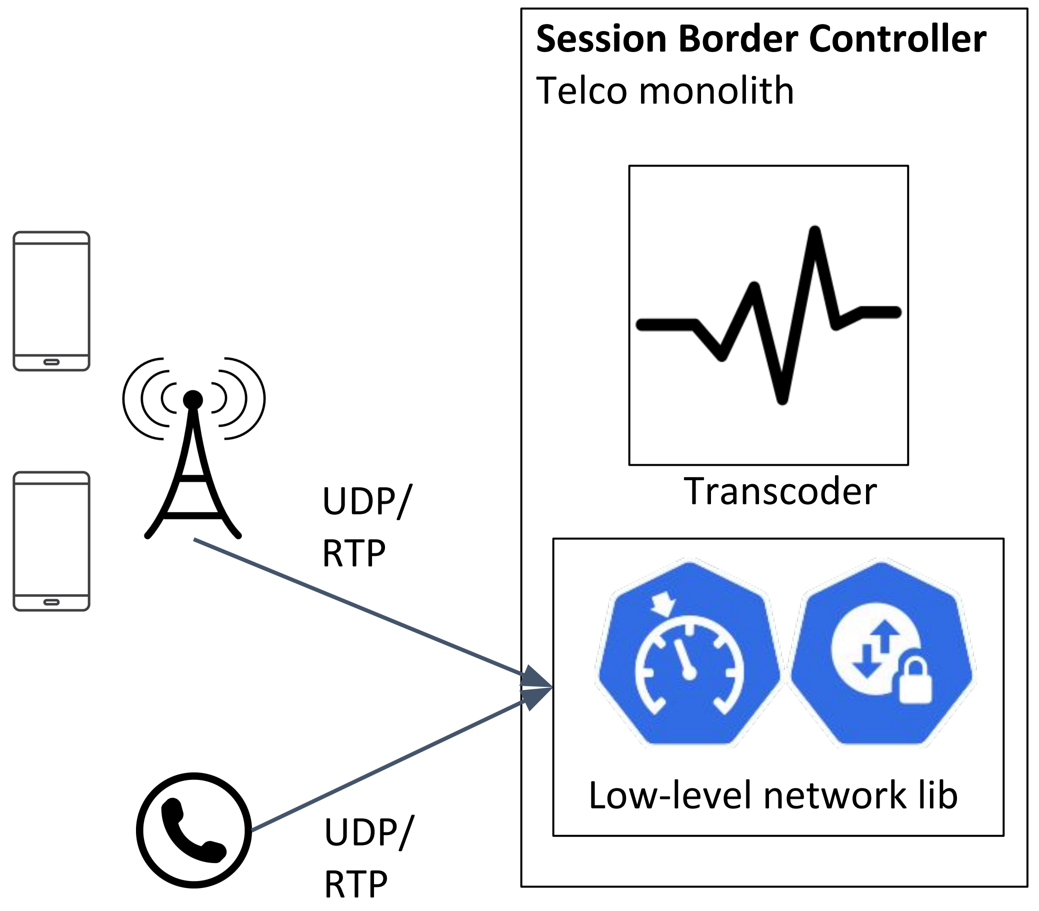
Current **service mesh offerings target the Web crowd** (HTTP-only)

In order to support legacy apps, industry has to **work-around the cloud-native infra**: Multus, NSM, Intel EPA

L7mp is a toy service mesh to experiment with radically new designs to **run legacy apps right on top of an unmodified cloud-native stack**

From a monolithic telco app...

ServiceMeshCon

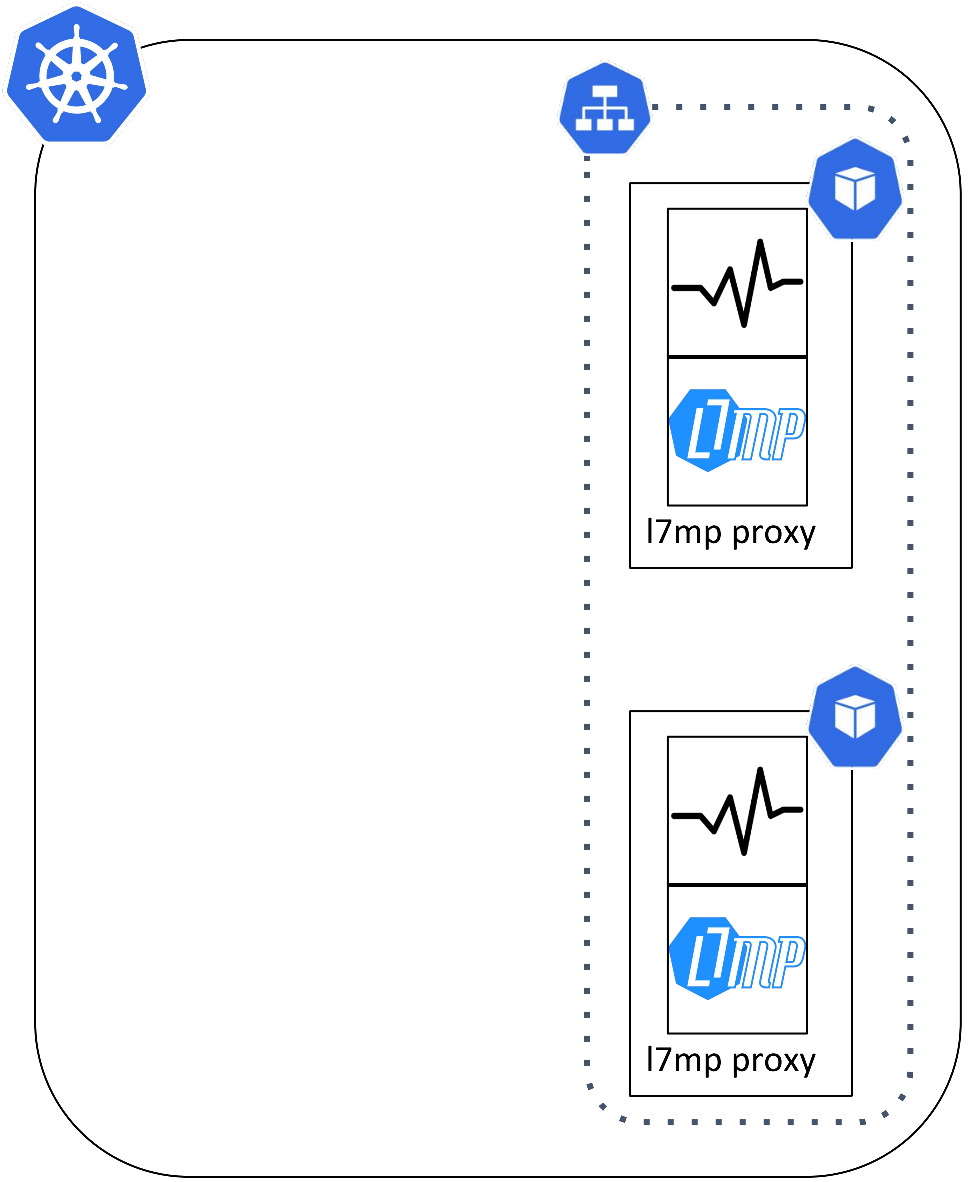


Session Border Controller
Mediate VoIP sessions between different types of user equipments
RealTime Protocol (RTP) over UDP

Firewall
Rate limiting
Media transcoding: **compute intensive!**

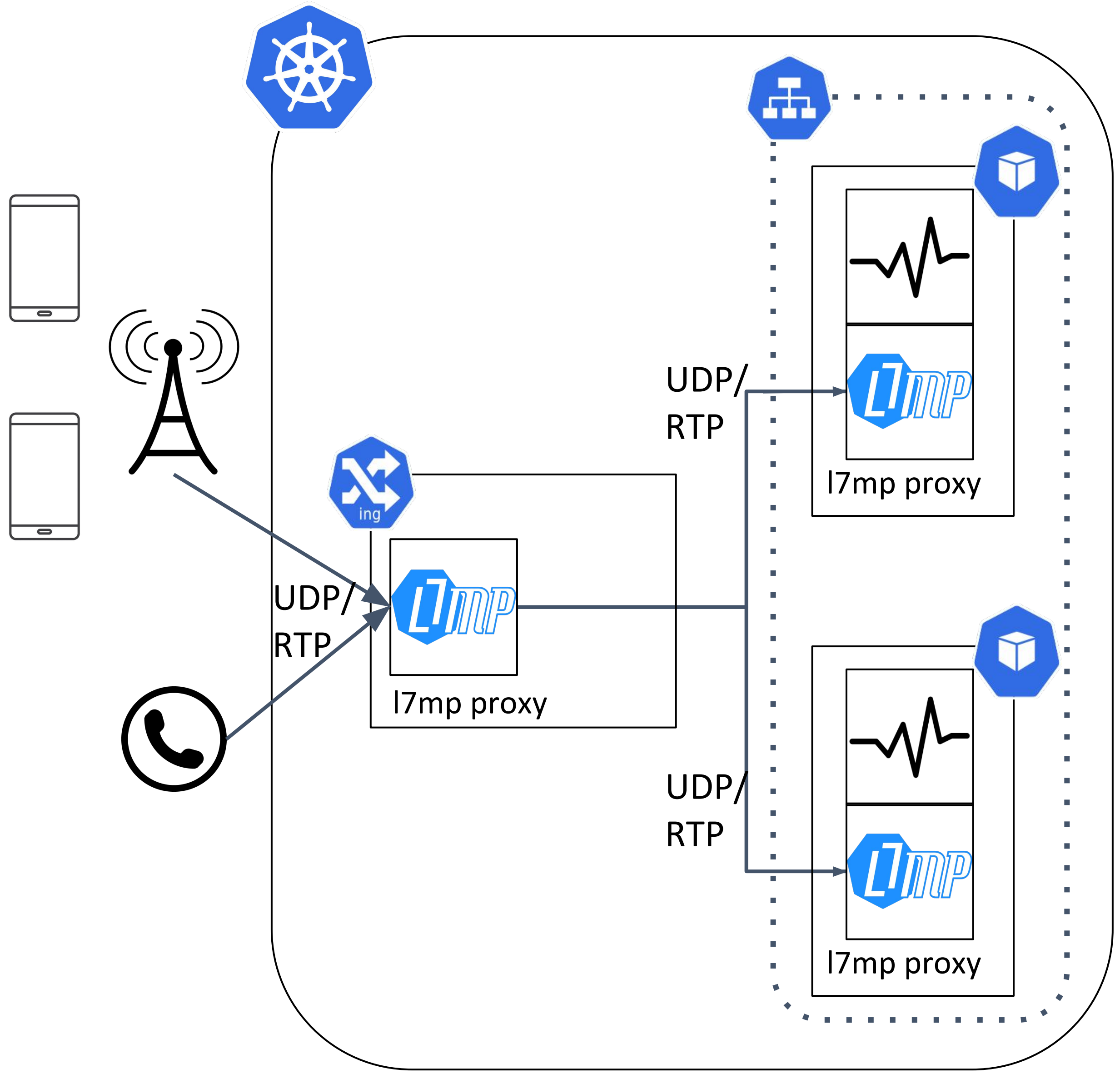
Scaling? Resiliency? Monitoring?

To a cloud-native SBC



Deploy the **transcoder as a microservice** behind a **sidecar proxy** on top of stock Kubernetes

To a cloud-native SBC

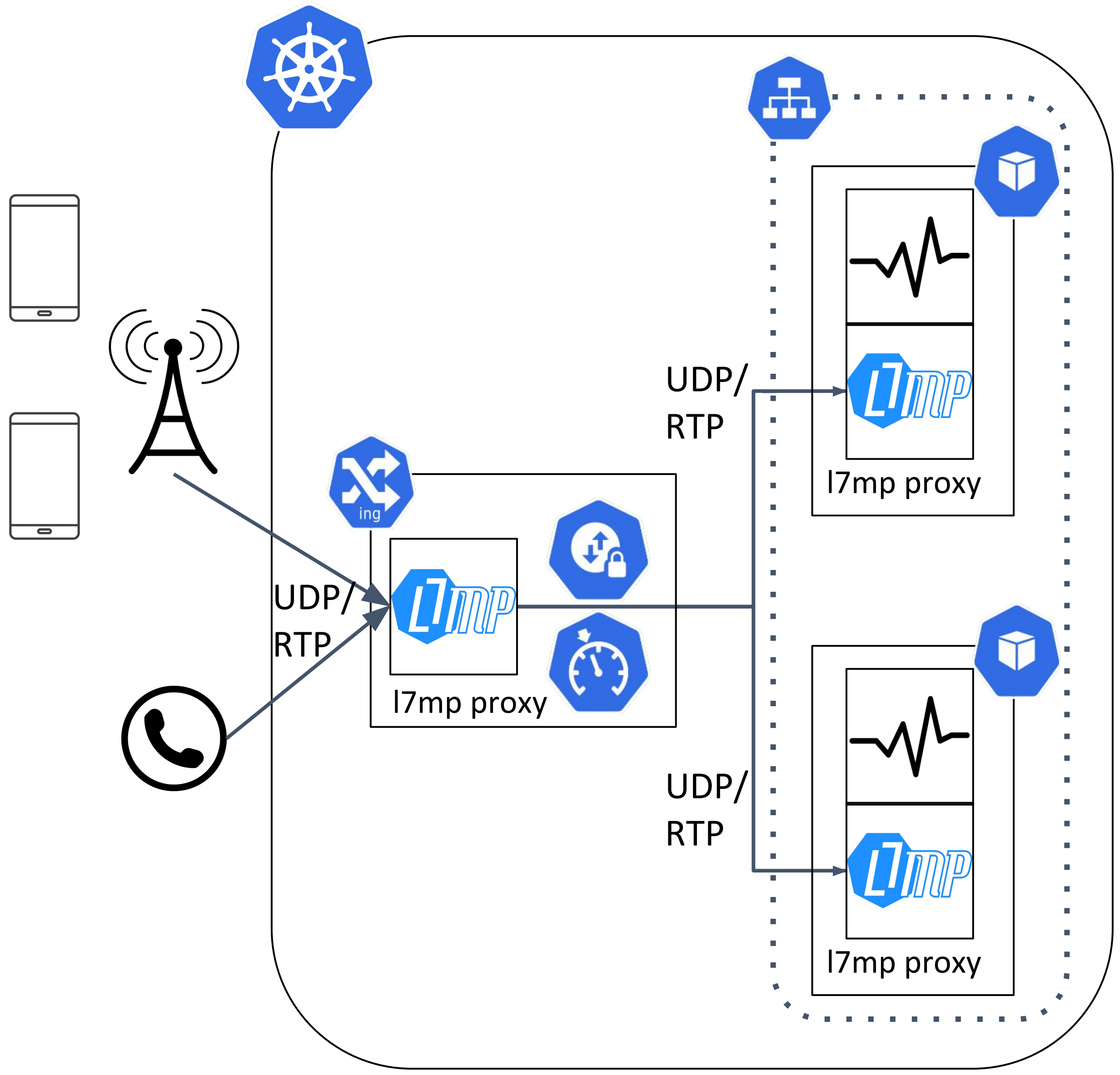


Deploy the **transcoder as a microservice** behind a **sidecar proxy** on top of stock Kubernetes

Let the ingress gateway **load-balance UDP/RTP streams** across the transcoder pods

To a cloud-native SBC

ServiceMeshCon

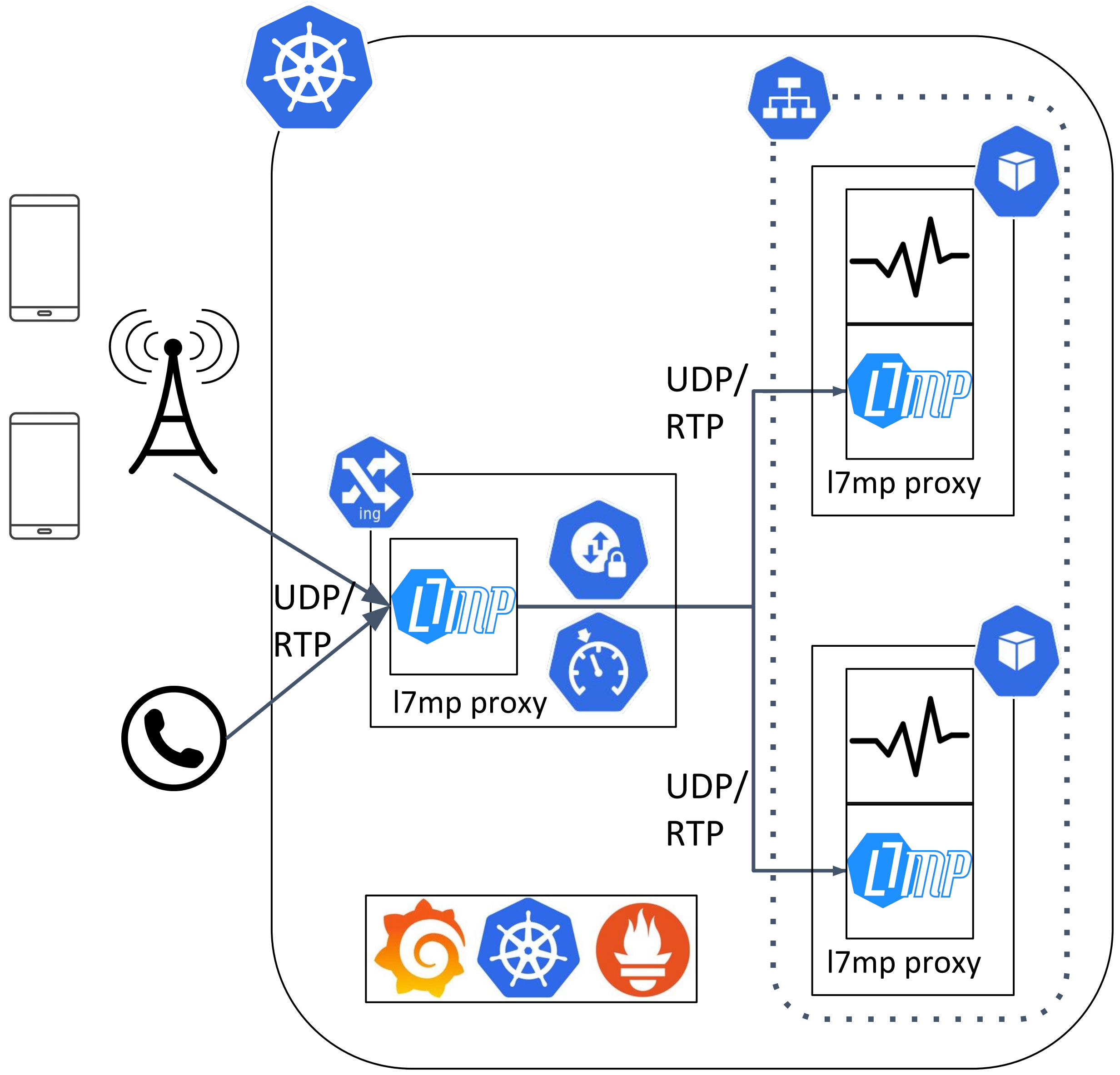


Deploy the **transcoder as a microservice** behind a **sidecar proxy** on top of stock Kubernetes

Let the ingress gateway **load-balance UDP/RTP streams** across the transcoder pods

Filter & rate-limit at the ingress

To a cloud-native SBC



Deploy the **transcoder as a microservice** behind a **sidecar proxy** on top of stock Kubernetes

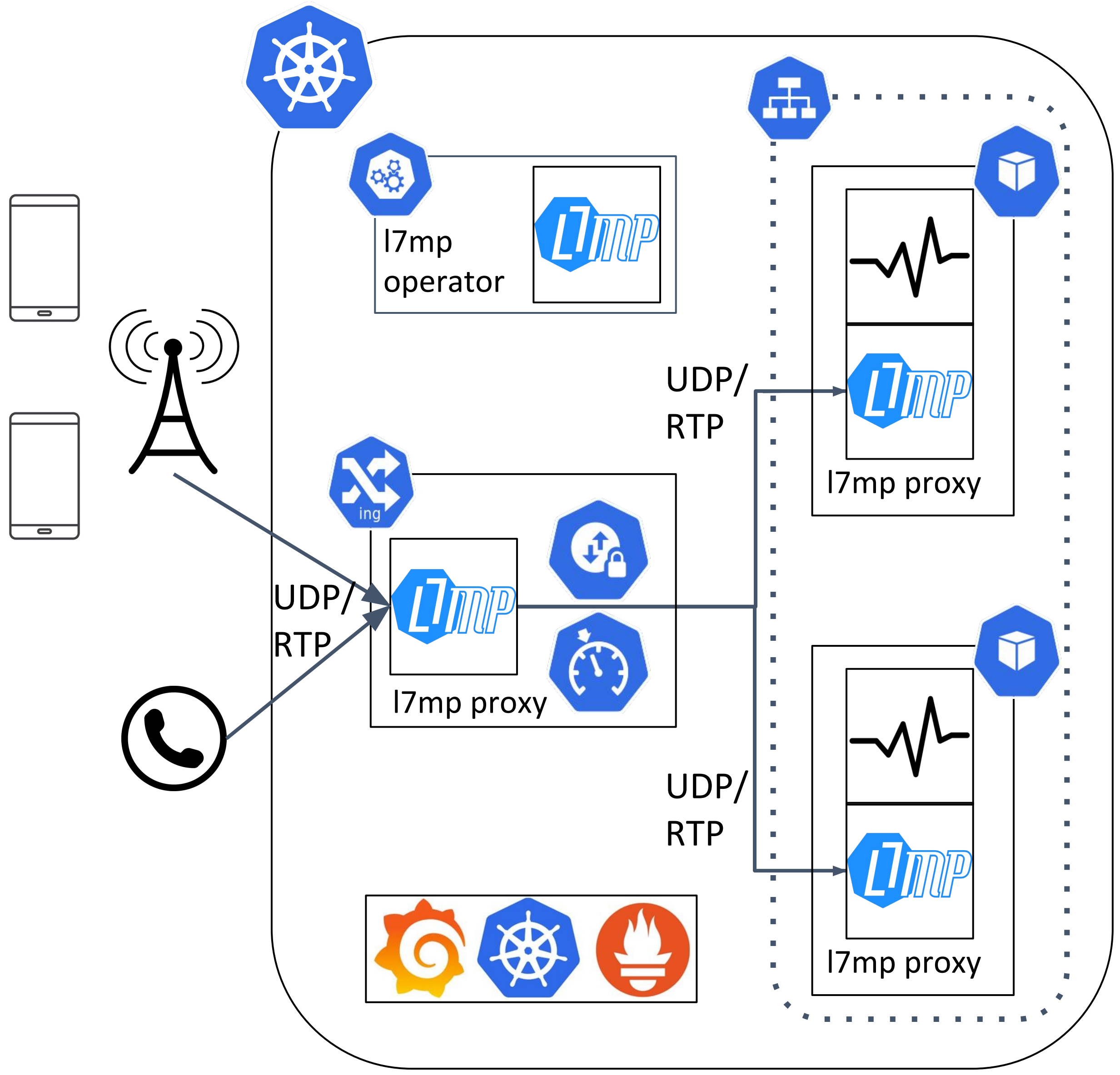
Let the ingress gateway **load-balance UDP/RTP streams** across the transcoder pods

Filter & rate-limit at the ingress

Monitor gateways & proxies

To a cloud-native SBC

ServiceMeshCon



Deploy the **transcoder as a microservice** behind a **sidecar proxy** on top of stock Kubernetes

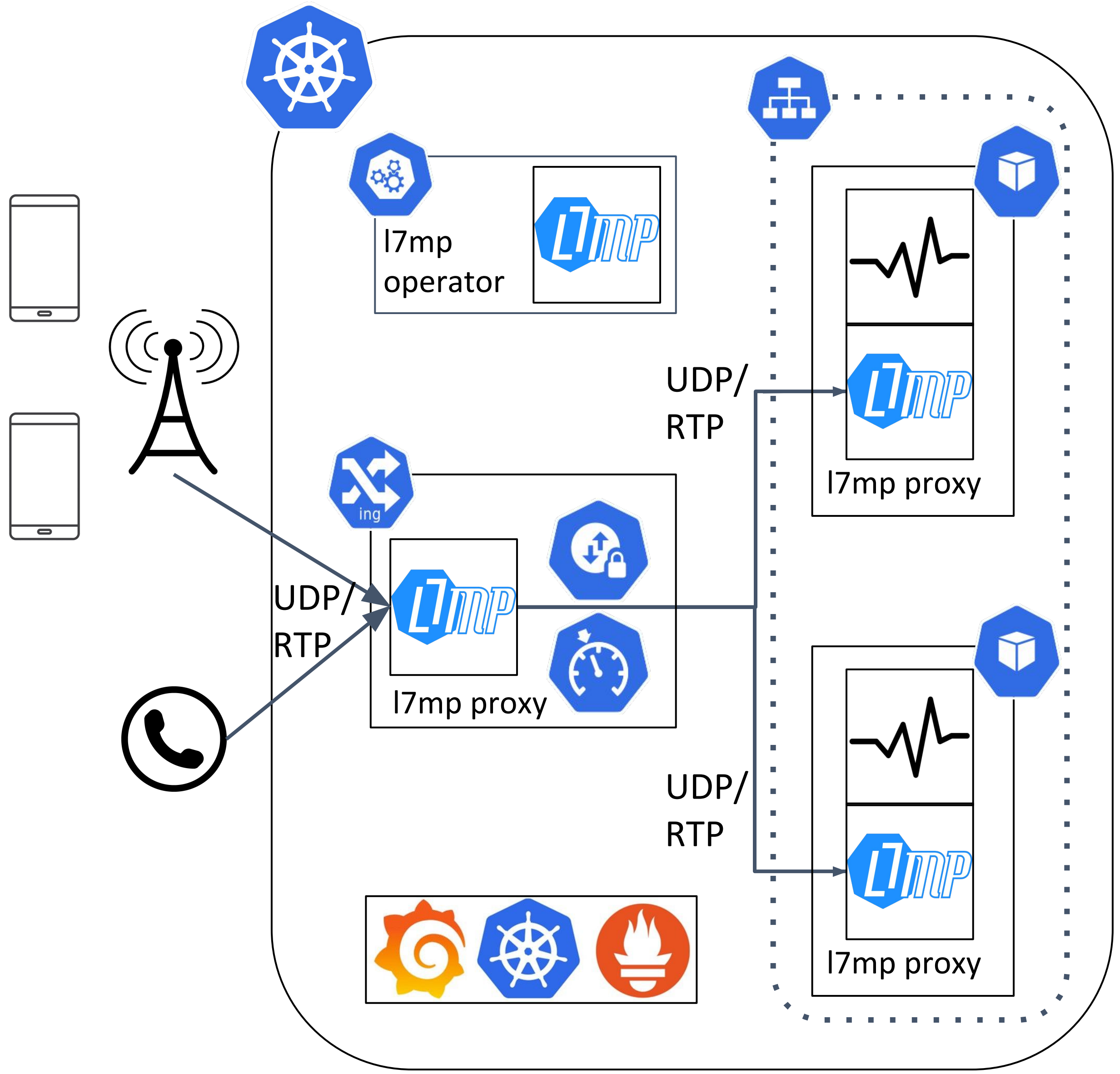
Let the ingress gateway **load-balance UDP/RTP streams** across the transcoder pods

Filter & rate-limit at the ingress

Monitor gateways & proxies

Deploy a **K8s controller** on top **to manage the gateways and sidecar proxies**

To a cloud-native SBC



Deploy the **transcoder as a microservice** behind a **sidecar proxy** on top of stock Kubernetes

Let the ingress gateway **load-balance UDP/RTP streams** across the transcoder pods

Filter & rate-limit at the ingress

Monitor gateways & proxies

Deploy a **K8s controller** on top **to manage the gateways and sidecar proxies**

This is the **service mesh pattern!**

Telco requirements are special

ServiceMeshCon

	Telco	**Cloud-native** (e.g., Istio/Envoy)
Network protocols	**Legacy**: UDP/SCTP, RADIUS, VPN, VxLAN/Geneve/GTP, SNMP, DNS, TFTP, LWM2M/CoAP	**HTTP/WebSocket/gRPC,** QUIC?
Traffic profile	long-lived **media streams**	short-lived **request-response**
KPIs	**Per-packet latency** (usecs) and **throughput** (million packet per sec)	**Per-HTTP-session latency** (10s of msecs) and **throughout** (few 10ks of HTTP request/sec)
Service mesh features (on top of routing, security, observability)	multiplexing/demultiplexing, encapsulation/decapsulation, etc.	**HTTP !**

L7mp: An experimental SM

ServiceMeshCon

http://l7mp.io

https://github.com/l7mp/l7mp

https://www.npmjs.com/package/@l7mp/l7mp

https://hub.docker.com/r/l7mp/l7mp

https://l7mp.slack.com

Sponsored by ERICSSON

L7mp is a **service mesh we built to experiment with new ideas** in order to support legacy apps over K8s

Multiprotocol: HTTP, WebSocket, TCP + UDP, DNS, RTP/RTCP, UDS + SCTP, RADIUS, SNMP easy to add

Extensible: control plane ~1k LOC Python + proxy is ~10k LOC node.js

Playground for new ideas: kernel proxy offload, service chaining

Upstream to Envoy what ends up useful

Example: SBC Transcoder

ServiceMeshCon

```
apiVersion: l7mp.io/v1
kind: VirtualService
metadata:
  name: worker-vsvc
  namespace: default
  selector:
    serviceName: worker-svc
spec:
  RTP:
    transport:
      UDP: { port: 19000 }
  rules:
    - action:
        route:
          destination:
            RTP
              transport:
                UDP:
                  port: 20020
                  bind:
                    port: 3986
            endpoints:
              - spec:
                  address: "127.0.0.1"
```

Services abstracted with the familiarly named **VirtualService** Kubernetes Custom Resource Definition

VirtualService is always backed by a Kubernetes service

Expose transpocoder service on **RTP over UDP**

Route calls to the transcoder service on localhost and bind local port

Example: SBC Gateway

ServiceMeshCon

```
apiVersion: l7mp.io/v1
kind: VirtualService
metadata:
  name: ingress-gateway-vsvc
  namespace: default
  selector:
    serviceName: ingress-gateway-svc
spec:
  RTP: { transport: { UDP: { port: 18002 } } }
  rules:
    - match:
        op: starts
        path: '/IP/src_addr'
        value: "192.168.0"
      action:
        route:
          destination:
            serviceName: transcoder-vsvc
            loadbalancer:
              policy: ConsistentHash
              key: "/RTP/SSRC"
        retry:
          retry_on: always
          num_retries: 3
          timeout: 2000
```

Expose SBC service on the gateway on RTP/UDP at a given port

Accept calls **only from a specific IP subnet**

Route calls to the transcoder service and **load-balance with a custom sticky session** rule

Timeout streams and **retry** each call at most 3 times

L7mp: Try it and let's talk!

ServiceMeshCon

http://l7mp.io

https://github.com/l7mp/l7mp

https://www.npmjs.com/package/@l7mp/l7mp

https://hub.docker.com/r/l7mp/l7mp

https://l7mp.slack.com

Sponsored by ERICSSON

L7mp is under active development, not everything works as expected

Can already **host a fully functional SBC**, providing traffic management and resiliency for plain UDP calls

But it is **more generic than telco**!

If you're from **telco or video-gaming,** or trying to deploy a legacy app on top of K8s, **come talk to us!**

If you're a **cloud-native vendor** and want to go after legacy use cases, **come talk to us!**