gRPConf INDIA 2026

Your First gRPC Service on Kubernetes

Package it · Health-check it · Route it · Load-balance it

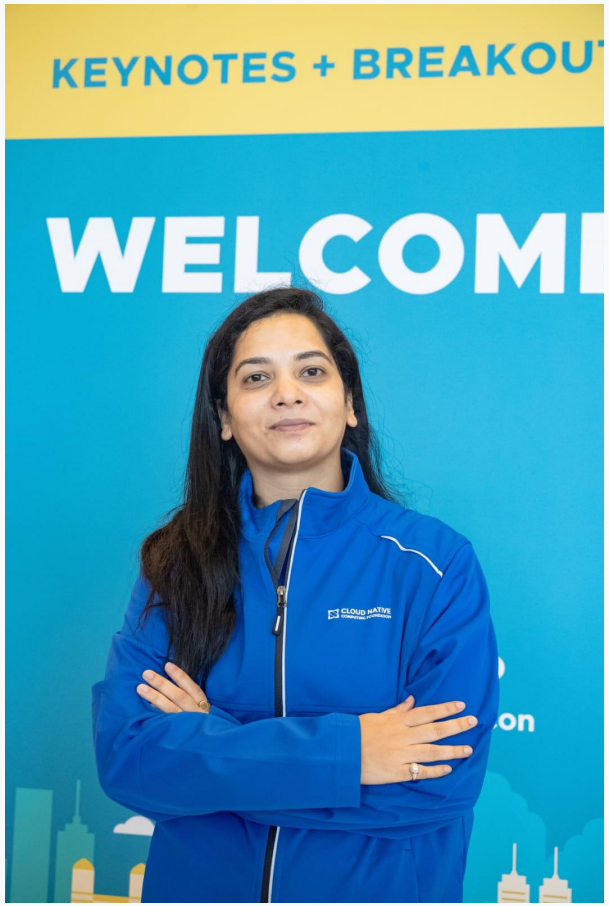
Oshi Gupta

Site Reliability Engineer, InfraCloud Technologies



$ whoami

gRPConf INDIA 2026



Oshi Gupta

Looks easy on paper

gRPConf INDIA 2026

Friday, 5 PM. **Greeter** works on your laptop. Your lead says: “Just put it on Kubernetes, same as any HTTP app.” Three surprises are waiting.

01

gRPC needs HTTP/2

Any hop that only speaks HTTP/1.1 (an Ingress, a probe) breaks it.

02

Health checks are different

The usual httpGet probe can’t talk to a gRPC server, so pods restart forever.

03

Connections live long

One HTTP/2 connection carries every RPC, so one pod can end up doing all the work.

Our story: Greeter Darshini, Koramangala

Every Bengalurean knows the **darshini**: walk in, pay, get a token, eat standing, go.

Short menu, no frills, hundreds of plates an hour.

Greeter is our darshini. Today it opens in Koramangala, and it has to survive the **1 PM delivery rush**.



The plan: four acts

gRPConf INDIA 2026

Greeter is our new dosa darshini in Koramangala. Each act: the beginner mistake, then the fix.

1

Package

Run a lean darshini

A small, static, non-root image.

2

Health-check

Open the shutter right

Native gRPC probes, readiness vs liveness.

3

Route

Be easy to find

Gateway API GRPCRoute.

4

Load-balance

Survive the 1 PM rush

Spread RPCs across every pod.

CHAPTER 1 · PACKAGE IT

Token No. 47.
Your idli is ready.

A Bengaluru darshini runs lean: a short menu, no tables, no waiters, and everyone is a number. Hundreds of plates an hour.

> **In Kubernetes:** a static binary in a distroless image, running as numeric user 65532.



gRPConf INDIA 2026

Act 1 · Package it

gRPConf INDIA 2026

SMALL, STATIC, NON-ROOT

Dockerfile · run stage

```
FROM gcr.io/distroless/static-debian12:nonroot
COPY --from=build /out/server /

# numeric UID, not a user name
USER 65532:65532
EXPOSE 50051
ENTRYPOINT ["/server"]
```

- **Commit the generated code.** The build only needs Go: one less tool in CI.
- **Static binary + distroless.** A few MB, no shell, no package manager.
- **Use a numeric USER.** With a named user, runAsNonRoot: true fails: “image has non-numeric user”.
- **No shell, no grpc_health_probe.** That’s why Act 2 uses the kubelet’s native probe.

CHAPTER 2 · HEALTH CHECKS

The gate isn't broken. It can't read paper.

Hold a BMTC paper ticket up to a Namma Metro gate and the flaps stay shut, however valid the ticket.

> **In Kubernetes:** an httpGet probe speaks HTTP/1.1; a gRPC server speaks only HTTP/2.



Act 2 · The copy-paste probe

gRPConf INDIA 2026

THE MISTAKE

```
01-deployment-WRONG-probes.yaml

livenessProbe:
  httpGet:  # HTTP/1.1
    path: /healthz
    port: 50051
  periodSeconds: 3
  failureThreshold: 2
```

READY 0/1 · RESTARTS keeps climbing

```
Liveness probe failed: … net/http:
HTTP/1.x transport connection broken:
malformed HTTP response
"\x00\x00\x06\x04…"
```

httpGet speaks HTTP/1.1; gRPC speaks only HTTP/2. Those bytes are an HTTP/2 SETTINGS frame. The server is fine, but the probe can't talk to it.

Same root cause as the classic Ingress breakage: **any HTTP/1.1-only hop breaks gRPC.**

Act 2 · Native gRPC probes

gRPConf INDIA 2026

THE FIX · GA SINCE KUBERNETES 1.27

```
02-deployment.yaml
readinessProbe:
  grpc:
    port: 50051  # a number!
  periodSeconds: 2
livenessProbe:
  grpc:
    port: 50051
    service: liveness
lifecycle:
  preStop:
    sleep: { seconds: 5 }
```

Probe	Asks	Warm-up
Readiness	service "": can I take traffic?	NOT_SERVING
Liveness	service "liveness": am I alive?	SERVING



IN BENGALURU

6:30 AM at the darshini: staff are in (**alive**), but the tawa is still heating, so the shutter stays half-down (not **ready**).

- **Port must be a number:** named ports aren't supported for gRPC probes.
- **On SIGTERM:** flip health to NOT_SERVING, then GracefulStop().

CHAPTER 3 · ROUTING

“12th Main, 3rd Cross…” Which one?

Every Bengaluru layout has a 12th Main. So we ask for the place instead: “Vidyarthi Bhavan, Gandhi Bazaar.”

In Kubernetes: GRPCRoute matches what you want, the gRPC service and method, not a URL path.



Act 3 · Route it with Gateway API

gRPConf INDIA 2026

WHAT MOST TUTORIALS SHOW

Ingress NGINX + annotations

backend-protocol: "GRPC" on an Ingress. Vendor-specific, and Ingress NGINX was retired in March 2026: no more releases or security fixes.

THE MODERN PATH

Gateway API · GRPCRoute

A first-class route type that matches on gRPC service and method. Portable across Gateway implementations.

WHO OWNS WHAT

GatewayClass
Platform team

→

Gateway
Cluster operator

→

GRPCRoute
App team

Act 3 · A GRPCRoute

gRPConf INDIA 2026

05-grpcroute.yaml

```yaml
kind: GRPCRoute
spec:
  parentRefs:
    - name: grpc-gw
  hostnames:
    - greeter.demo.local
  rules:
    - matches:
        - method:
            service: greeter.v1.Greeter
      backendRefs:
        - name: greeter
          port: 50051
```

- **Match on method.service** (and optionally method.method): gRPC concepts, not URL paths or annotations.
- **Set hostnames.** Wrong host, no route: the call gets a 404.
- **On the Service port:** appProtocol: kubernetes.io/h2c, the portable way to say “talk HTTP/2 to me”.
- **Reflection** is routed so grpcurl list works. Many teams don’t expose it in prod.

CHAPTER 4 · THE 1 PM RUSH

3 replicas. 300 requests.

1 pod did all the work.

1 PM at a Koramangala cloud kitchen: the guard sends a rider to counter 1, and he places all 300 orders there.

> **In Kubernetes:** kube-proxy balances connections, not requests, and gRPC sends every RPC over one connection.

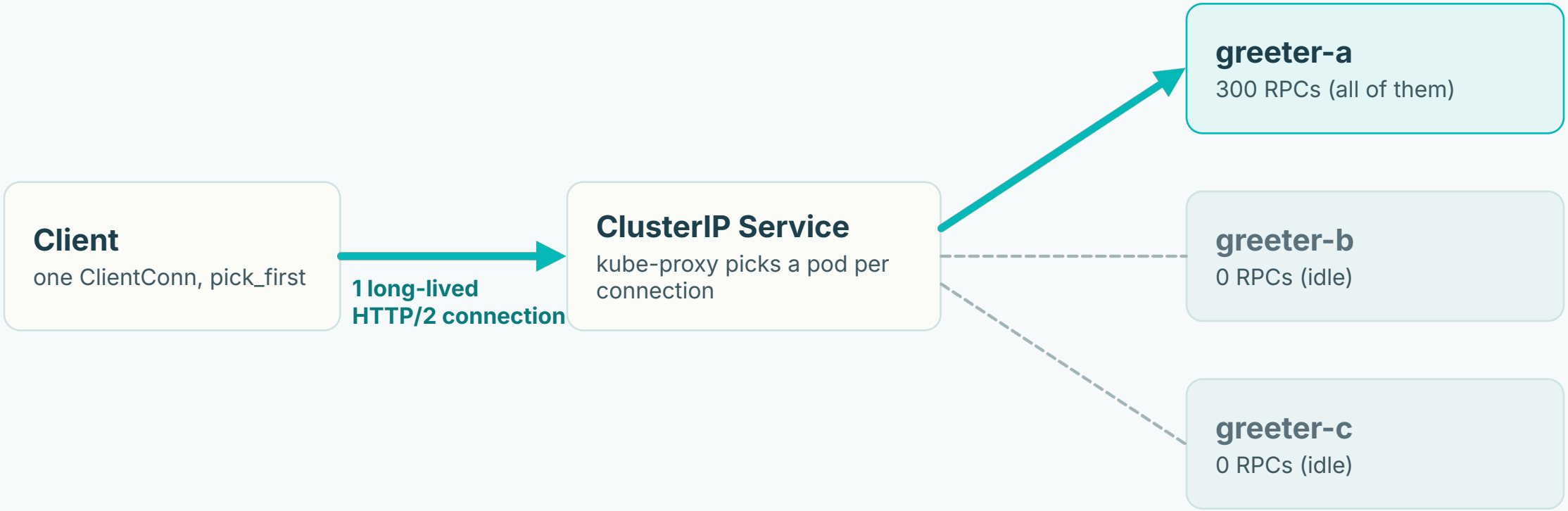


gRPConf INDIA 2026

Act 4 · One connection, every RPC

gRPConf INDIA 2026

At a Koramangala cloud kitchen, the gate guard (kube-proxy) assigns a counter when the rider walks in. He never walks back out, so counters 2 and 3 stand idle.



Act 4 · Two ways to spread the RPCs

gRPConf INDIA 2026

FIX 1 · CLIENT-SIDE

Headless Service + dns:/// + round_robin

```
target: dns:///greeter-headless
  .grpc-demo.svc.cluster.local:50051
lb policy: round_robin
```

The rider knows every counter and rotates his orders. DNS returns every pod IP; one connection per pod. **Drop any one of the three and you're back to one pod.**

Result: 100 / 99 / 101

FIX 2 · PROXY-SIDE

Let an L7 proxy balance per request

A floor manager takes each order and hands it to the next counter. Envoy balances each request, not each connection.

Zero client changes. The same idea is behind a service mesh.

Result: even spread per request

The sequel: new pods get no traffic

Scale 3 → 4: the new pod turns Ready and gets **0** requests. The client resolved DNS once and never looks again.

cmd/server/main.go

```
grpc.KeepaliveParams(
  keepalive.ServerParameters{
    MaxConnectionAge:      age,
    MaxConnectionAgeGrace: 10 * time.Second,
  })
```



IN BENGALURU

A 4th counter opens; the rider never looks up. Every 15s the counter says “please re-queue” (**GOAWAY**), so he walks back and finds it.

- **Server sends GOAWAY** once a connection is older than MaxConnectionAge (15s in the demo).
- **Client reconnects and re-resolves DNS,** and the new pod joins the rotation.
- **Expect up to ~30s.** grpc-go re-resolves DNS at most every 30 seconds.

Demo

gRPConf INDIA 2026

Video



THREE LANES, SAME PODS

- **ClusterIP + pick_first:** one pod does all the work.
- **Headless + round_robin:** spread across pods.
- **Gateway API GRPCRoute:** per request, via Envoy.

Run it yourself:
github.com/oshi36/first-grpc-service-k8s

Bengaluru, decoded

gRPConf INDIA 2026

In Bengaluru	In Kubernetes
A darshini token: called by number	Distroless image, numeric non-root USER
Namma Metro gate that can’t read a paper ticket	httpGet probe against an HTTP/2-only server
Staff in, tawa heating, shutter half-down	Readiness ≠ liveness
Ask for the place, not “12th Main, 3rd Cross”	GRPCRoute matching service and method
Rider stuck at counter 1 in a Koramangala kitchen	One HTTP/2 connection, one pod, every RPC
Rider rotates across counters	Headless Service + dns:/// + round_robin
Floor manager hands out each order	Envoy via Gateway API, per-request balancing
“Please re-queue” every 15 seconds	MaxConnectionAge → GOAWAY → re-resolve

Greeter Darshini survived the 1 PM rush. Here’s the checklist to make yours survive too.

Today’s menu: your take-home checklist

gRPConf INDIA 2026

Package

- Static binary, distroless image
- Numeric non-root USER
- Register the grpc.health.v1 service

Health checks

- Native grpc: probes, not httpGet or tcpSocket
- Probe port is a number
- Readiness ≠ liveness
- SIGTERM: NOT_SERVING, then GracefulStop()

Routing

- GRPCRoute, not Ingress annotations
- Match service/method; set hostnames
- appProtocol: kubernetes.io/h2c
- Every hop speaks HTTP/2

Load balancing

- Headless + dns:/// + round_robin, or an L7 proxy
- Set MaxConnectionAge
- Verify per-pod spread

Full recipe: **CHECKLIST.md** in the demo repo.

gRPConf INDIA 2026

Thank you! Questions?

Code Repo

