



gart.

Gotchas for Developers Kubernetes Migration



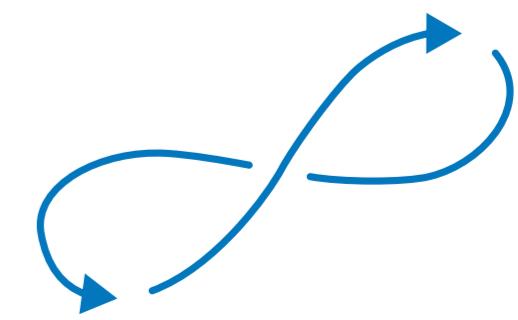
This post is ideal for application
developers who want to migrate
their apps to Kubernetes but lack
a dedicated DevOps team.





8 Gotchas to Avoid

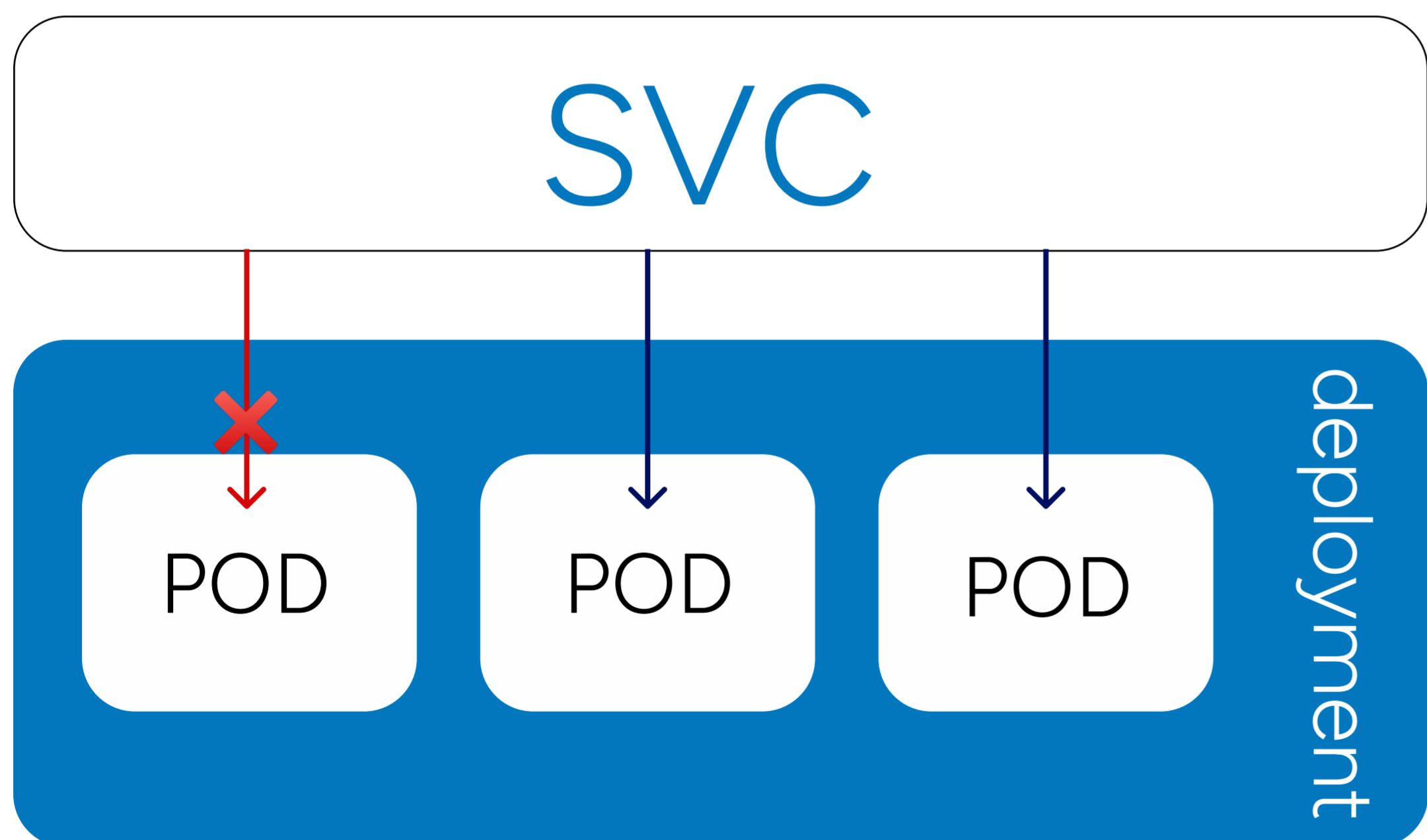
gart.



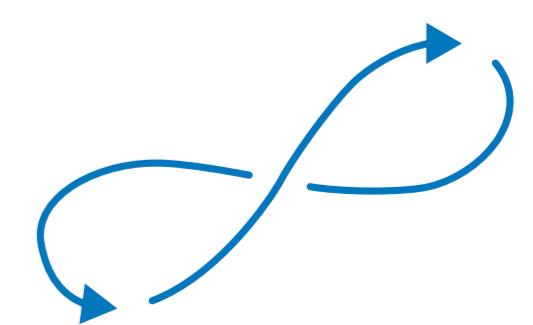


Stateful Applications

Kubernetes is designed for stateless applications. Stateful applications require extra effort for fault tolerance and data management.

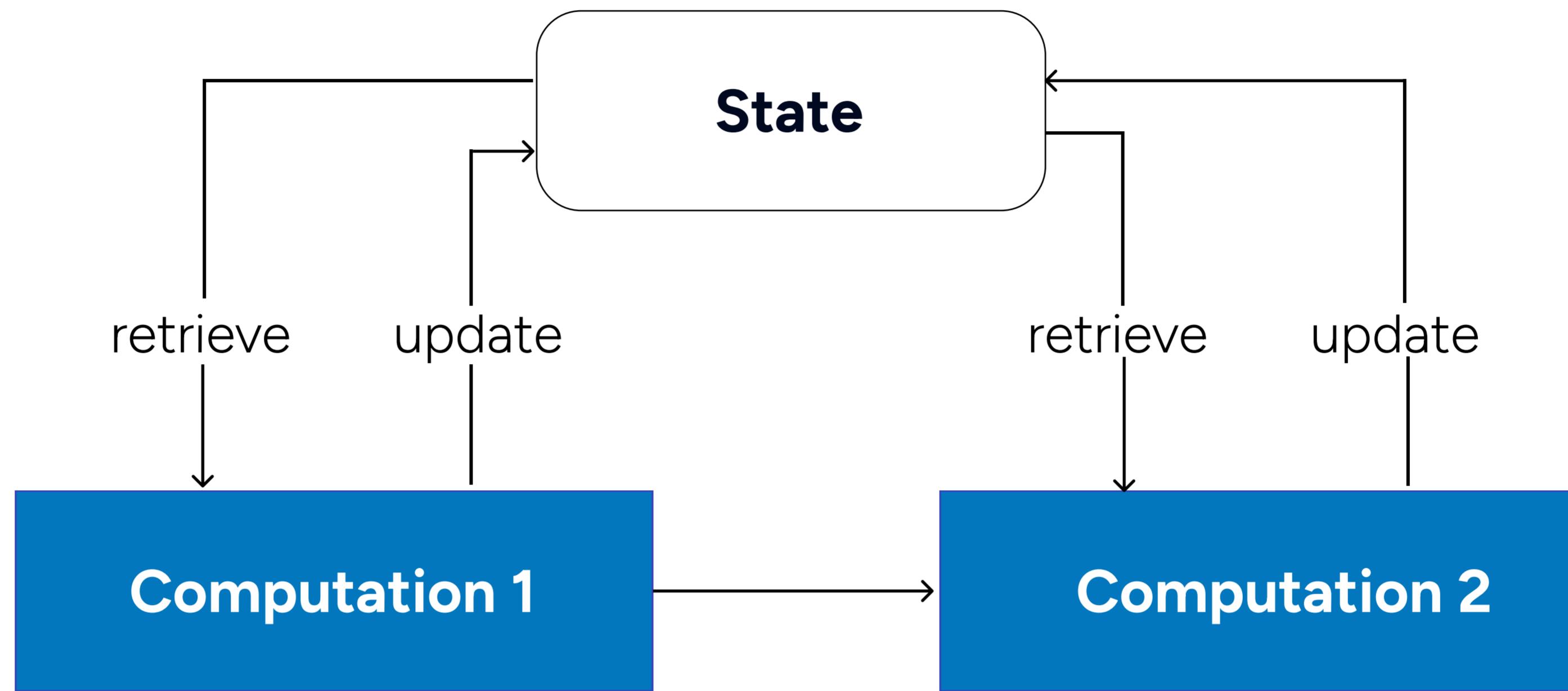


gart.

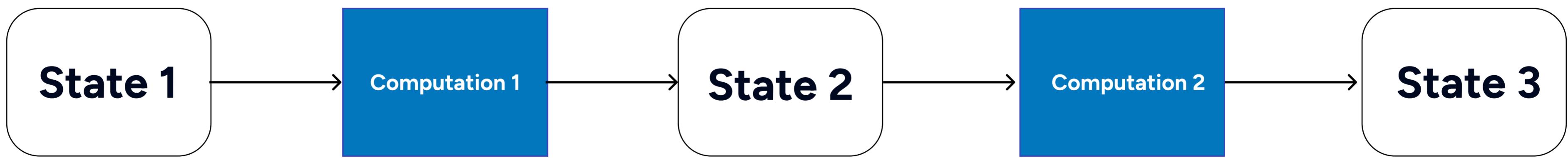




Stateful Application



Stateless Application





Missing Health Checks

Kubernetes relies on health checks to monitor application health. Improper configuration can lead to issues like cascading pod restarts.

Unpredictable Resource Consumption

Kubernetes relies on health checks to monitor application health. Improper configuration can lead to issues like cascading pod restarts.





Configuration Management

Don't hardcode configurations in your code. Leverage Kubernetes features like ConfigMaps and Secrets for easier management.

Ungraceful Shutdown

Ensure your application handles SIGTERM signals gracefully to allow for clean shutdown before termination.

Pod Affinity

Your application shouldn't rely on specific pods. Design your application to function across multiple replicas.





SSL Certificate Management

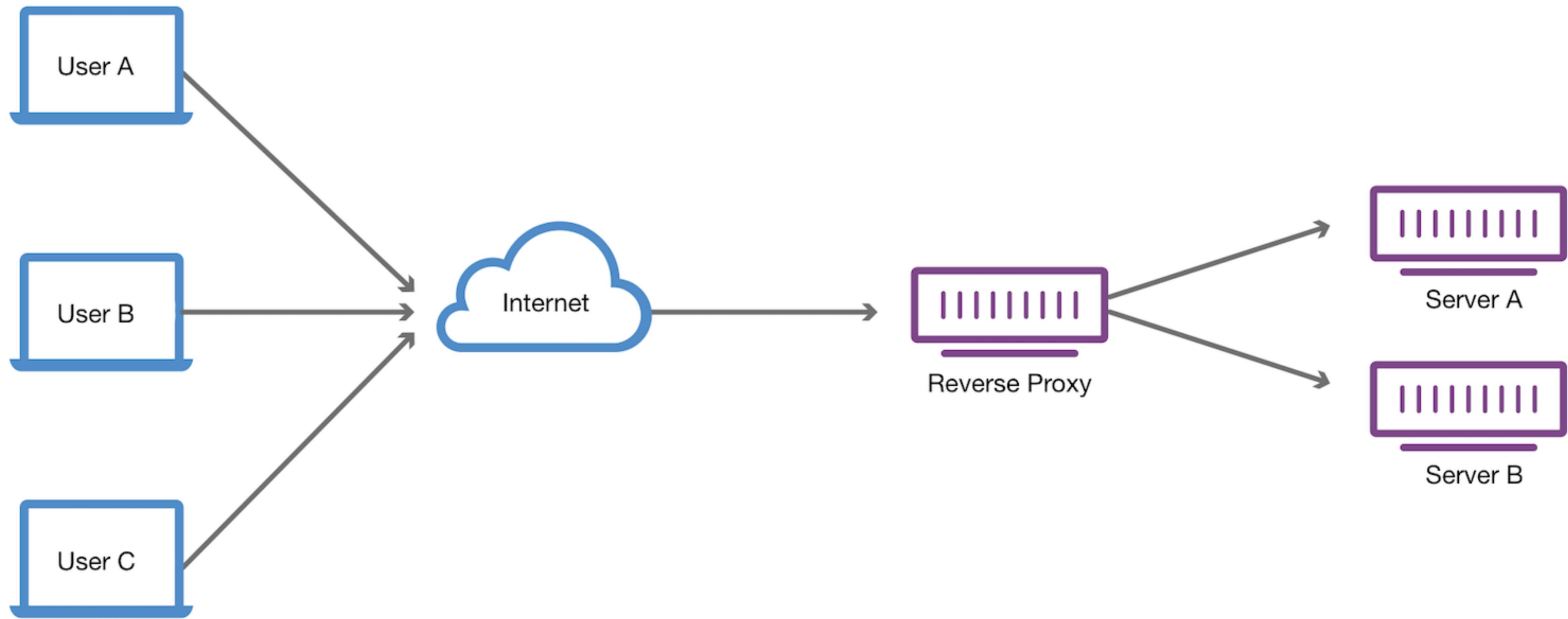
Let Kubernetes handle SSL certificate management with tools like cert-manager.





Reverse Proxy and HTTPS

Use a reverse proxy like Ingress and serve links over HTTPS for added security.



Avoid these gotchas and ensure a smooth Kubernetes migration with a dedicated DevOps team!

Contact Gart Solutions today for our expert
DevOps as a Service (DaaS) and let us handle the
heavy lifting.



Roman Burdiuzha
Cloud Architect | Co-Founder & CTO at Gart