

Kurs DevOps

Lista 8

4 i 10 grudnia 2025

Jako że zazwyczaj w dużych grupach znajduje się co najmniej jeden użytkownik Windowsa, to informuje się, iż zadania należy wykonywać z linii poleceń, o ile nie powiedziano inaczej w treści.

Używanie pluginów/rozszerzeń open-source do Ansible, Kubernetesa i Jenkinsa jest dozwolone, o ile w treści zadania nie powiedziano inaczej.

Zadanie 1.

Zapoznaj się z RabbitMQ bądź z Redis. Do czego służy wybrany przez Ciebie system? Zademonstruj, na prostym przykładzie jak go wykorzystać.

Zadanie 2. (3 p.)

Na SKOS-ie znajduje się archiwum ze zdjęciami kilku stron książki *Artis Magnae Artilleriae pars prima*¹ Kazimierza Siemienowicza, wraz ze skryptem do przetwarzania zdjęć (bez używania AI) na tekst, który bardziej lub mniej nadaje się do druku. Zademonstruj, w jaki sposób użyć Job z kubernetesa w celu wykonania przetwarzania załączonych skrypt w sposób rozproszony w klastrze.

Skrypt w bashu używa:²

- Pythona 3.13.7
- GNU Parallel 20251022³
- page-dewarp 0.2.0⁴
- ImageMagick 7.1.2.8⁵
- unpaper 7.0.0⁶

Zadanie 3. (2 p.)

Pokaż, w jaki sposób z zastosowaniem **Ingress** wystawić dwa różne serwery HTTP poprzez wspólne API. W tym celu rozszerz zadanie 2 o:⁷

- Przechowywanie oryginalnych zdjęć na serwerze A.
- Wysyłanie wyników Job na serwer B.

Następnie skonfiguruj **Ingress** tak, by wystawiać klientowi jeden adres dostępowy, który w zależności od adresu HTTP będzie przekierowywał na odpowiedni serwer (np. `/org` na serwer A i `/res` na serwer B). Zademonstruj wynik działania.⁸

Zadanie 4.

Pokaż jak przenieść (usunąć) wszystkie pody z jednego noda na drugi bez przerywania działalności serwisu (np. w przypadku, gdy trzeba wymienić serwer).⁹

¹Wielkiej sztuki artylerii część pierwsza

²Podaję wersje, na których testowałem skrypt podczas tworzenia listy zadań, na starszych także powinien działać.

³<https://www.gnu.org/software/parallel/sphinx.html>

⁴<https://github.com/lmmx/page-dewarp/tree/master>

⁵<https://usagemagick.org/>

⁶<https://github.com/unpaper/unpaper>

⁷Jeśli nie zrobiłeś/zrobiłaś zadania 2, to za połowę punktów możesz po prostu zademonstrować konfigurację **Ingress** na dwóch serwerach o różnej zawartości.

⁸Jeśli jako serwer HTTP używasz **miniserve** może się przydać opcja `--route-prefix`.

⁹Może `taint` i `toleration` będą przydatne?

Zadanie 5.

Zademonstruj, w jaki sposób z użyciem `StatefulSet` i `local PersistentVolume`¹⁰ stworzyć serwis, którego pody będą używały danych nieulotnych.

Zadanie 6.

- a) Do czego służy opcja `imagePullPolicy: Never`.
- b) Zademonstruj jak skonfigurować i użyć GUI Kubernetesa.
- c) Podaj kilka (min. 3) przykładów serwisów, które nie przechowują danych (są *stateless*) i jednocześnie moglibyśmy chcieć je zdeployować z użyciem Kubernetesa.

Zadanie 7. ★

Opowiedz o Kubernetes Event Driven Autoscaler (KEDA)¹¹ i następnie zaprezentuj jego działanie.

¹⁰<https://kubernetes.io/docs/concepts/storage/volumes/#local>

¹¹<https://keda.sh/>