

Container Orchestration. Docker

Ce este un container

Ne putem referi la un container ca la un proces sau un set de procese care rulează într-un mod izolat față de mașina gazdă, interacțiunea între cele două fiind făcută doar la nivel de kernel. Ce înseamnă asta? Atunci când un proces containerizat va fi nevoit să interacționeze cu kernel-ul sistemului, o va face cu kernel-ul mașinii gazdă.

Totodata, un astfel de proces poate avea acces la un set de librării disponibil doar containerului, astfel evitând poluarea host-ului cu date ce nu sunt necesare sistemului de operare sau altor aplicații.

Pentru a containeriza un proces, avem nevoie de un set de caracteristici și resurse de care acest proces va avea nevoie. Suma acestora se numește imagine. Un spațiu de stocare pentru imagini se numește registru (registry).

Interacțiunea cu containere

La nivelul liniei de comandă, putem interacționa cu containere prin mai multe metode:

- sistemul de management al containerelor oferit de sub-comanda “docker container”
- sub-comenzi ale comenzii “docker” care se adresează containerelor în mod explicit
- alte utilitare de orchestrare care interacționează cu Docker

Orchestrarea containerelor este procesul de implementare a containerelor pe un cluster de computere format din noduri multiple. Este un proces care automatizează implementarea prin numeroase caracteristici, incluzând:

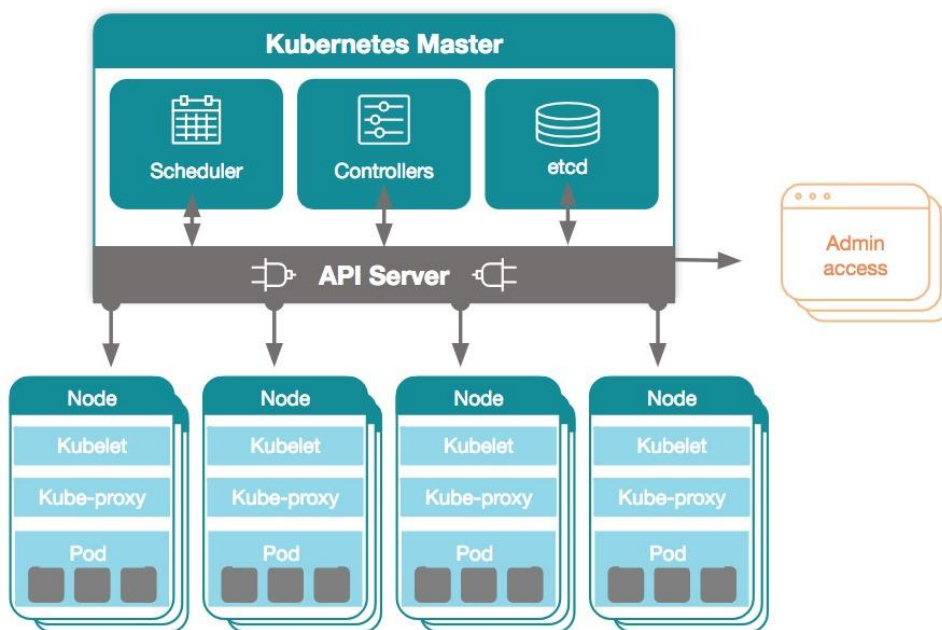
- instanțierea unui set de containere
- conectarea containerelor prin interfețe
- furnizarea serviciilor la mașinile din afara clusterului
- scalarea clusterului prin adăugarea sau eliminarea containerelor

Kubernetes este un sistem de orchestrare și clustering, întreținut de Google, care are particularități raportat la operațiile Docker native. Totuși, Kubernetes este cel mai popular sistem când vine vorba de medii de producere scalate care trebuie să se adapteze la diferite cerințe în ceea ce privește resursele pe baza încărcării și a traficului, cu grad mare de încredere.

Cel mai mare avantaj în utilizarea Kubernetes este instalarea automată de microservicii având la bază scripturi.

Configurarea unui load-balancer nu necesită multă informație sau experiență. Cu o singură comandă, Google Cloud rezervă IP-ul static și configurează load-balancer. Configurarea sistemului de filtrare și firewall este ușoară.

Kubernetes Architecture



Rancher este un tool util deoarece are un UI prietenos ce permite gestionarea facilă a containerelor și a clusterelor. Ca orice UI prietenos, vine cu prețul "ascunderii" elementelor interne Docker, făcând foarte grea monitorizarea și rezolvarea problemelor.

Rancher este foarte util în dezvoltare. Este ușor de configurat și instalat servicii, care mai apoi să fie configurate cu Rancher UI, pe care îl recomandăm pentru dezvoltarea rapidă a proiectelor de mici dimensiuni care utilizează un număr mic de containere. Este de asemenea logic să folosiți toată configurația de micro-servicii din aplicație atât din punctul de vedere al dezvoltării, cât și al structurării. Orice programator devops abil poate face acest lucru, iar sistemul final de producție se poate baza pe Kubernetes sau pe orice altă configurație.