

Diskless Kubernetes

Contents

1	Temat	2
2	Systemy orkiestracji kontenerami	2
2.1	Fleet	2
2.2	Docker Swarm	2
2.3	Kubernetes	2
2.4	Mesos	2
2.5	Rancher	2
3	Proces uruchamiania maszyn bezdyskowych	2
4	Systemy operacyjne	2
4.1	CoreOS	2
4.1.1	Konfiguracja	3
4.1.2	Tematy	3
4.1.3	Linki	3
4.2	RancherOS	3
4.2.1	Konfiguracja <code>cloud-config</code>	3
4.2.2	Tematy	3
4.2.3	Linki	3
4.3	Project Atomic	4
4.3.1	Linki	4
4.4	Alpine Linux	4
4.4.1	Konfiguracja	4
4.4.2	Tematy	4
4.4.3	Linki	4
4.5	ClearLinux	4
4.5.1	Linki	5

1 Temat

2 Systemy orkiestracji kontenerami

2.1 Fleet

Rozproszony system inicjalizacji systemu, czyli prosta nakładka na systemd.

2.2 Docker Swarm

Rozwiązanie od twórców Docker'a. Proste w konfiguracji, nie oferuje tak dużych możliwości jak niżej wymienione.

2.3 Kubernetes

Najpopularniejsze narzędzie orkiestracji kontenerami. Stworzone przez Google i bazowane na wewnętrznym systemie Borg.

2.4 Mesos

Zaawansowane narzędzie orkiestracji kontenerami,

2.5 Rancher

Platforma zarządzania kontenerami umożliwiającą między innymi zarządzanie klastrem Kubernetes.

3 Proces uruchamiania maszyn bezdyskowych

Na uruchamianie maszyn bezdyskowych w protokole PXE składają się 3 podstawowe elementy: 1. serwer DHCP, np. isc-dhcp-server lub dnsmasq 2. firmware wspierające PXE, np. iPXE 3. serwer plików (np. TFTP, HTTP, NFS)

4 Systemy operacyjne

4.1 CoreOS

Oryginalna dystrybucja linux'a dedykowana zarządzaniu kontenerami.

4.1.1 Konfiguracja

Konfiguracja przez Container Linux Config transpilowany do Ignition na popularne platformy cloud.

`cloud-config` wychodzący z użycia

4.1.2 Tematy

- brak kompilatora konfiguracji pod BSD
- przeciętny rozmiar 327 MB
- stosunkowo niewygodna konfiguracja (transpilowanie do jednego pliku)

4.1.3 Linki

- CoreOS - <https://coreos.com/>
- Ignition - <https://coreos.com/ignition/docs/latest/>
- Container Linux Config - <https://coreos.com/os/docs/latest/provisioning.html>
- Cloud Config - <https://coreos.com/os/docs/latest/cloud-config.html>

4.2 RancherOS

System operacyjny, w którym zamiast `init`'a jest 3-poziomowa konfiguracja dockera: `bootstrap_docker`, `system-docker` i `docker`.

- `bootstrap_docker` uruchamia system operacyjny
- `system-docker` pełni rolę `init`'a
- `docker` umożliwia użytkownikowi systemu uruchamianie kontenerów

4.2.1 Konfiguracja `cloud-config`

Sekwencyjnie uruchamiane pliki YAML bez konieczności

4.2.2 Tematy

- mały rozmiar (45 MB)
- możliwość wielokrokowej konfiguracji bez transpilacji

4.2.3 Linki

- <https://rancher.com/rancher-os/>
- `cloud-config` - <http://rancher.com/docs/os/v1.1/en/configuration/>

- <https://forums.rancher.com/t/good-cloud-config-reference/5238/2>

4.3 Project Atomic

Konfiguracja systemów operacyjnych dedykowana środowiskom cloud i kontenerom. Podobnie skonfigurowany jeden z trzech systemów operacyjnych (tzw. Atomic Host): CentOS, Fedora lub Red Hat.

4.3.1 Linki

- <https://www.projectatomic.io/>
-

4.4 Alpine Linux

Minimalna dystrybucja Linux'a bazowana na musl-libc i busybox.

- nie bootuje się w trybie diskless (tworcy nie umieją naprawić błądów)

4.4.1 Konfiguracja

Alpine Backup z lbu - spakowane pliki wypakowywane w sekwencji boot'u

4.4.2 Tematy

4.4.3 Linki

- <https://alpinelinux.org/>
- Alpine Backup (lbu) - https://wiki.alpinelinux.org/wiki/Alpine_local_backup
- Alpine Configuration Framework - http://wiki.alpinelinux.org/wiki/Alpine_Configuration_Framework_Development

4.5 ClearLinux

- "bundle" zamiast pakietów systemowych aktualizowane z całym systemem
- skoncentrowany na wydajności na procesorach Intel
- niewygodny format dokumentacji (brak kompletnej mapy dokumentacji, duże zagłębienia)
- lokalizacja wszystkich modyfikacji w /var i /etc (prosty reset)

4.5.1 Linki

- <https://clearlinux.org/>
- <https://www.infoworld.com/article/3159658/linux/6-key-points-about-intels-hot-new-linux-distro.html>