

Naše IT řešení vás nadchnou!

Motivace pro Kubernetes cluster on-premise



Motivací k úvaze o nasazení kontejnerové platformy a Kubernetes clusteru on-premise je realizace nových podnikových aplikací konceptem microservices "CloudReady". Cílem je pak nejen umožnit aplikacím migrovat do veřejného cloudového prostoru, ale také automatizovat vyvažování zátěže interní infrastruktury a automatizovat řadu procesů pro nasazování nových verzí aplikací.

Krátce o Kubernetes

Kubernetes je otevřený systém pro orchestraci virtualizace na úrovni operačního systému. Původně jej vyvinula společnost Google a jako podřízené nástroje podporuje například Docker. Můžeme hovořit i o virtuální kontejnerizaci na úrovni OS.



Výhody Kubernetes jsou následující:

1 ORCHESTRACE

- Možnost autoškálování podle zátěže.
- Možnosti škálování do šířky přidáním dalších worker-nodes do Kubernetes clusteru.
- HA konfigurace – v případě havárie worker-node automatická migrace podů (tj. container v Kubernetes) na jiný worker-node.
- Health-check a self-healing – orchestrate umožňuje nastavit health-check mechanismy pro startup i run.
- Traffic routing a load-balancing – nově definujeme aplikaci jako službu; ta je nasazená pomocí orchestrace v Kubernetesu. V rámci definice služby definujeme počet replik. Komunikace se službami se děje pomocí definice tzv. Ingressů – tzn. vstupních bodů na službu. Kubernetes pak směřuje požadavky uvnitř své sítě.
- Automatizované roll-outs a roll-backs, kdy je možné postupně "upgradovat" jednotlivé pody nesoucí aplikaci tak, aby pro uživatele nenastal žádný výpadek.
- Canary deployments – možnost otestovat novou verzi produkčního systému vedle původní verze, a pak jednoduše přepnout.
- Možnost rychle vystavět a zrušit jakoukoliv službu pomocí Kubernetes orchestrace.

— Automatizovaný deployment pomocí CI/CD pipeline – existují CI/CD nástroje, které se umí připojit přímo na API Kubernetes a deployovat na ně pomocí orchestračních konfiguračních souborů nové verze docker images.

— Transformace aplikace na "službu".

— Příprava na pozdější migraci do outsourcovaných cloudových služeb (AWS, MS Azure, Google cloud).

Koncept Kubernetes clusteru on-premise od GEM System



2 REALIZACE KUBERNETES CLUSTERU – KUBERNETES CLUSTER JE MOŽNÉ REALIZOVAT VÍCE ZPŮSOBY:

- Kubernetes v cloud (MS Azure, AWS, Google Cloud a další) – nezabýváme se správou platformy, ale platíme ji.
- Plain Kubernetes na Bare-metal je čistá instalace, naopak Docker Engine je poměrně náročná realizace platformy a rovněž je s ní spojena méně triviální správa v praxi.
- Předpřipravené distribuce obsahující Kubernetes a podpůrnou infrastrukturu (Red Hat OpenShift, Rancher, Kubespray, MetalK8s ...) – řádově snadnější realizace a správa, ale pouze platby za některé z uvedených distribucí.
- Existují i minidistribuce, které 100% naplňují funkcionalitu Kubernetes, ale současně je možné je provozovat s minimálními nároky na HW – vhodné pro vývojáře, pokud chtějí vyvíjet na svých workstationech/laptopech – např. minikube.

3 TYPICKÁ ARCHITEKTURA KUBERNETES CLUSTERU



- **Front reverse proxy** (v případě použití Nodeport), či pro chod v DMZ -> směřuje požadavky z Internetu/Intranetu na Kubernetes cluster. Je možno použít jakoukoliv softwarovou proxy (Apache HTTP Server, Nginx, HAProxy) nebo dedikovaný síťový prvek (F5 Networks, Cisco apod.).
- **Kubernetes cluster** – master servers, na nichž běží základní služby kubernetesu (kubeproxy, etcd – key/value databáze pro konfiguraci Kubernetes).
- **Kubernetes cluster** – worker-nodes, na těchto serverech běží jednotlivé workloads, tzn. jednotlivé pody s aplikacemi.
- **Úložiště** – buď hardwarové spolupracující s Kubernetes pomocí API, nebo softwarově definované úložiště (NFS, GlusterFS) a k nim případný storage-backend (diskové pole).
- **Pomocné systémy** – Monitoring a Log collection – Prometheus, Elastic Stack a další.
- **CI/CD platforma** – Atlassian nástroje, GitLab, Stack a další.

