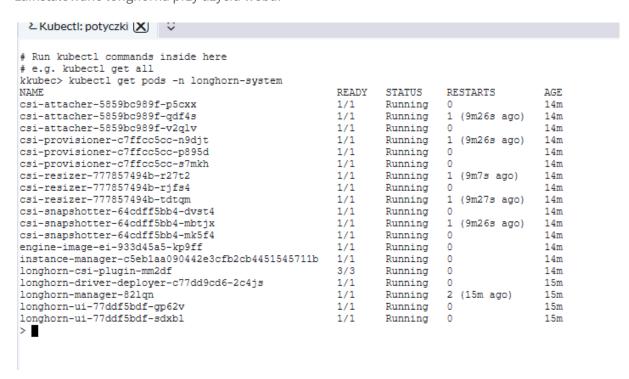
# 404 Not Found (sslip.io)

cała usługa została utworzona przy użyciu webui ranchera za wyjątkiem obrazu który został zbudowany i wrzucony do repo lokalnie

#### #zad.2

zainstalowano ręcznie wymagane paczki i aktywowano brakujący moduł kernela, następnie zainstalowano longhorna przy użyciu webui



### #zad. 3

dodaję repozytorium rodeo do aplikacji, without authentication

instaluję aplikację z webui

SUSE Demo: Tetris Game (sslip.io) tetris działa i ma się dobrze

### #zad.4

neuvector zainstalowany z webui

autoscan włączony

#zad.5

dodano repo i włączono cykliczne skany

wynik skanowania



nvbeta/node:latest ma najwięcej podatności

### #zad. 6

Aby zachować PersistentVolume po przeskalowaniu do 0 i automatycznie podłączyć się do niego ponownie przy wznowieniu usługi, możemy wykorzystać StatefulSet zamiast Deployment. StatefulSet jest przeznaczony do zarządzania stanowymi aplikacjami, takimi jak bazy danych, które wymagają stałego identyfikatora w klastrze.

W pliku YAML (nie-usuwaj.yaml) zmieniamy rodzaj zasobu z Deployment na StatefulSet. Aby zmodyfikować używamy komendy kubectl apply -f nie-usuwaj.yaml

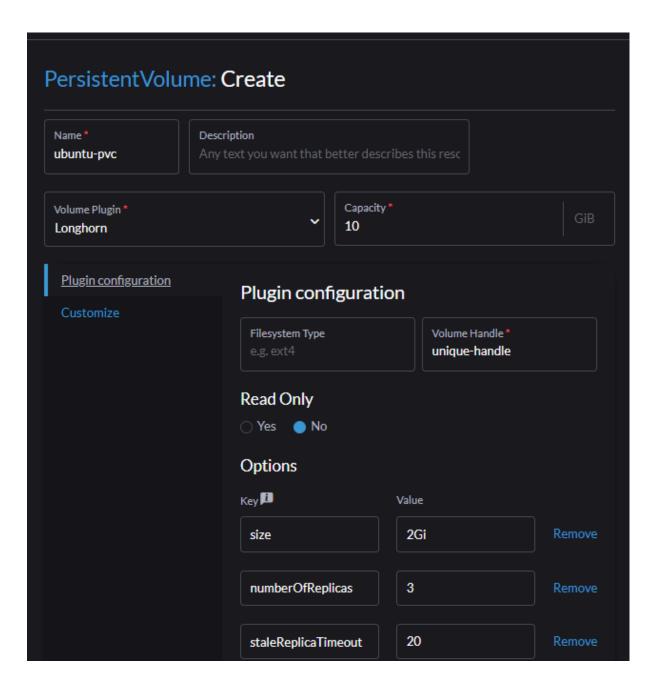
# np. nie-usuwaj.yaml:

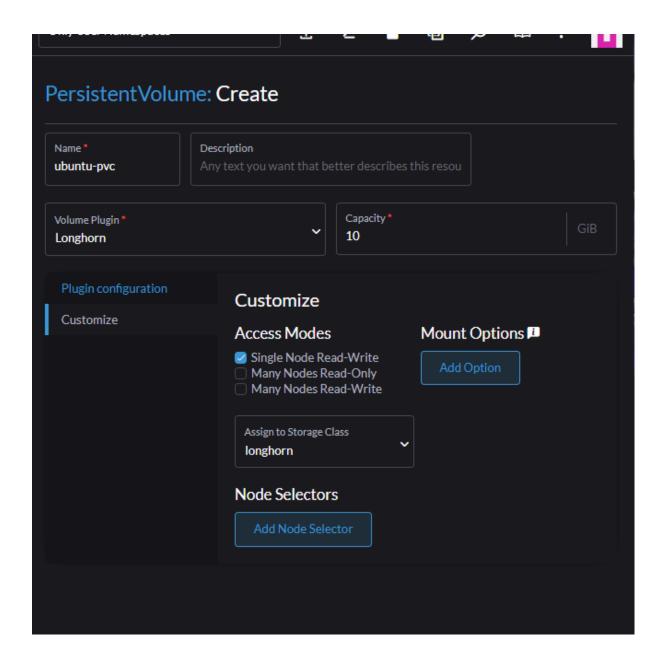
```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
   name: ubuntu-pvc
spec:
   accessModes:
   - ReadWriteOnce
   storageClassName: longhorn
   resources:
      requests:
      storage: 1Gi
---
apiVersion: apps/v1
kind: StatefulSet
```

```
metadata:
spec:
          - "86400"
```

```
volumes:
- name: volv

persistentVolumeClaim:
    claimName: longhorn-volv-pvc
```





podjęliśmy też próbę wykonania tego w rancherze

#zad. 7

metoda 1:

kubectl get all -n <nazwa\_namespace>

np. kubectl get all -n nginx

wybieramy resource, którego konfigurację chcemy wyświetlić, u mnie replicaset.apps/nginx-deployment-7f975d9949

kubectl edit <nazwa\_resource> -n <nazwa\_namespace>

np. kubectl edit replicaset.apps/nginx-deployment-7f975d9949 -n nginx otwiera nam się konfiguracja yaml

metoda 2:

Należy wejść w klaster->projects/namespaces->wybrać projekt, np. Project: web-server i tam kliknąć np. w nginx-> kliknąć reosurce, który chcemy, np. configmaps -> pod nazwą ConfigMaps będzie przycisk Download YAML, który po kliknięciu pobierze plik yaml zasobu

#zad.8

Gateway w Kubernetes to zasób, który reprezentuje instancję infrastruktury obsługującej ruchGatewayClass opisuje klasę dostępnych bram dla użytkownika, a HTTPRoute definiuje reguły specyficzne dla protokołu do przekierowywania ruchu do usług w klastrze. Narzędzie to umożliwia dynamiczne zarządzanie infrastrukturą sieciową i trasowanie ruchu.

#zad.9

Deployment umożliwia łatwe wdrożenie nowych wersji aplikacji, automatycznie tworząc lub usuwając pody.

Deployment obsługuje aktualizacje w trybie "rolling", zapewniając płynne przejście między wersjami, pozwala wrócić do poprzedniego deploymentu jeśli nowy jest niestabilny

Natomiast ReplicaSet utrzymuje określoną liczbę replik podów, zapewniając dostępność określonej liczby podów, nie ma wbudowanych mechanizmów do aktualizacji wersji.

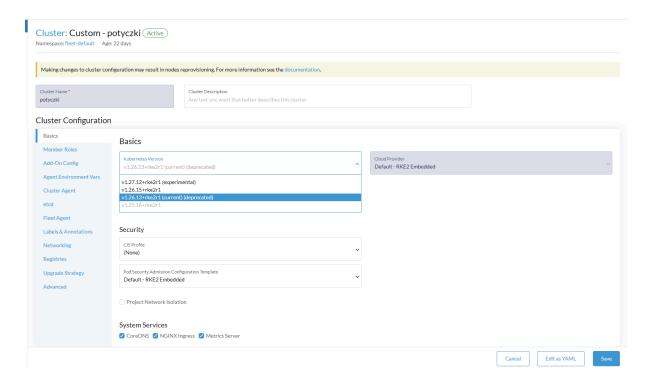
Deployment korzysta z ReplicaSet jako mechanizmu do zarządzania replikami

#zad.10

dobrą praktyką przed zaktualizowaniem klastra albo czegokolwiek jest zrobienie backupu więc wykonaliśmy snapshota

złą praktyką jest potencjalne uszkodzenie całe systemu pół godziny przed końcem czasu więc tego nie zrobiliśmy (mam nadzieję)

ale można to zrobić tutaj:



## #zad.11

użytkownik muhammed.yassuff z hasłem DSYoyt7AQNrT9neR (obowiązek zmiany hasła przy logowaniu)

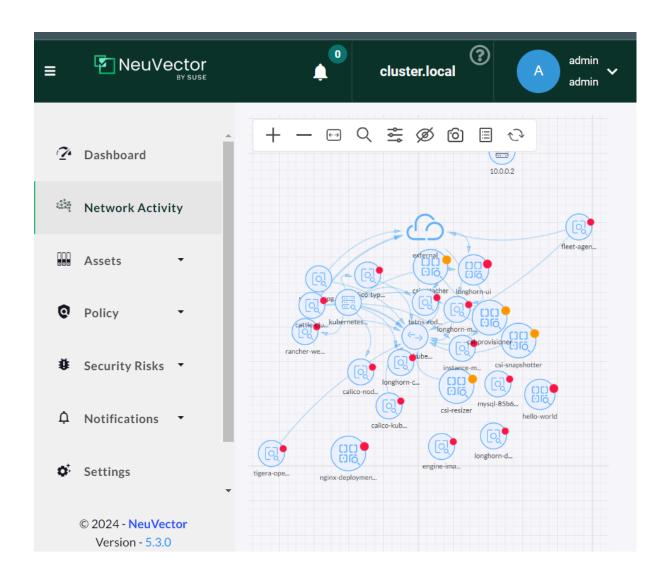
użytkownik muhhamad.yussuff z hasłem Ygo3aak1ElbuVenS (obowiązek zmiany hasła przy logowaniu)

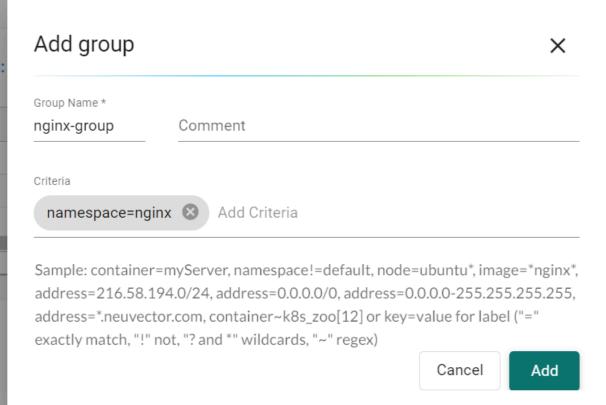
#zad.12

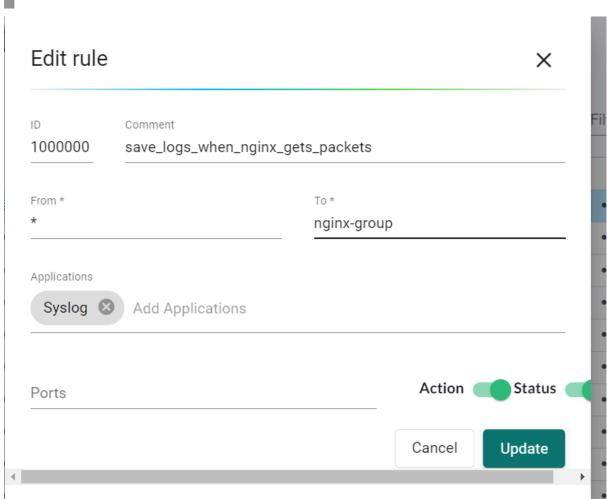
kontener crashuje przy uruchamianiu

zwiększenie ilości dostępnej pamięci do 512mib eliminuje problem

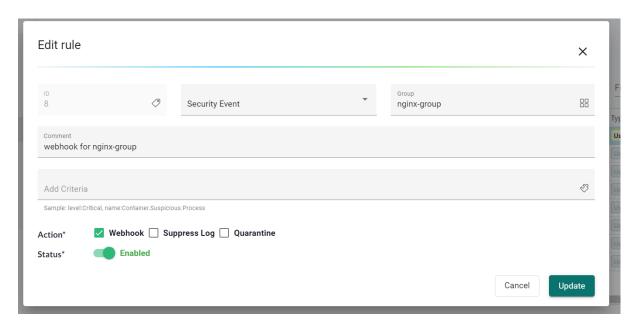
#zad. 13

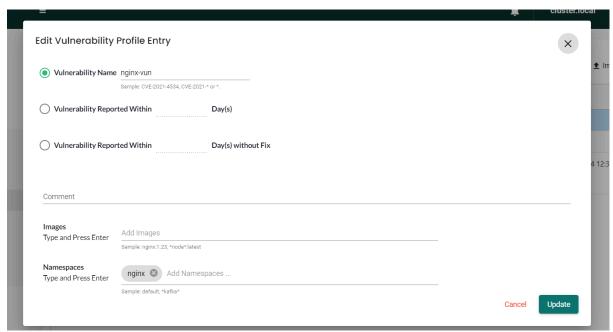






jeśli dobrze rozumiem nie ma blokować tych pakietów i powinny się zapisywać logi, gdy coś przyjdzie do nginx





vunerabilities wyświetlą się w Security Risks->Vunerabilities

Analiza pakietów to proces monitorowania, przechwytywania i badania ruchu sieciowego w celu zrozumienia, diagnozowania i rozwiązywania problemów związanych z siecią. Narzędzia do analizy pakietów pozwalają na szczegółowe badanie pojedynczych pakietów, identyfikację anomalii, wykrywanie ataków oraz optymalizację wydajności sieci. Możemy do tego użyć np. Wireshark

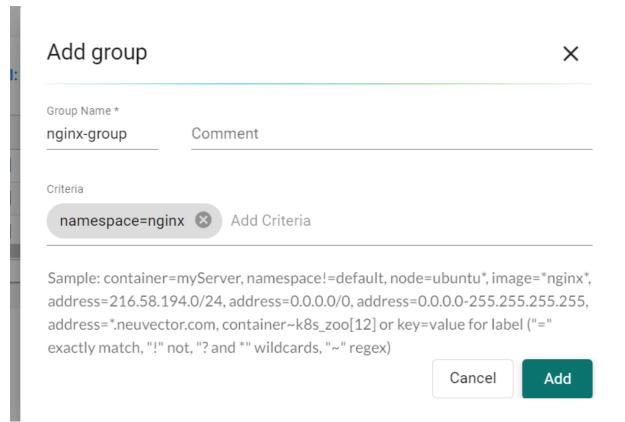
kubectl create namespace adrian

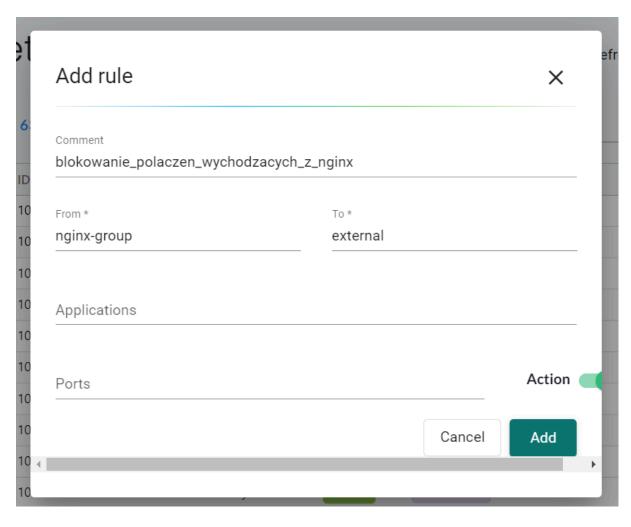
adrian nie przypisał limitów zasobów oraz dodał healtchecki które nie przechodzą i zabijają kontener

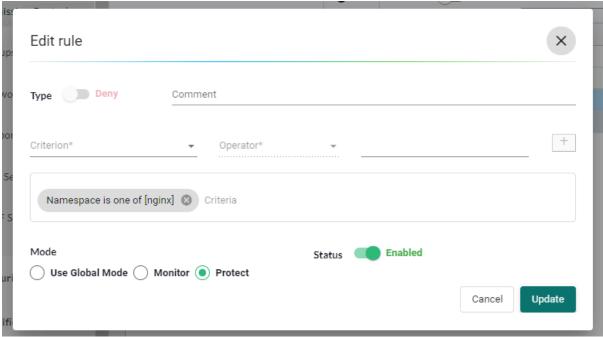
możliwa naprawa to wyłączenie healtchecków lub zmienienie konfiguracji nginxa żeby przechodziły

my wyłączyliśmy healtchecki

#zad. 15







do pliku yaml zakradł się błąd w linijce 21. zmiana MYSQL\_ROT\_PASSWORD na MYSQL\_ROOT\_PASSWORD pozwala na uruchomienie poda

#### zadanie 18

adrian korzystał z repozytorium nieposiadającego obrazu dla architektury x86\_64

zmiana repozytorium na poprawne powoduje uruchomienie się kontenera i wyświetlenie wiadomości w logach

```
Mon, Apr 8 2024 12:20:08 pm

2024-04-08T12:20:08.742824068+02:00 Hello from Docker!
2024-04-08T12:20:08.742837188+02:00
2024-04-08T12:20:08.742837488402+02:00 To generate this message, Docker took the following steps:
2024-04-08T12:20:08.742847879+02:00
2024-04-08T12:20:08.742847879+02:00
2024-04-08T12:20:08.74285499+02:00
2024-04-08T12:20:08.74285099+02:00
2024-04-08T12:20:08.742866042+02:00
2024-04-08T12:20:08.742866042+02:00
2024-04-08T12:20:08.742876819+02:00
2024-04-08T12:20:08.742876819+02:00
2024-04-08T12:20:08.7428809402:00
2024-04-08T12:20:08.7428809402:00
2024-04-08T12:20:08.7428809402:00
2024-04-08T12:20:08.7428809402:00
2024-04-08T12:20:08.7428809402:00
2024-04-08T12:20:08.742890702+02:00
2024-04-08T12:20:08.742905051+02:00
2024
```