Services

A group of **pods that work together**
- grouped by a selector

Defines access policy
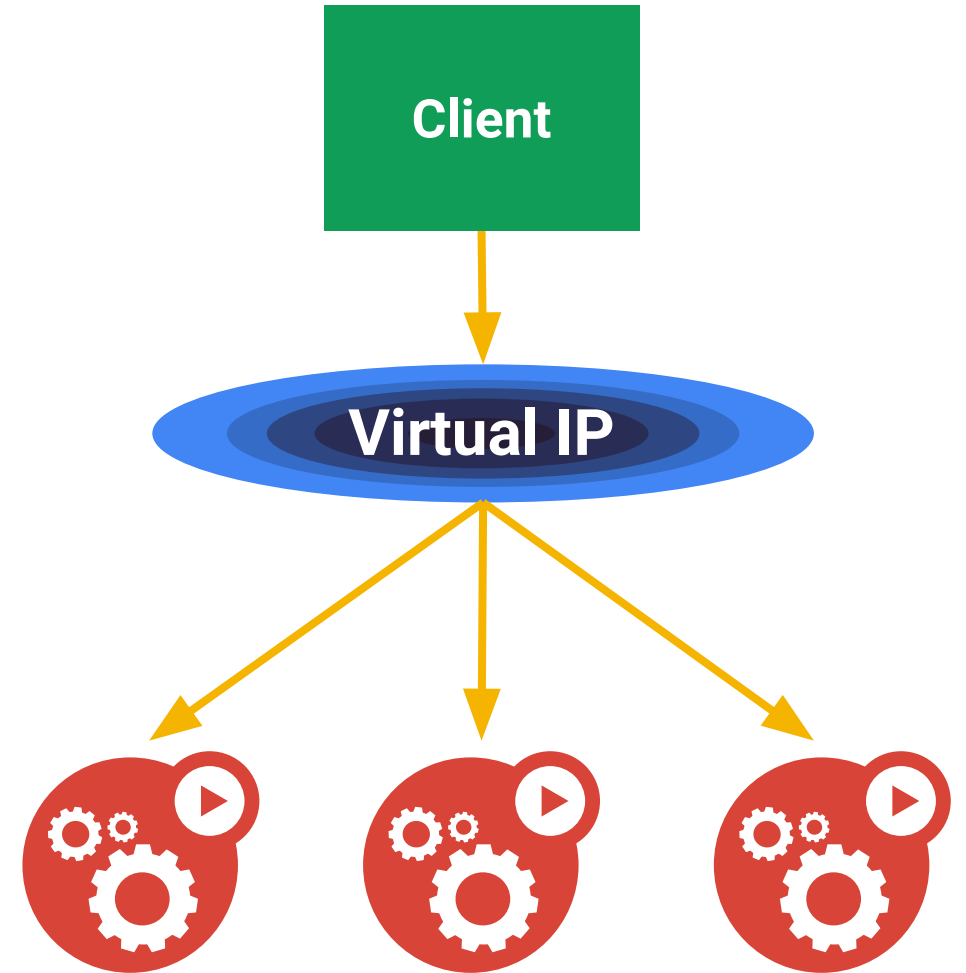- "load balanced" or "headless"

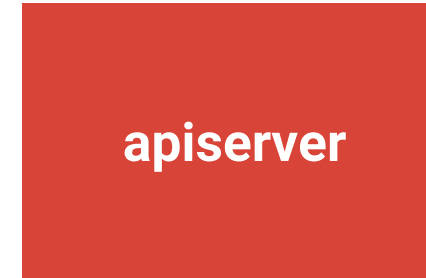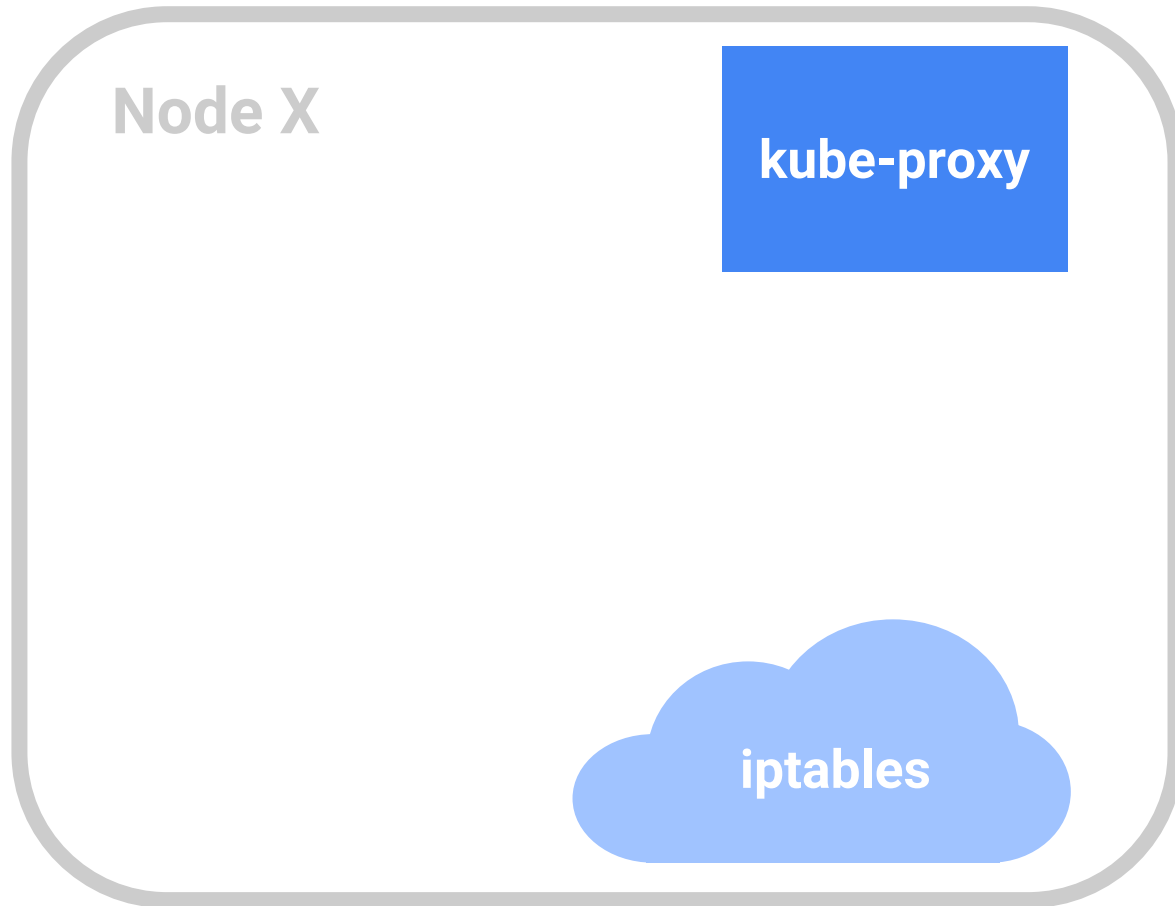Can have a stable **virtual IP** and port
- also a DNS name

VIP is managed by *kube-proxy*
- watches all services
- updates iptables when backends change
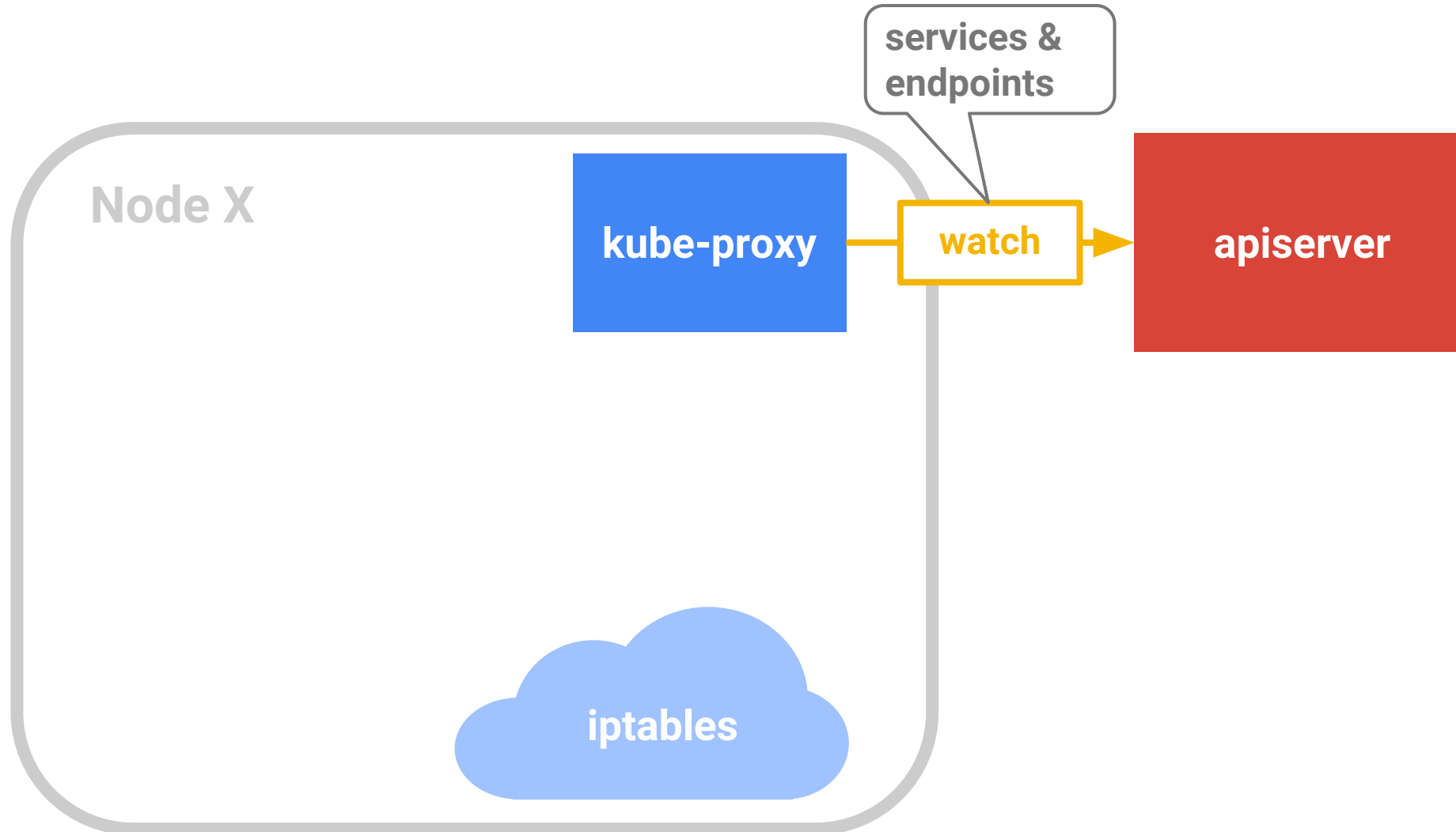- default implementation - can be replaced!
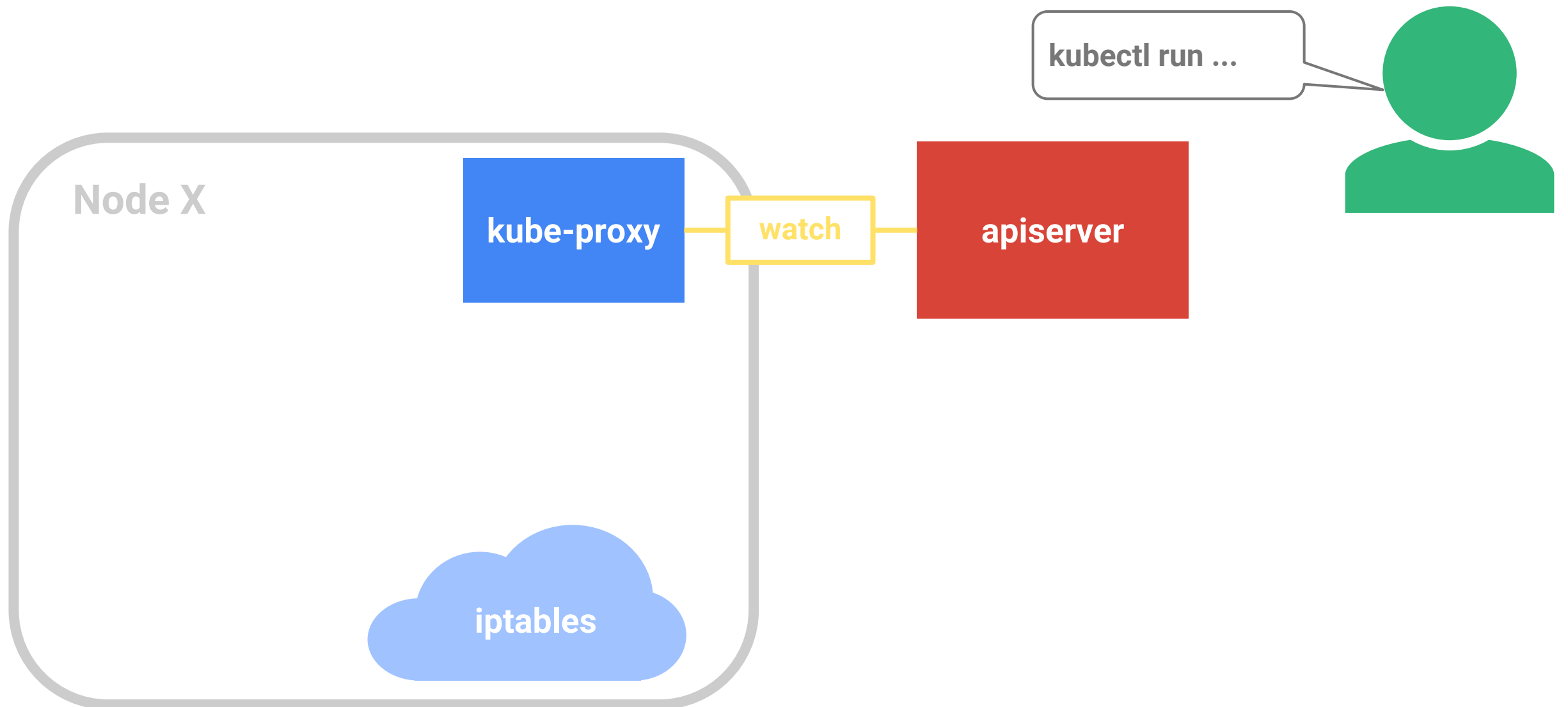
Hides complexity

Client

Virtual IP

iptables kube-proxy

iptables kube-proxy

iptables kube-proxy

iptables kube-proxy

Node X

kube-proxy — watch — apiserver

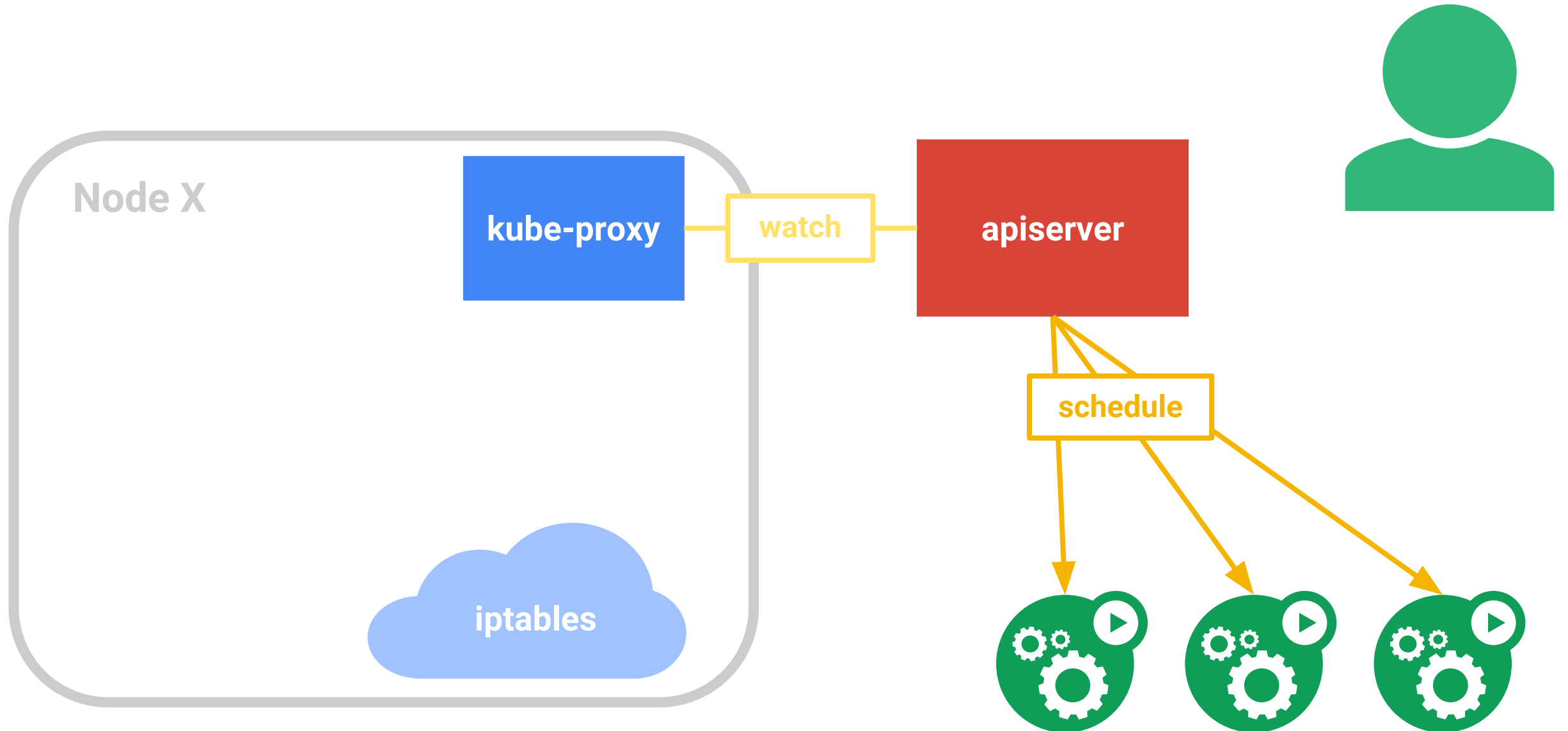iptables

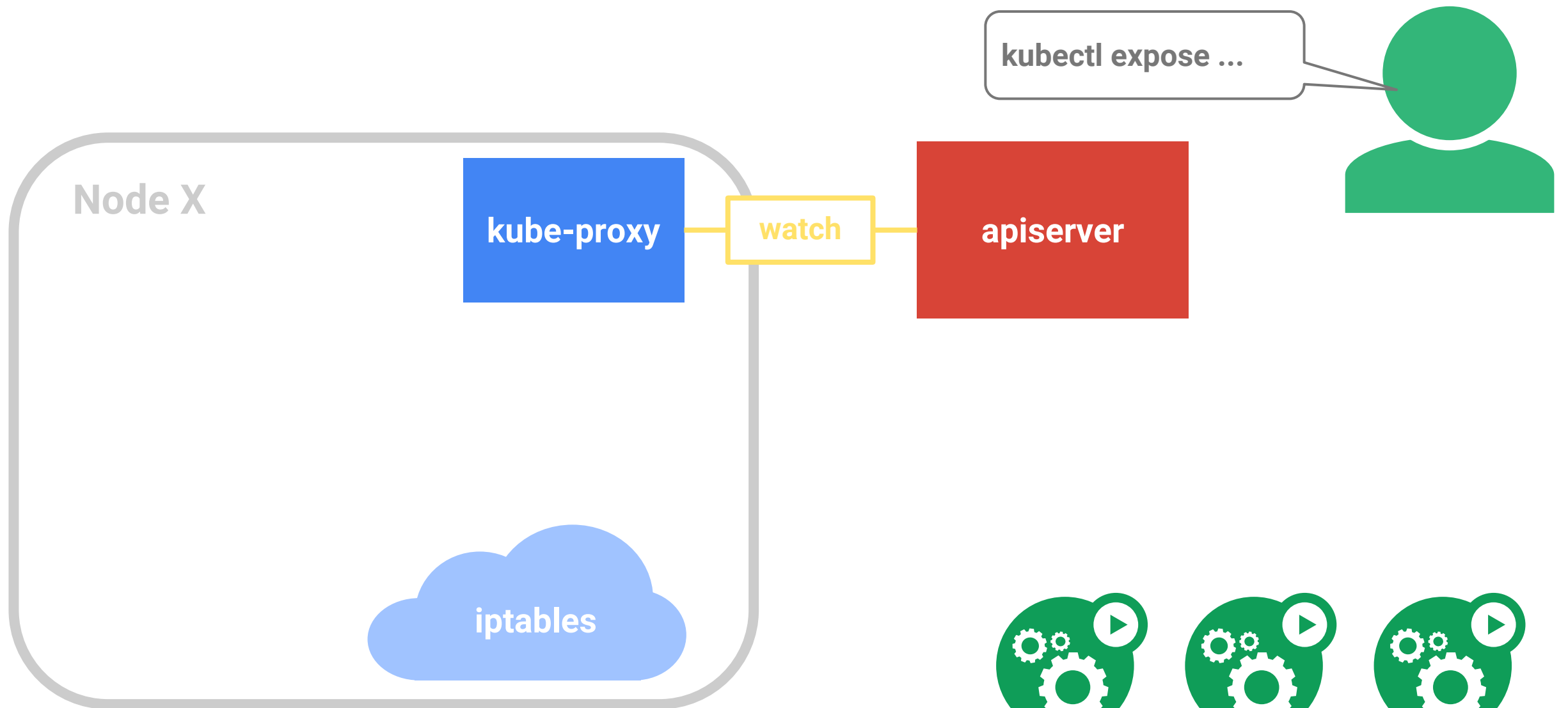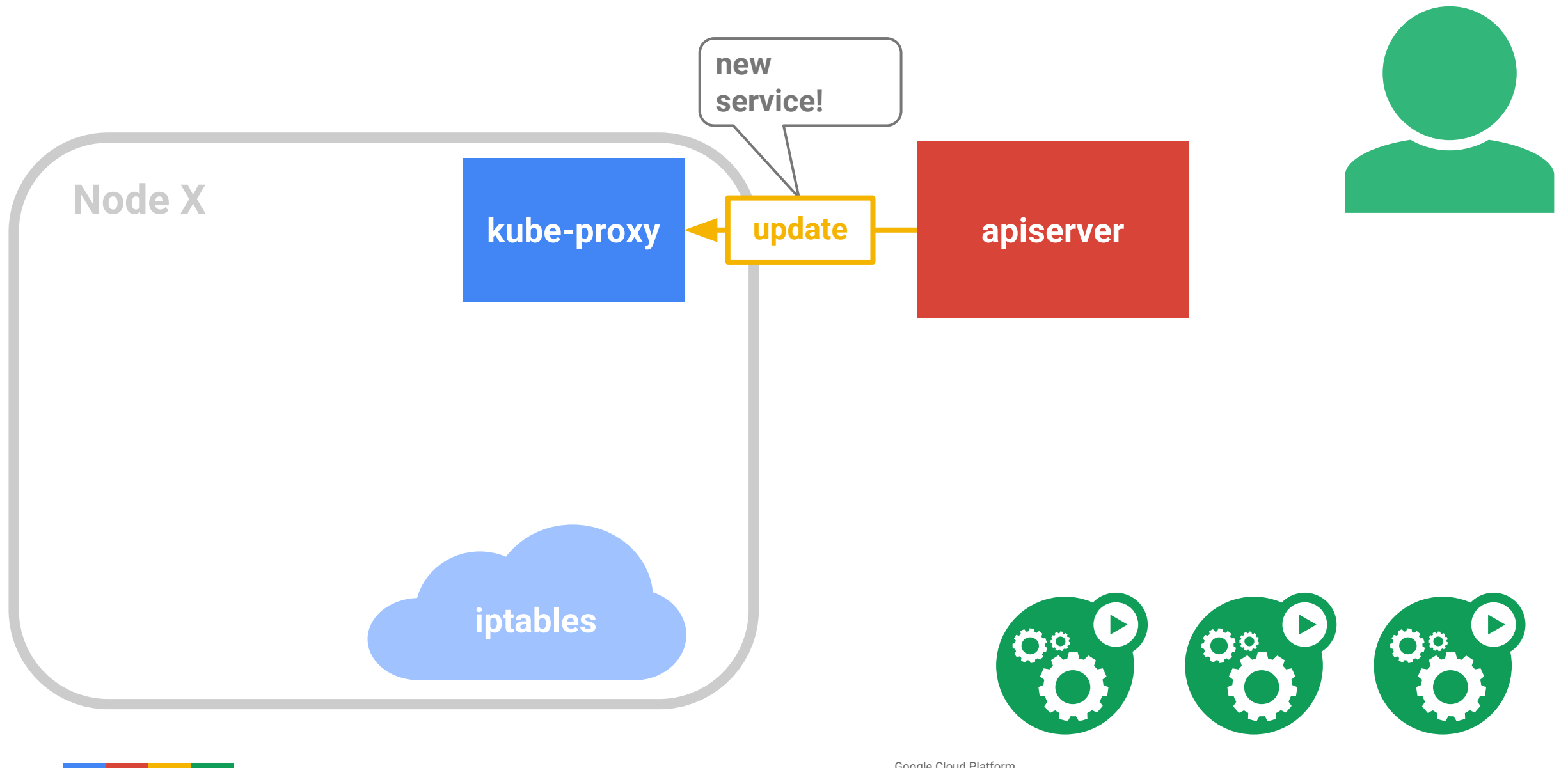schedule
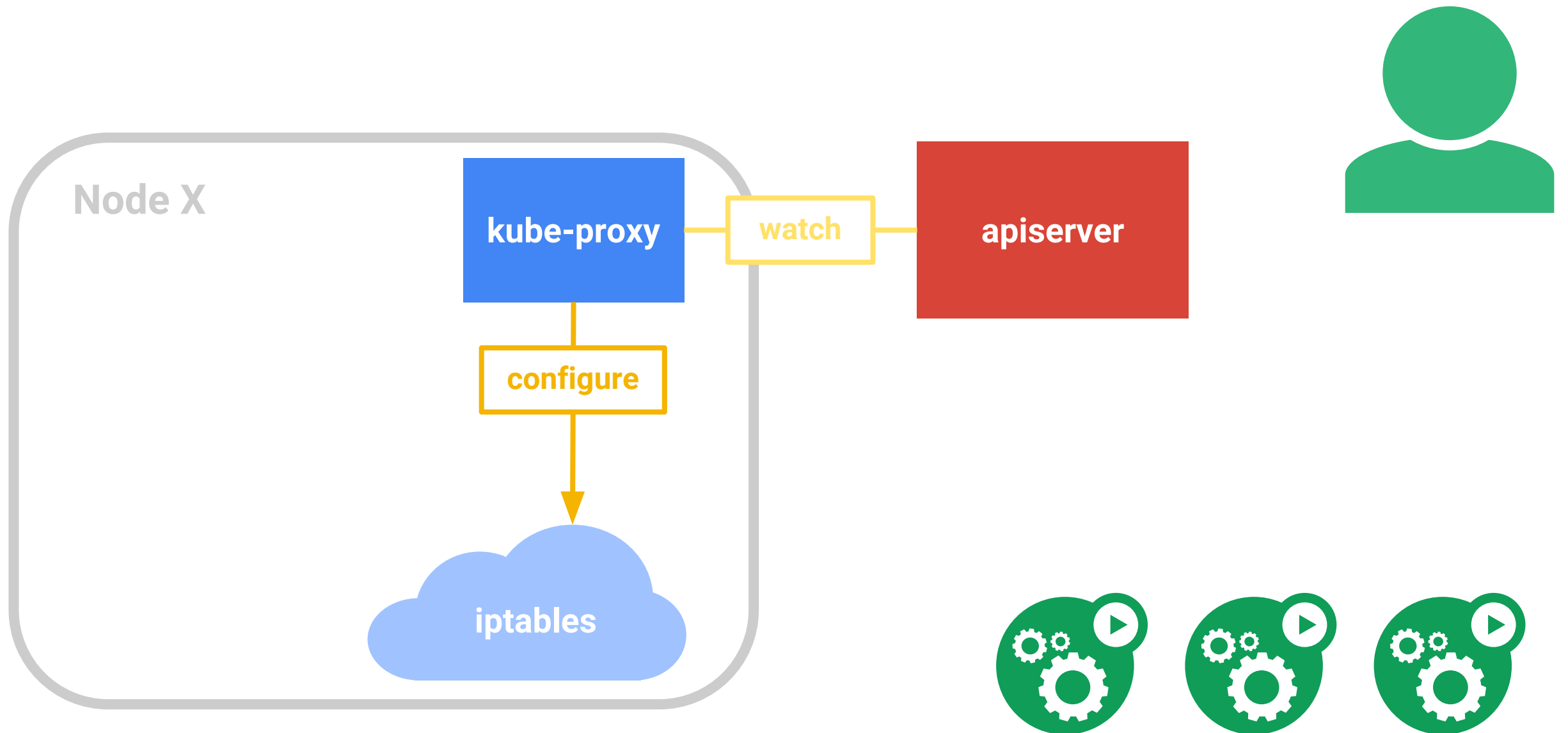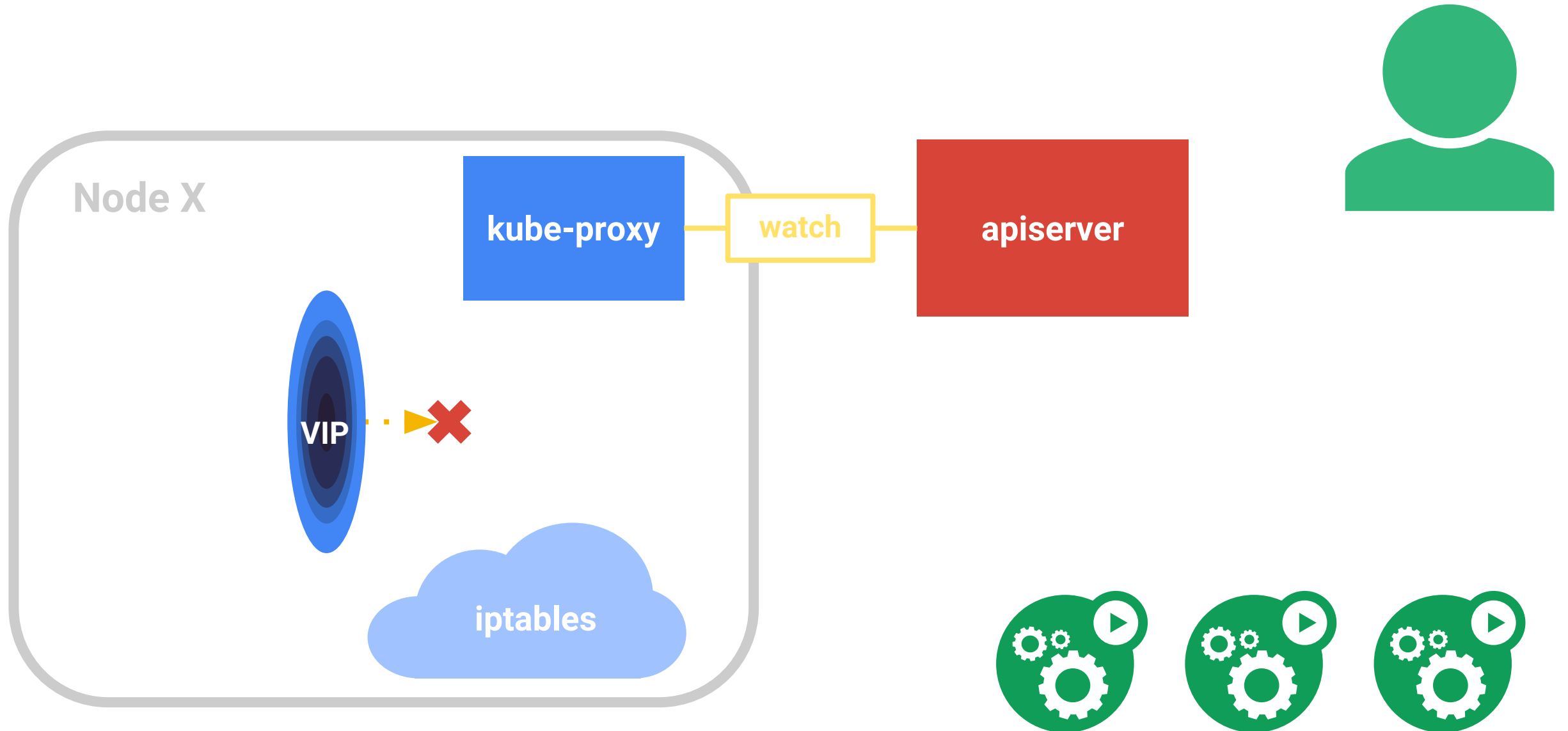
Google Cloud Platform

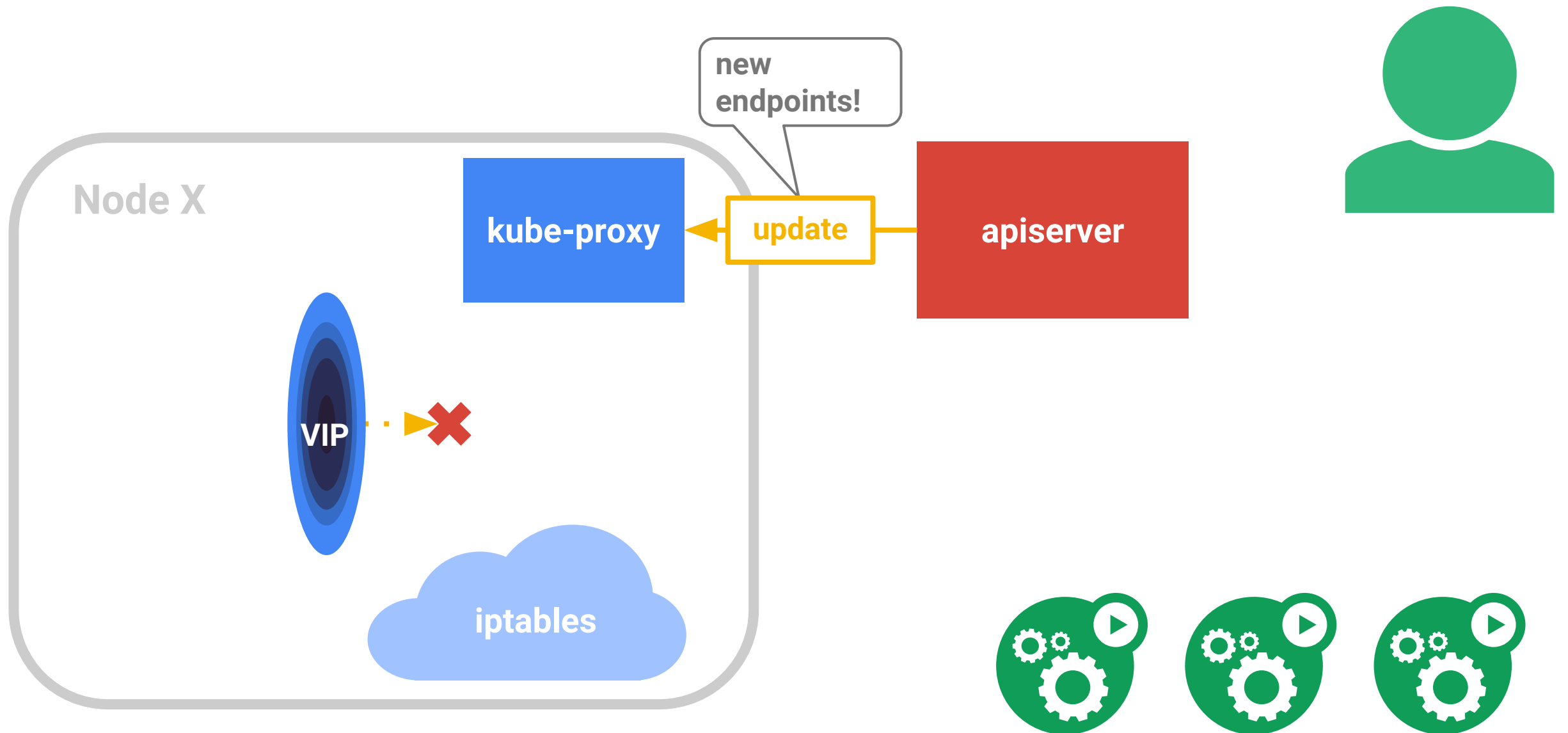# iptables kube-proxy

iptables kube-proxy

iptables kube-proxy

iptables kube-proxy

Node X

kube-proxy

watch

apiserver

VIP

iptables

Google Cloud Platform

iptables kube-proxy

iptables kube-proxy

Node X

kube-proxy

watch

apiserver

VIP

configure

iptables

Google Cloud Platform

iptables kube-proxy

Node X

kube-proxy — watch — **apiserver**

VIP

20

iptables

iptables kube-proxy

Node X

Client

kube-proxy —— watch —— apiserver

VIP

20

iptables

iptables kube-proxy

Node X

kube-proxy

watch

apiserver

Client

VIP

20

iptables

iptables kube-proxy

Node X

Client

VIP

iptables

kube-proxy

watch

apiserver

iptables kube-proxy

Node X

Client — VIP — iptables

kube-proxy — watch — apiserver
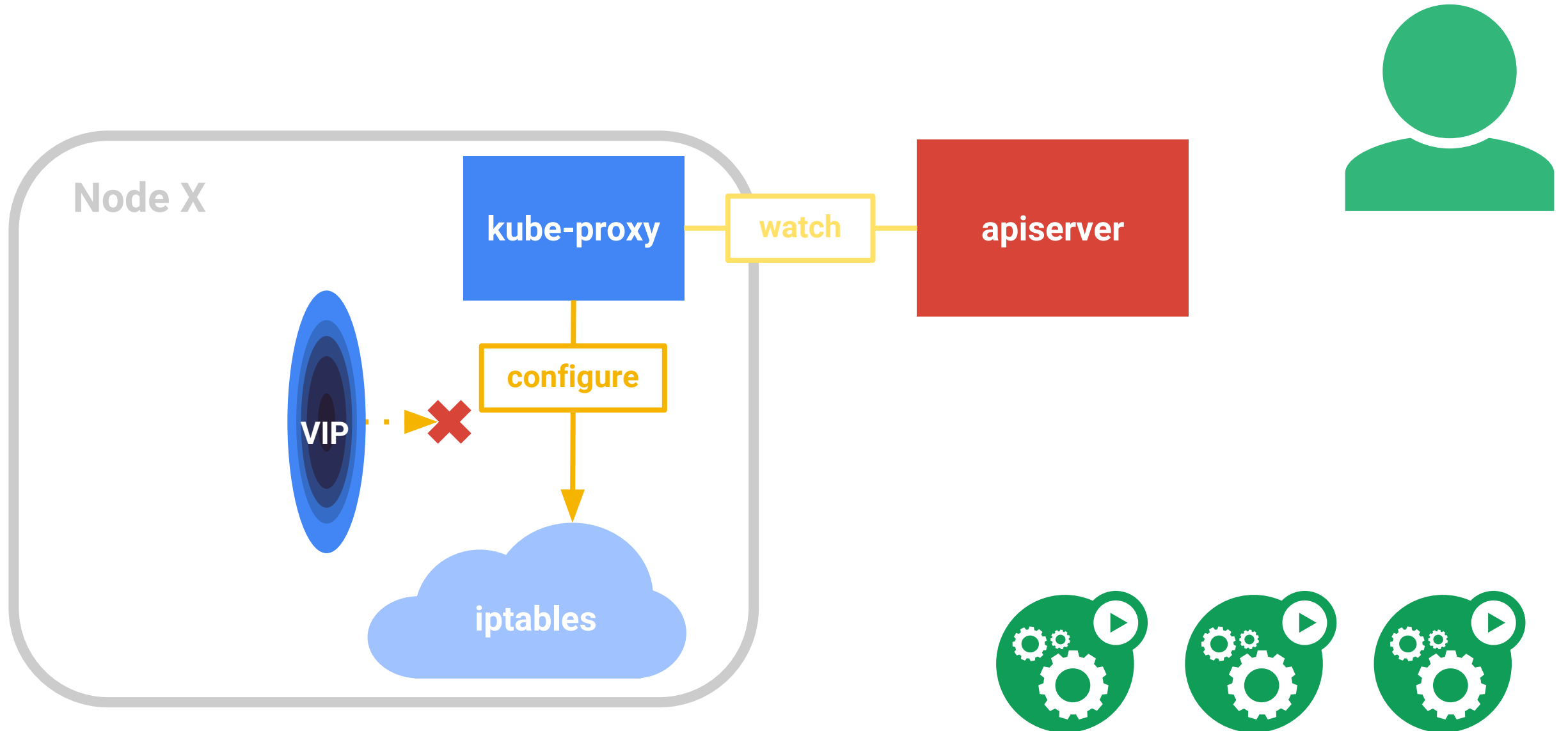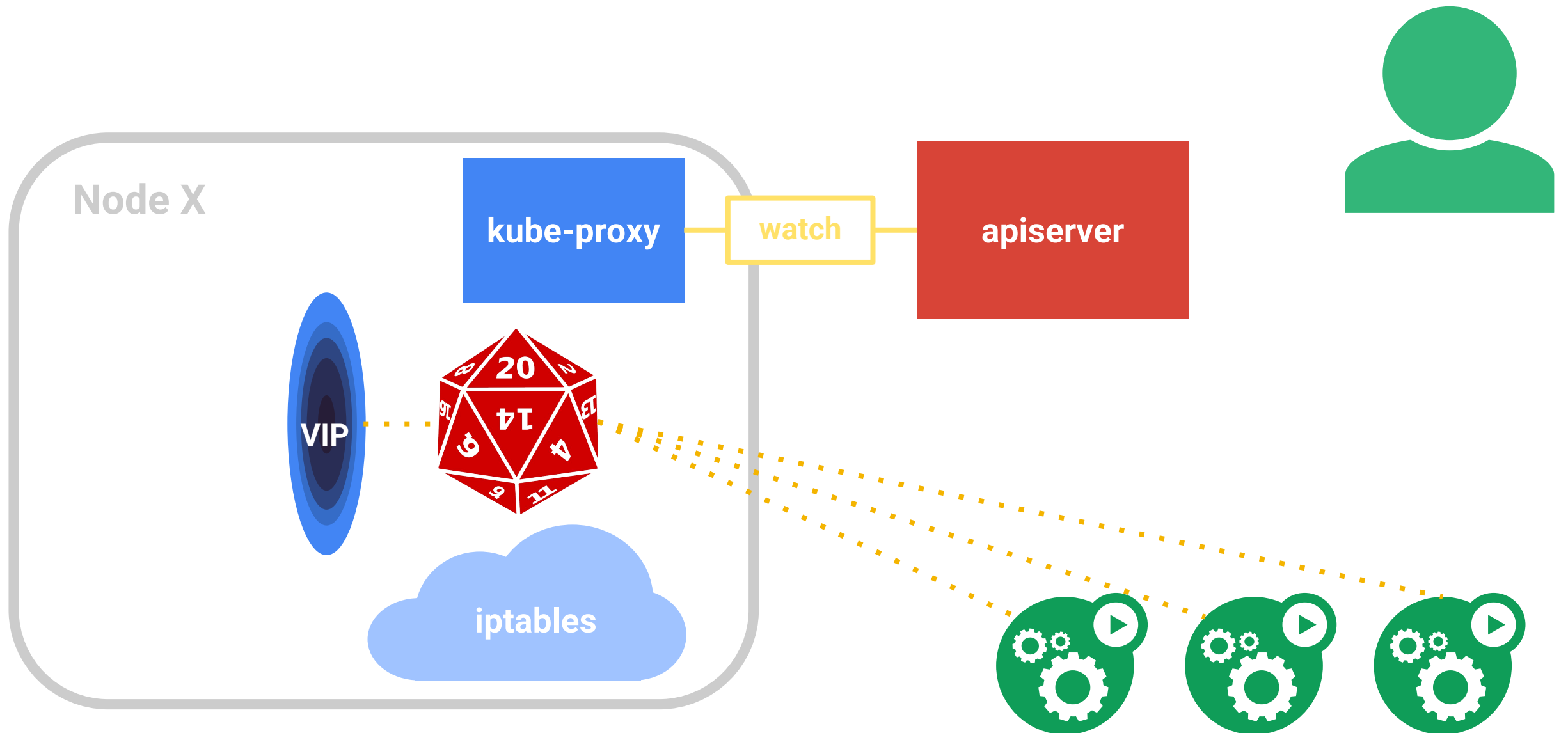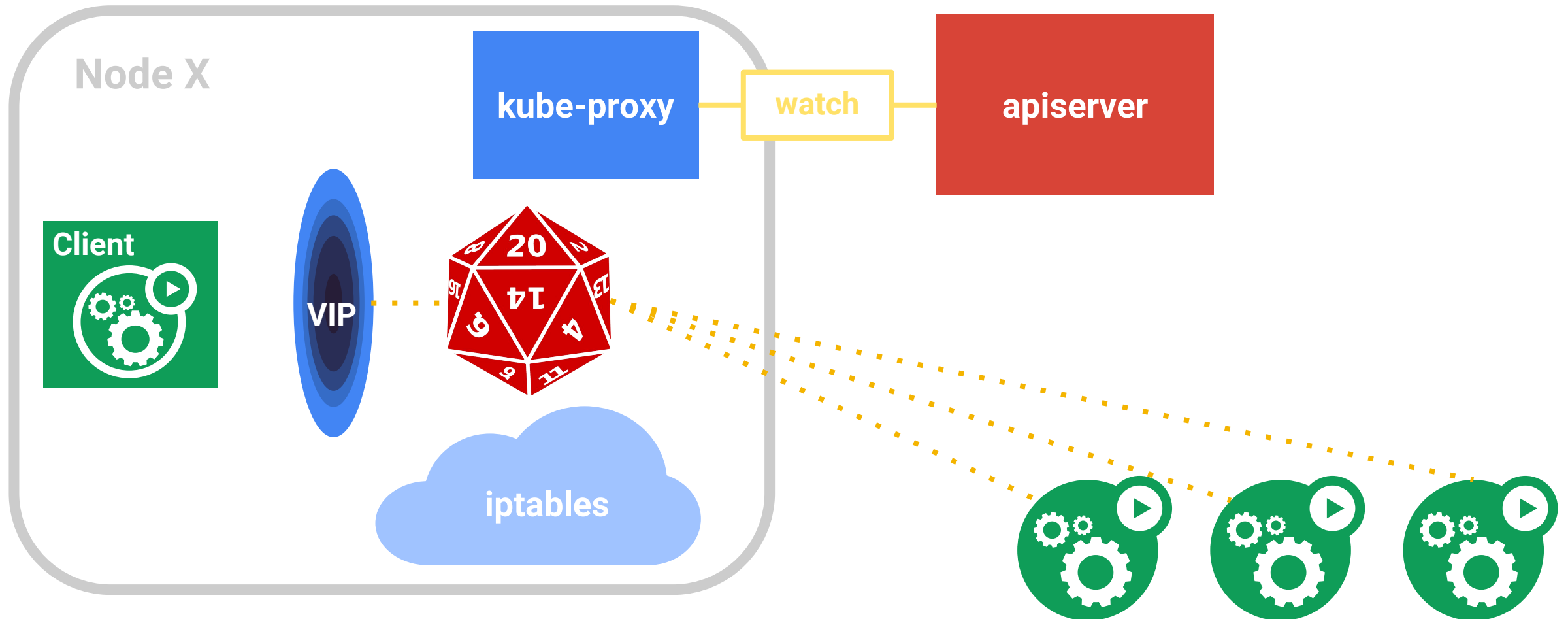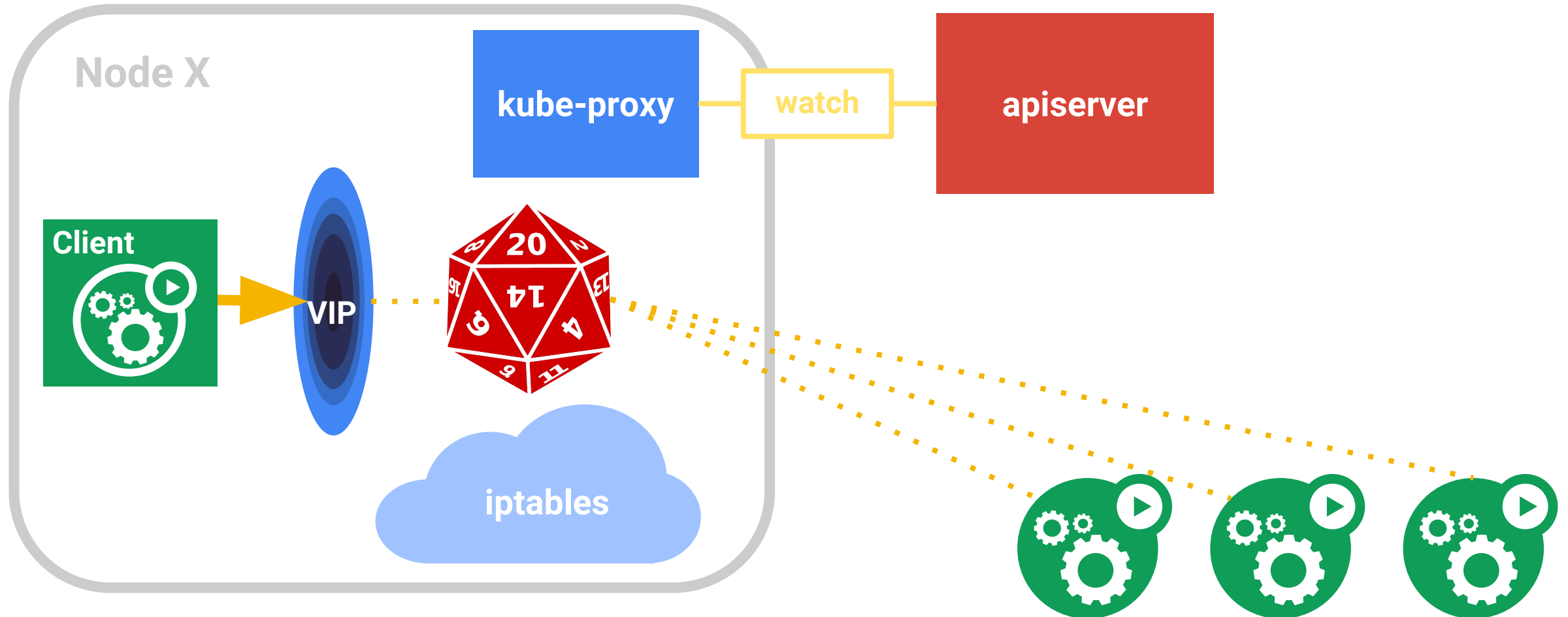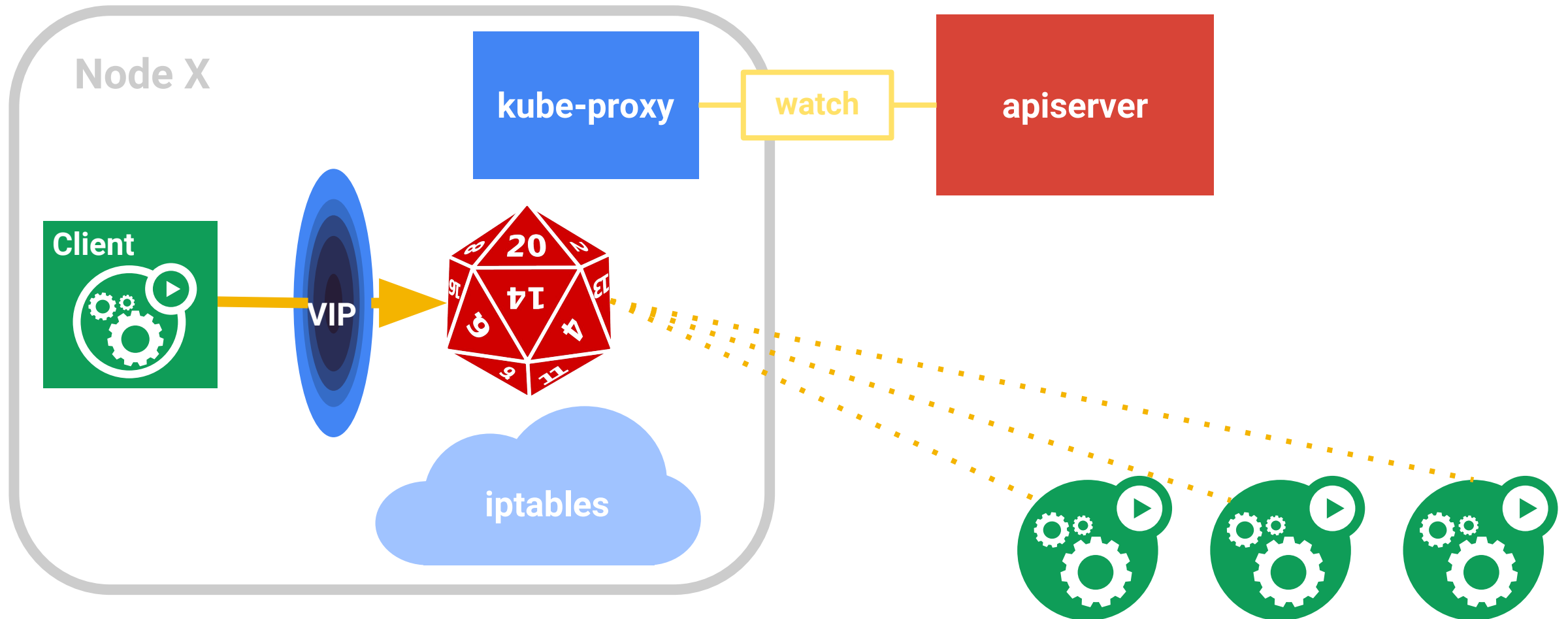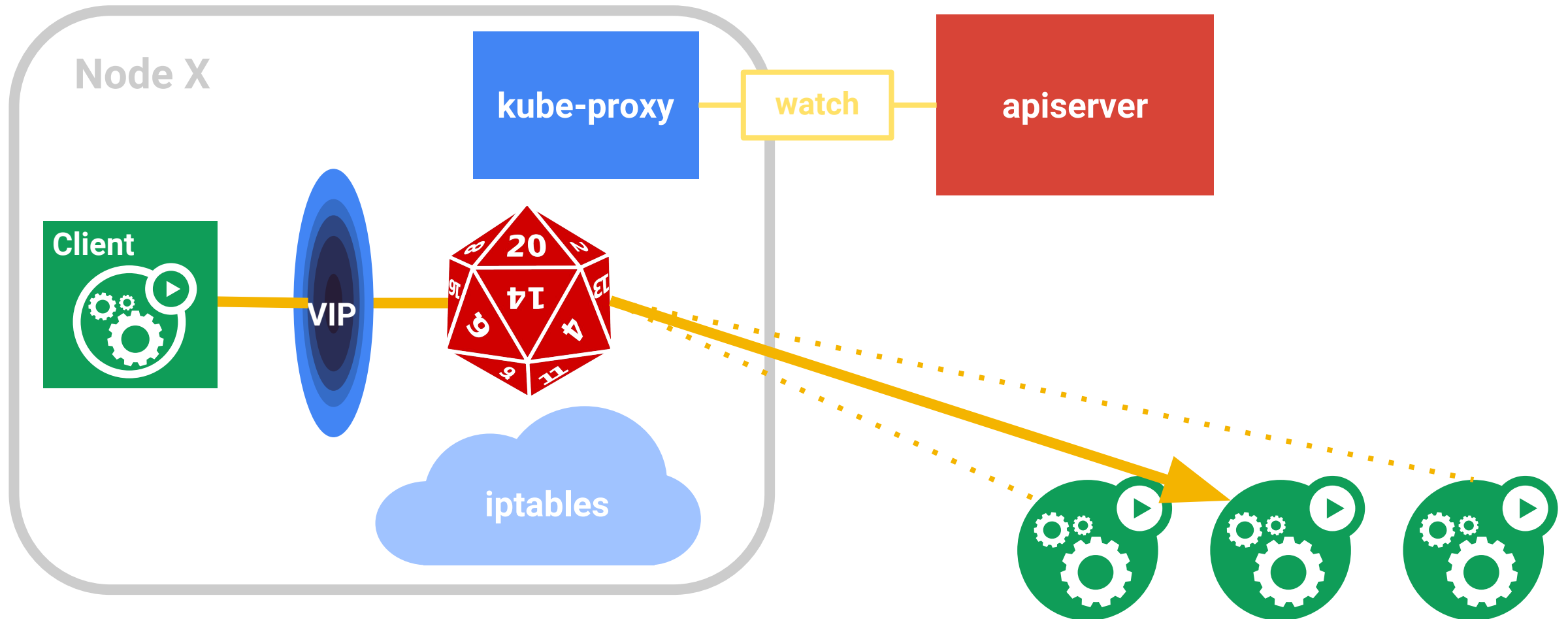
External services

Services VIPs are only available **inside** the cluster

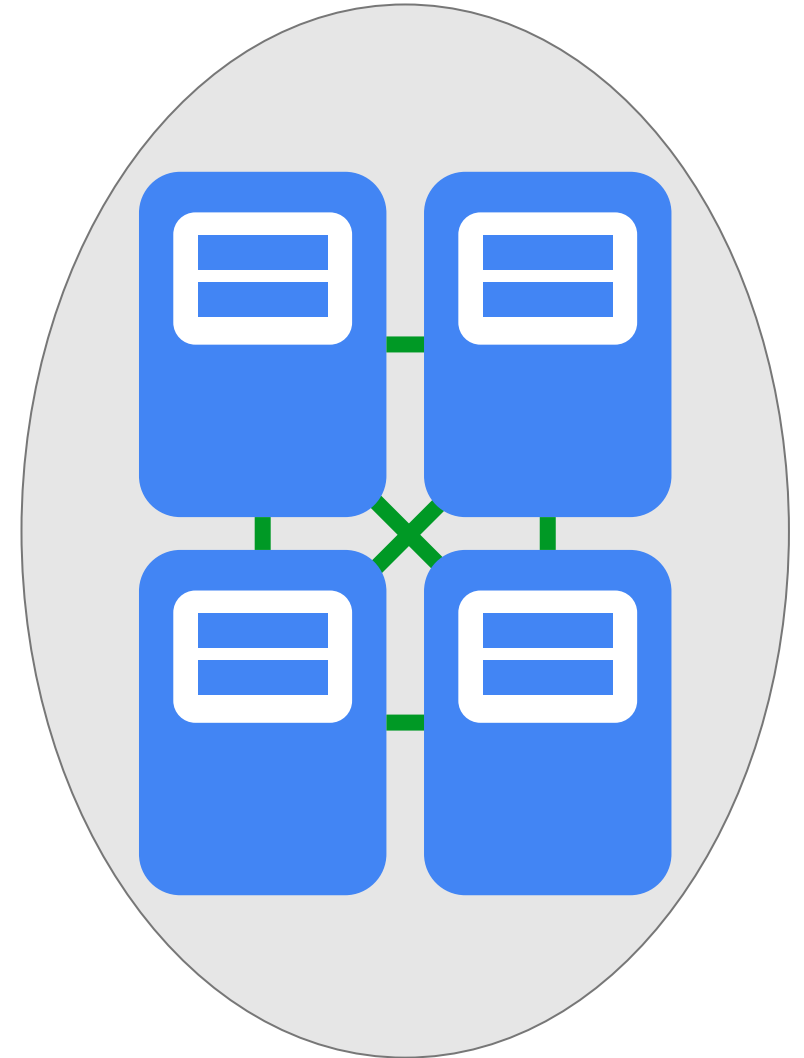Need to receive traffic from "the outside world"

Service "type"
- NodePort: expose on a port on every node
- LoadBalancer: provision a cloud load-balancer

DiY load-balancer solutions
- socat (for nodePort remapping)
- haproxy
- nginx

Ingress (L7 LB)

Ingress (L7 LB)

Many apps are HTTP/HTTPS

Services are L4 (IP + port)

Ingress maps incoming traffic to backend services
- by HTTP host headers
- by HTTP URL paths

HAProxy, NGINX, AWS and GCE implementations in progress

Now with SSL!

Status: BETA in Kubernetes v1.2

Google Cloud Platform