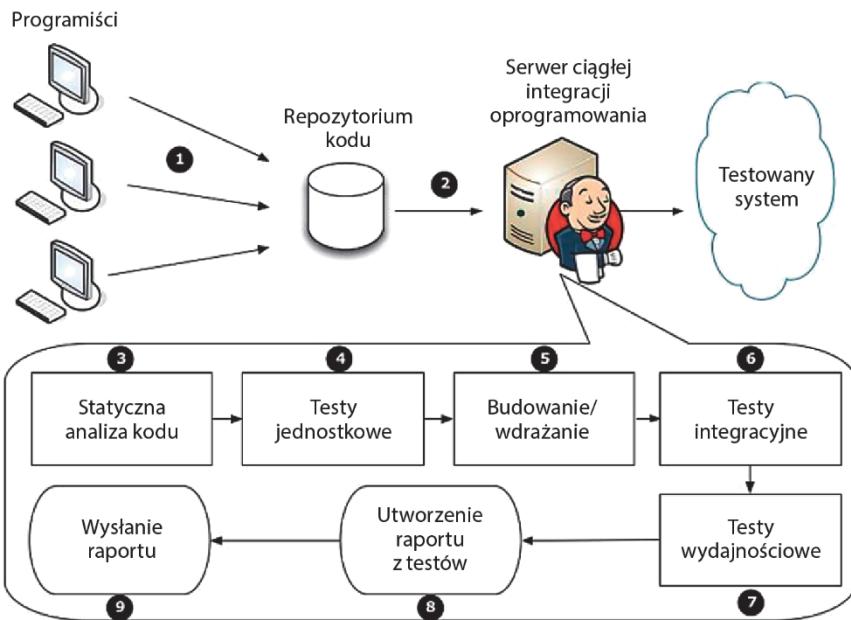
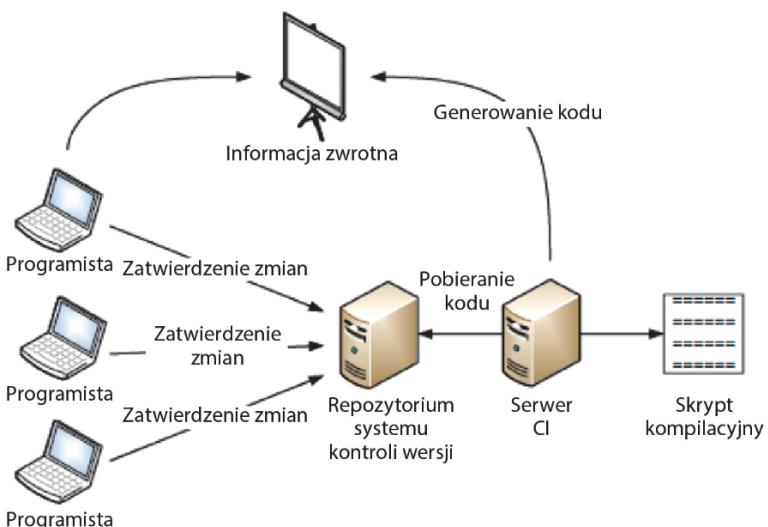


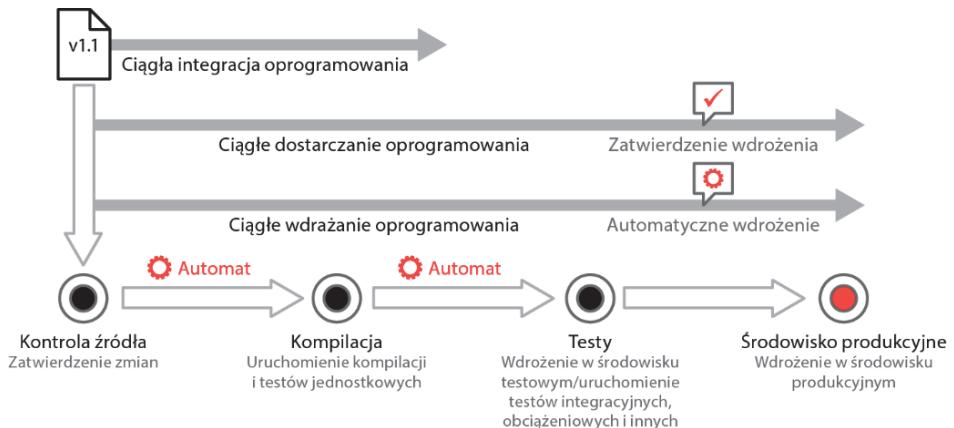
Bezpieczeństwo kontenerów w DevOps. Zabezpieczanie i monitorowanie kontenerów Docker



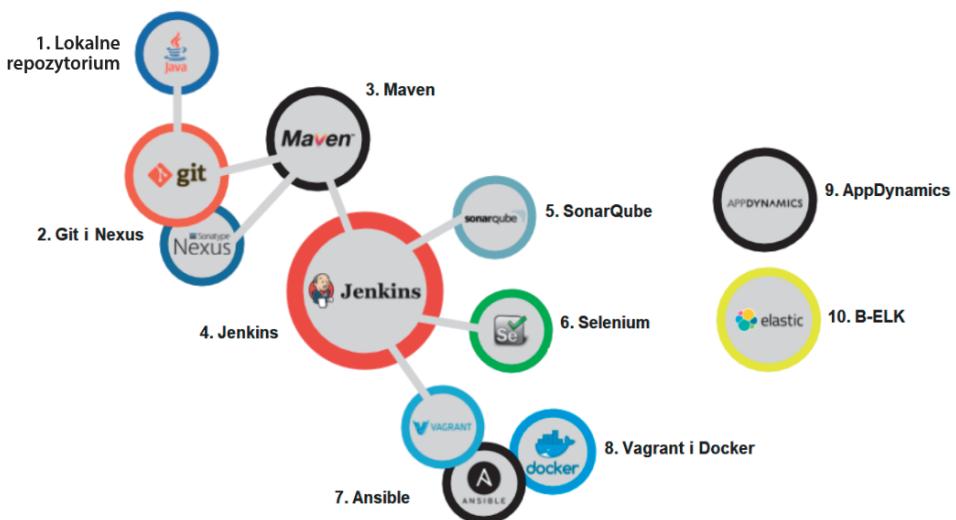
Rysunek 1.3. Proces ciągłego dostarczania oprogramowania



Rysunek 1.4. Proces ciągłej integracji oprogramowania



Rysunek 1.5. Kanał dostarczania oprogramowania



Rysunek 1.6. Narzędzia DevOps

Periodic Table of DevOps Tools (v3)																	
		Tool Categories								Tool Examples							
Category	Type	Open Source				Commercial				Deployment				Analytics			
		OS	Fr	Fr	Fr	Os	Fr	Fr	Fr	Ds	Cn	St	Ad	EI	Ni	Zb	Cx
1	Os	Gl	GitLab							Source Control Mgmt.	Deployment		Analytics				
2	En									Database Automation	Containers	Monitoring					
3	Fm	Gh	Github	Dt	Octalit					Continuous Integration	Release Orchestration	Security					
4	Fr									Testing	Cloud	Collaboration					
5	Fr									Configuration	AI Ops						
6	En	Sv	Subversion	Db	DbVisualStudio												
7	En	Cw	SPW	Dp	Depixis	Jn	Jenkins	Cs	Codename	Fn	FNRescue	Ka	Karma	Su	Tf	Xlr	En
8	En	At	Antifactory	Rg	Redgate	Ba	Bamboo	Vs	VSTS	Se	SmartBear JMeter	Jm	Juju	Ja	Sauce Labs	Dk	Aws
9	En	Nx	Nexus	Fw	Flyway	Tc	TeamCity	Tc	Getting	Tn	TestNG	Tt	Spock	Pe	Puppet	Ur	Az
10	En	Bb	BitBucket	Pf	Perforce HelixCore	Cb	Atlassian CodeBeautify	Cu	Cucumber	Mc	Mocha	Lo	Locust	Mf	Macro Focus UFT	Si	Gc
11	En																Op
12	En																Sl
13	En																Sp
14	En																Hv
15	En																Fr
16	En																Si
17	En																Fd
18	En																Os
19	En																Os
20	En																Os
21	En																Os
22	En																Os
23	En																Os
24	En																Os
25	En																Os
26	En																Os
27	En																Os
28	En																Os
29	En																Os
30	En																Os
31	En																Os
32	En																Os
33	En																Os
34	En																Os
35	En																Os
36	En																Os
37	En																Os
38	En																Os
39	En																Os
40	En																Os
41	En																Os
42	En																Os
43	En																Os
44	En																Os
45	En																Os
46	En																Os
47	En																Os
48	En																Os
49	En																Os
50	En																Os
51	En																Os
52	En																Os
53	En																Os
54	En																Os
55	En																Os
56	En																Os
57	En																Os
58	En																Os
59	En																Os
60	En																Os
61	En																Os
62	En																Os
63	En																Os
64	En																Os
65	En																Os
66	En																Os
67	En																Os
68	En																Os
69	En																Os
70	En																Os
71	En																Os
72	En																Os
73	En																Os
74	En																Os
75	En																Os
76	En																Os
77	En																Os
78	En																Os
79	En																Os
80	En																Os
81	En																Os
82	En																Os
83	En																Os
84	En																Os
85	En																Os
86	En																Os
87	En																Os
88	En																Os
89	En																Os
90	En																Os
91	En																Os
92	En																Os
93	En																Os
94	En																Os
95	En																Os
96	En																Os
97	En																Os
98	En																Os
99	En																Os
100	En																Os
101	En																Os
102	En																Os
103	En																Os
104	En																Os
105	En																Os
106	En																Os
107	En																Os
108	En																Os
109	En																Os
110	En																Os
111	En																Os
112	En																Os
113	En																Os
114	En																Os
115	En																Os
116	En																Os
117	En																Os
118	En																Os
119	En																Os
120	En																Os
121	En																Os
122	En																Os
123	En																Os
124	En																Os
125	En																Os
126	En																Os
127	En																Os
128	En																Os
129	En																Os
130	En																Os
131	En																Os
132	En																Os
133	En																Os
134	En																Os
135	En			</td													

Rysunek 1.7. Układ okresowy narzędzi DevOps



ANSIBLE

ANSIBLE

[Website](#) | [Wikipedia](#)

Ansible is an open-source software platform for configuring and managing computers. It combines multi-node software deployment, ad hoc task execution, and configuration management. It manages nodes over SSH or PowerShell and requires Python (2.4 or later) to

 **CHEF**

 **cobbler**

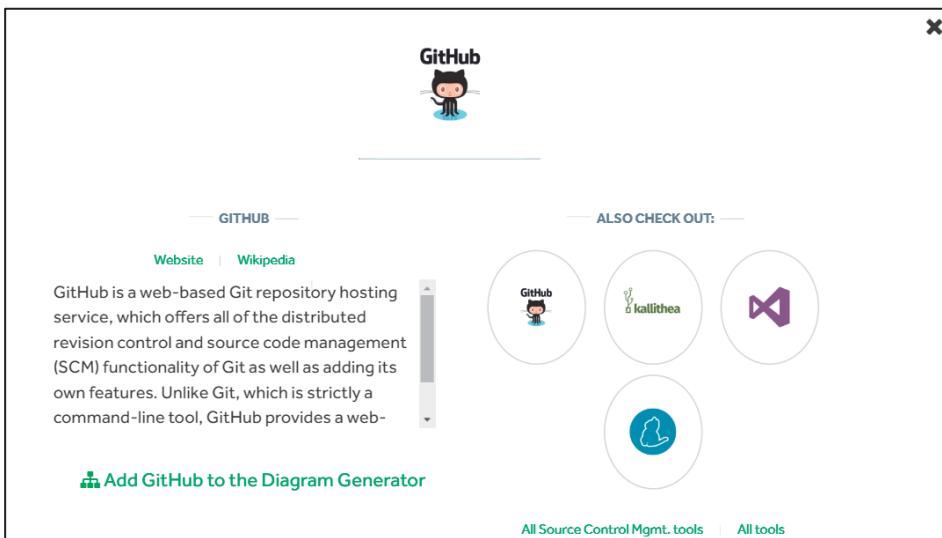
 **Quali**

 **congruit**

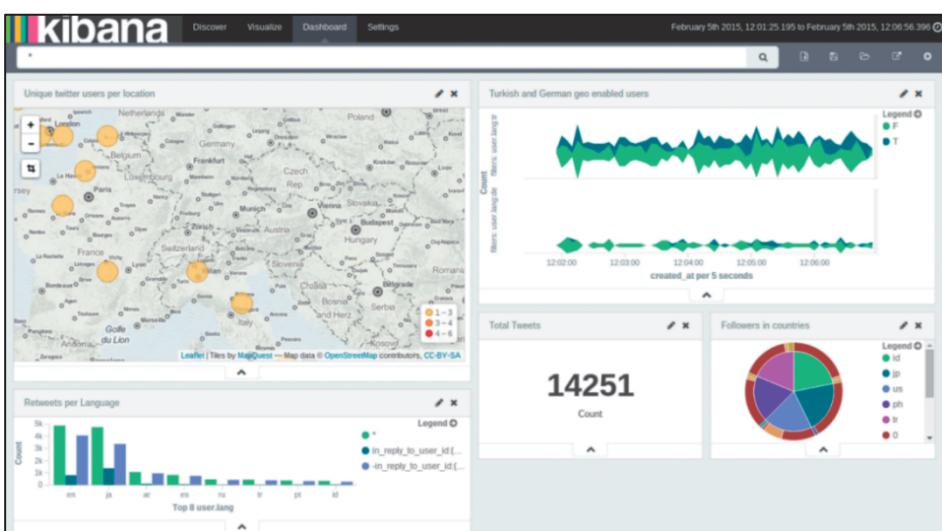
 [Add Ansible to the Diagram Generator](#)

All Configuration tools | All tools

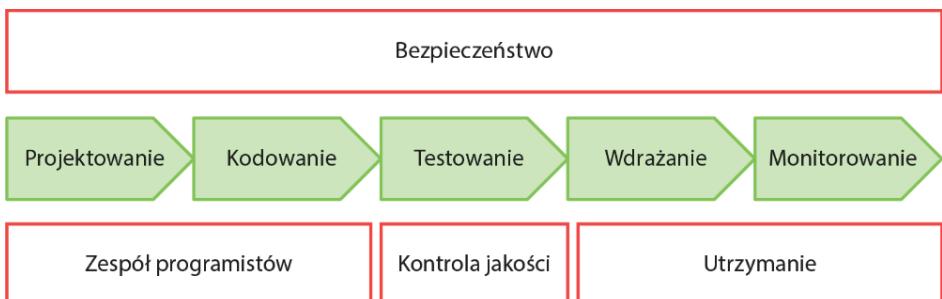
Rysunek 1.8. Opis narzędzia Ansible



Rysunek 1.9. Opis narzędzia GitHub



Rysunek 1.10. Narzędzie Kibana do analizy danych



Rysunek 1.11. Cykl DevSecOps

```

$ docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
5b7339215d1d: Pull complete
14ca88e9f672: Pull complete
a31c3b1caad4: Pull complete
b054a26005b7: Pull complete
Digest: sha256:9b1702dcfe32c873a770a32cf306dd7fc1c4fd134adfb783db68defc8894b3c
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
[node1] (local) root@192.168.0.23 ~
$ docker images
REPOSITORY          TAG           IMAGE ID      CREATED        SIZE
ubuntu              latest        4c108a37151f   3 weeks ago   64.2MB

```

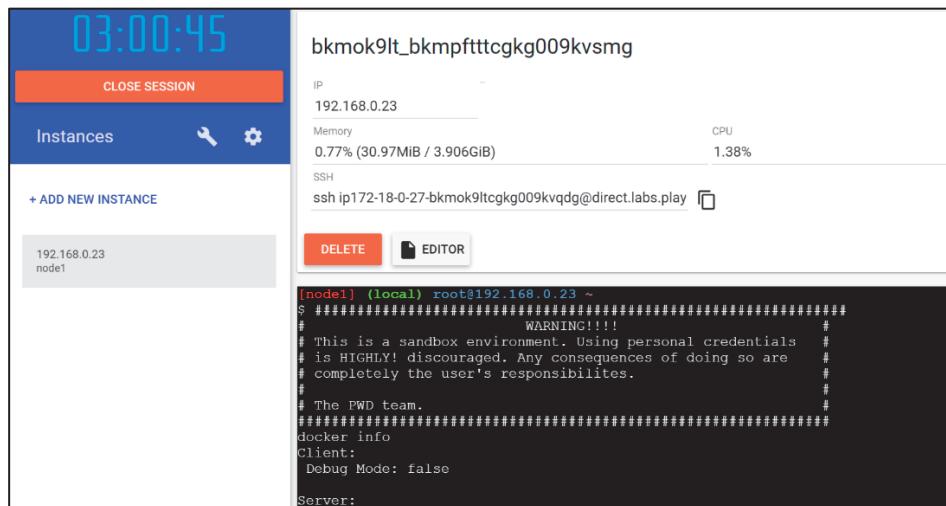
Rysunek 2.3. Przykład użycia polecenia docker pull

```

$ docker run -i -t ubuntu /bin/bash
root@760dca7304c6:/# ls -l
total 4
drwxr-xr-x    2 root root 4096 Jun 12 16:55 bin
drwxr-xr-x    2 root root     6 Apr 24  2018 boot
drwxr-xr-x    5 root root  360 Jul 16 09:10 dev
drwxr-xr-x    1 root root    66 Jul 16 09:10 etc
drwxr-xr-x    2 root root    6 Apr 24  2018 home
drwxr-xr-x    8 root root  96 May 23  2017 lib
drwxr-xr-x    2 root root   34 Jun 12 16:55 lib64
drwxr-xr-x    2 root root    6 Jun 12 16:54 media
drwxr-xr-x    2 root root    6 Jun 12 16:54 mnt
drwxr-xr-x    2 root root    6 Jun 12 16:54 opt
dr-xr-xr-x  1179 root root     0 Jul 16 09:10 proc
drwx-----    2 root root   37 Jun 12 16:55 root
drwxr-xr-x    1 root root  21 Jun 18 22:51 run
drwxr-xr-x    1 root root  21 Jun 18 22:51 sbin
drwxr-xr-x    2 root root    6 Jun 12 16:54 srv
dr-xr-xr-x   13 root root    0 Jul 15 01:26 sys
drwxrwxrwt    2 root root    6 Jun 12 16:55 tmp
drwxr-xr-x    1 root root   18 Jun 12 16:54 usr
drwxr-xr-x    1 root root   17 Jun 12 16:55 var

```

Rysunek 2.4. Wewnętrz kontenera



Rysunek 2.5. Uruchamianie kontenera Dockera w chmurze

Play with Docker classroom

About

First Alpine Linux Containers

Sep 19, 2017 • @jimmodified

In this lab you will run a popular, free, lightweight container and explore the basics of how containers work, how the Docker Engine executes and isolates containers from each other. If you already have experience running containers and basic Docker commands you can probably skip this intro exercise.

Concepts in this exercise:

- Docker engine
- Containers & images
- Image registries and Docker Store (AKA Docker Hub)
- Container isolation

Tips:

Code snippets are shown in one of three ways throughout this environment:

- Code that looks like **this** is sample code snippets that is usually part of an explanation.
- Code that appears in box like the one below can be clicked on and it will automatically be

If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing browser window.

```
#####
# This is a sandbox environment. Using personal credentials #
# is HIGHLY! discouraged. Any consequences of doing so are #
# completely the user's responsibilities. #
#
# The PWD team. #
#####
[node1] (local) root@192.168.0.43 ~
$ docker info
Client:
  Debug Mode: false

Server:
  Containers: 0
  Running: 0
  Paused: 0
  Stopped: 0
  Images: 0
  Server Version: 19.03.0-beta2
```

Your use of Play With Docker is subject to the Docker

f t

Rysunek 2.6. Serwis Play with Docker

```
$ docker node ls
ID          HOSTNAME   STATUS    AVAILABILITY
BILITY      MANAGER   VERSION
vz79cf4danb0njb3mg9p8vwpu *  node1     Ready     Active
          Leader       19.03.0-beta2
trmx4medo7ffaxy0e3z5xkhvt   node2     Ready     Active
          Leader       19.03.0-beta2
[node1] (local) root@192.168.0.37 ~
$ [node1] (local) root@192.168.0.37 ~
# completely the user's responsibilities.
#
# The PWD team.
#####
[node2] (local) root@192.168.0.38 ~
$ docker swarm join --token SWMTKNC-1-lu4oi5o8db483lipn37cyof8attas3h0xt2bf2y0fdf2my0id5-6xje7khf
2.168.0.37:2377
This node joined a swarm as a worker.
```

Rysunek 2.8. Dodanie węzła roboczego do klastra

```
$ docker service create --replicas 1 --name helloworld alpine ping www.google.com
4b5umtqhuhjxj88xv3yk91520
overall progress: 1 out of 1 tasks
1/1: running
verify: Service converged
[node1] (local) root@192.168.0.23 ~
$ docker service ls
ID          NAME      MODE      REPLICAS
GE          PORTS
4b5umtqhuhjx  helloworld replicated  1/1
ine:latest
```

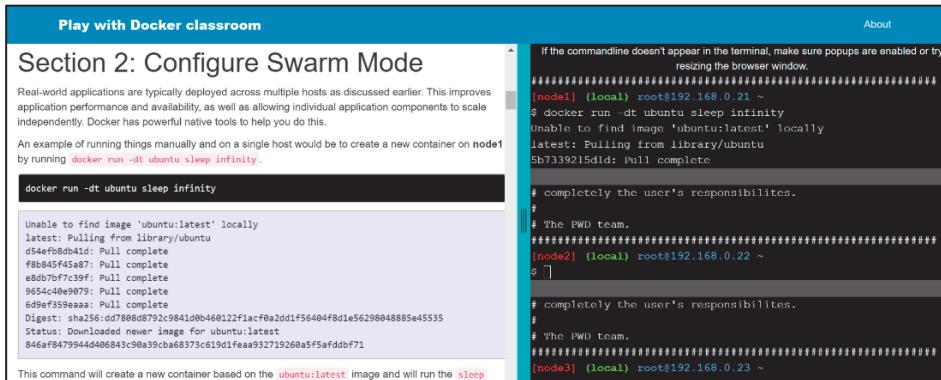
Rysunek 2.9. Tworzenie repliki usługi w klastrze Dockera

```

$ docker service ps helloworld
ID          NAME      IMAGE      NODE      DESIRED STATE
          CURRENT STATE      ERROR      PORTS
mf33ajplktg3    helloworld.1    alpine:latest  node2      Running
          Running about a minute ago
[node1] (local) root@192.168.0.23 ~
$ docker service scale helloworld=4
helloworld scaled to 4
[node1] (local) root@192.168.0.23 ~
$ docker service ps helloworld
ID          NAME      IMAGE      NODE      DESIRED STATE
          CURRENT STATE      ERROR      PORTS
mf33ajplktg3    helloworld.1    alpine:latest  node2      Running
          Running 2 minutes ago
tidzecyt0z8    helloworld.2    alpine:latest  node2      Running
          Running 7 seconds ago
fqmow6k5j2wa   helloworld.3    alpine:latest  node1      Running
          Running 7 seconds ago
3qx814c75jlv   helloworld.4    alpine:latest  node1      Running
          Running 7 seconds ago

```

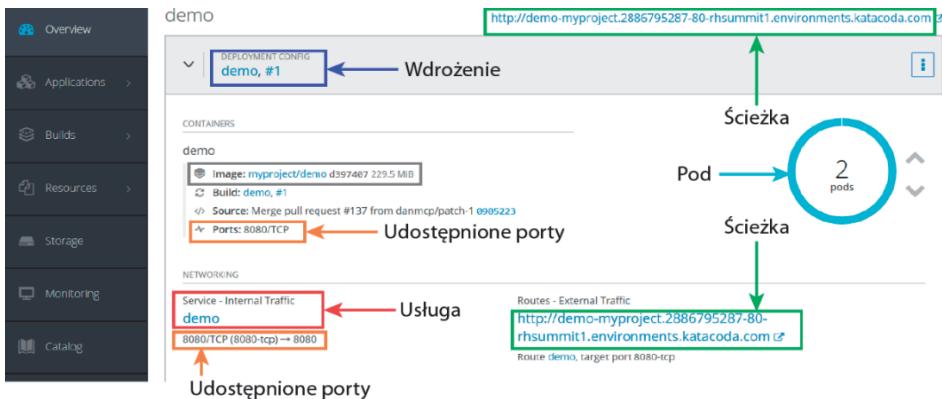
Rysunek 2.10. Zwiększenie liczby replik usługi w klastrze



Rysunek 2.11. Konfiguracja trybu Swarm



Rysunek 2.12. Wprowadzenie do trybu Swarm dla profesjalistów IT



Rysunek 2.13. Projekt OpenShift

Builds > demo

demo created 3 minutes ago

Start Build Actions

app demo

History Configuration Environment Events

Details

Build Strategy: Source https://github.com/openshift/django-ex.git

Source Repo: https://github.com/openshift/django-ex.git

Source Ref: master

Builder Image: openshift/python:3.6

Output To: myproject/demo:latest

Run Policy: Serial

Triggers Learn More

Config Change For: Build config demo

New Image For: openshift/python:3.6

Generic Webhook URL: https://2886795287-8443-frugo02.environment.katacoda.com

GitHub Webhook URL: https://2886795287-8443-frugo02.environment.katacoda.com

Manual (CLI): oc start-build demo -n myproject

Rysunek 2.14. Konfiguracja wdrożenia

Deployments > demo > #1

demo-1 created 7 minutes ago

Actions

app demo openshift.io/deployment-config.name demo

Details Environment Logs Events

Status: Active

Deployment Config: demo

Status Reason: config change

Selectors: deployment=demo-1 deploymentconfig=demo

Replicas: 2 current / 2 desired

Rysunek 2.15. Szczegóły konfiguracyjne wdrożenia

Containers

demo

- Image: myproject/demo e1d501d 229.5 MiB
- Build: demo, #1
- Source: Merge pull request #137 from dammcp/patch-1 [#905223](#) authored by Ben Parees
- Ports: 8080/TCP

Volumes

[Add Storage](#) | [Add Config Files](#)

Pods

Name	Status	Containers Ready	Container Restarts	Age
demo-1-qsqx6	Running	1/1	0	6 minutes
demo-1-7l45h	Running	1/1	0	8 minutes

Rysunek 2.16. Lista uruchomionych podów

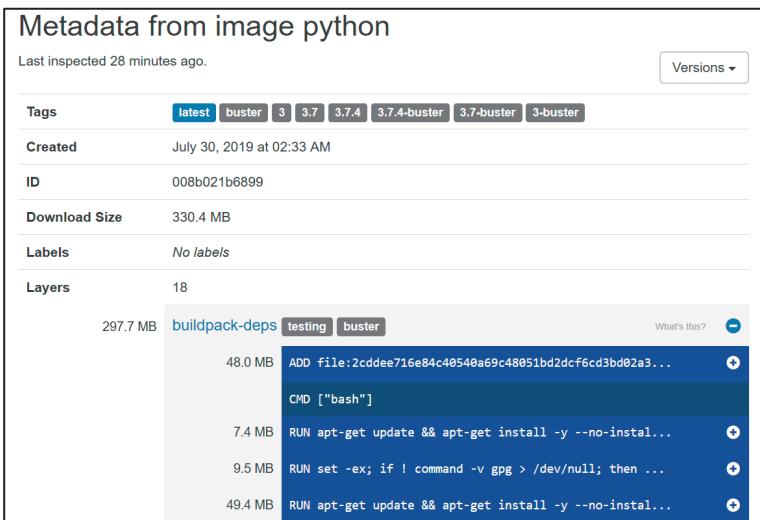
The screenshot shows the OpenShift Catalog interface. On the left, there's a sidebar with options like Overview, Applications, Builds, Resources, Storage, Monitoring, and Catalog. The Catalog option is selected. The main area has a heading "Select an item to add to the current project". Below it, there are tabs for All, Languages, Databases, Middleware, CI/CD, and Other. The Languages tab is selected. Under Languages, there are categories for All, Java, JavaScript, .NET, Perl, Ruby, PHP, and Python. Each category has a sub-section with a "Filter" dropdown set to "20 Items". Below these, specific templates are listed: .NET Core, .NET Core + PostgreSQL (Persistent), .NET Core Example, .NET Core Runtime Example, and CakePHP + MySQL.

Rysunek 2.17. Wbudowane szablony platformy OpenShift

The screenshot shows the OpenShift training course catalog. It features six course cards arranged in two rows of three. Each card has a dark background with white text and a red "START COURSE" button at the bottom. The courses are:

- Foundations of OpenShift
- Building Applications On OpenShift
- Subsystems, Components, and Internals
- OpenShift Playgrounds
- Service Mesh Workshop with Istio
- Building Operators on OpenShift

Rysunek 2.18. Scenariusze szkoleniowe platformy OpenShift



Rysunek 3.3. Strona usługi MicroBadger prezentująca metadane obrazu zawierającego środowisko Pythona

Supported tags and respective Dockerfile links

Simple Tags

- [3.8.0b2-buster](#), [3.8-rc-buster](#), [rc-buster](#)
- [3.8.0b2-slim-buster](#), [3.8-rc-slim-buster](#), [rc-slim-buster](#), [3.8.0b2-slim](#), [3.8-rc-slim](#), [rc-slim](#)
- [3.8.0b2-alpine3.10](#), [3.8-rc-alpine3.10](#), [rc-alpine3.10](#), [3.8.0b2-alpine](#), [3.8-rc-alpine](#), [rc-alpine](#)
- [3.8.0b2-windowsservercore-ltsc2016](#), [3.8-rc-windowsservercore-ltsc2016](#), [rc-windowsservercore-ltsc2016](#)
- [3.8.0b2-windowsservercore-1803](#), [3.8-rc-windowsservercore-1803](#), [rc-windowsservercore-1803](#)
- [3.8.0b2-windowsservercore-1809](#), [3.8-rc-windowsservercore-1809](#), [rc-windowsservercore-1809](#)
- [3.7.4-buster](#), [3.7-buster](#), [3-buster](#), [buster](#)
- [3.7.4-slim-buster](#), [3.7-slim-buster](#), [3-slim-buster](#), [slim-buster](#), [3.7.4-slim](#), [3.7-slim](#), [3-slim](#), [slim](#)

Rysunek 3.4. Etykiety oficjalnego obrazu zawierającego środowisko Pythona, umieszczonego w serwisie Docker Hub

```
$ docker image pull python:3.8-rc-alpine3.10
3.8-rc-alpine3.10: Pulling from library/python
050382585609: Pull complete
dac2222ca532: Pull complete
a5a8a13f5210: Pull complete
48ed6fe4c480: Pull complete
f5c21fef32f5: Pull complete
Digest: sha256:e686f6b5cf95f23bbb19d4f38a9f541abfdca0f7f6be6b74fbf862db068793be
Status: Downloaded newer image for python:3.8-rc-alpine3.10
docker.io/library/python:3.8-rc-alpine3.10
[node1] (local) root@192.168.0.43 ~
$ docker images
REPOSITORY          TAG           IMAGE ID      CREATED        SIZE
python              latest        14a2caeca327    17 hours ago   918MB
python              3.8-rc-alpine3.10  70dal2d86711    17 hours ago   109MB
```

Rysunek 3.5. Etykieta pobranego obrazu

```
$ docker build -t fedora_image .
Sending build context to Docker daemon 1.747MB
Step 1/4 : FROM fedora:latest
--> 2b74bf3d2430
Step 2/4 : MAINTAINER maintainer
--> Using cache
--> 471a4d43a7c5
Step 3/4 : RUN echo "This container was built on $(date)." > /tmp/built.txt
--> Using cache
--> 2c90489b36b6
Step 4/4 : ENTRYPOINT ["cat","/tmp/built.txt"]
--> Using cache
--> 8a83454955f9
Successfully built 8a83454955f9
Successfully tagged fedora_image:latest
[node1] (local) root@192.168.0.43 ~
$ docker run fedora_image
This container was built on Tue Jul 30 18:19:06 UTC 2019.
```

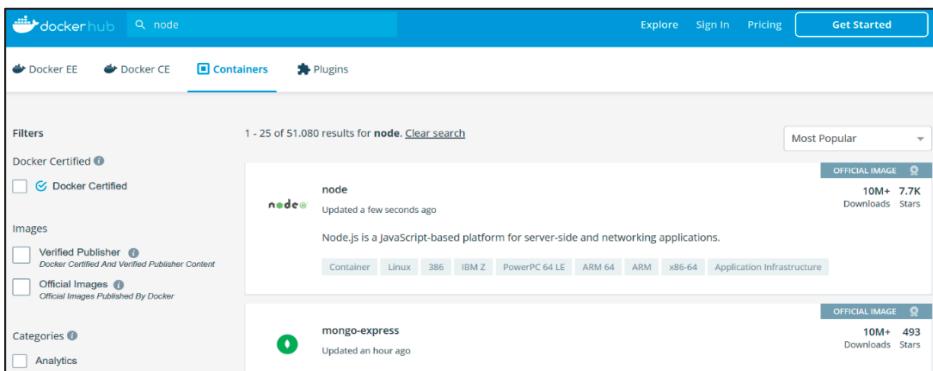
Rysunek 3.6. Przykład użycia poleceń docker build i docker run

```
$ docker images
REPOSITORY          TAG           IMAGE ID      CREATED        SIZE
test_dockerfile     latest        e22b016f25f7    3 seconds ago  176MB
ubuntu              latest        3556258649b2    3 days ago   64.2MB
[node1] (local) root@192.168.0.33 ~/ubuntutest
$ docker run -dti --name mycontainer e22b016f25f7
aca0408912343d042b3352146dd87e891179941498237e220a5d254175288625
[node1] (local) root@192.168.0.33 ~/ubuntutest
$ docker ps
CONTAINER ID        IMAGE               COMMAND            CREATED          STATUS          PORTS
NAMES
aca040891234        e22b016f25f7      "bash"             12 seconds ago  Up 10 sec
mycontainer
```

Rysunek 3.8. Polecenia docker images, docker run i docker ps

```
$ docker exec -i -t mycontainer /bin/bash
root@aca040891234:/# dstat
You did not select any stats, using -cdngy by default.
--total-cpu-usage-- -disk/total- -net/total- ---paging-- ---system--
usr sys idl wai stl| read wrt| recv send| in out| int csw
 31 17 52 0 0| 174k 2591k| 0 0 0| 0 0| 12k 47k
 28 27 45 0 0| 0 0| 0 0| 0 0| 27k 87k
 31 28 41 0 0| 0 0| 32k| 0 0 0| 0 0| 28k 100k
 28 25 47 0 0| 0 0| 4096B| 0 0 0| 0 0| 28k 89k
 30 30 39 0 0| 0 0| 396k| 0 0 0| 0 0| 28k 100k
 29 26 45 0 0| 0 0| 296k| 0 0 0| 0 0| 28k 91k
 31 29 41 0 0| 0 0| 4096B| 0 0 0| 0 0| 27k 99k
 28 26 45 0 0| 0 0| 48k| 0 0 0| 0 0| 27k 89k
```

Rysunek 3.9. Przykład użycia polecenia dstat wewnątrz kontenera



Rysunek 3.11. Wyszukiwanie obrazu w serwisie Docker Hub

```
$ docker pull gcr.io/distroless/python3
Using default tag: latest
latest: Pulling from distroless/python3
e8d8785a314f: Pull complete
e005d777a298: Pull complete
3e010093287c: Pull complete
609f69c3154c: Pull complete
Digest: sha256:b83bd4dc7c34d1c3a1b8400474163fccfe5b9110d0d1cb12b48e1786473d5ba2
Status: Downloaded newer image for gcr.io/distroless/python3:latest
gcr.io/distroless/python3:latest
[node1] (local) root@192.168.0.43 ~
$ docker pull python:alpine
alpine: Pulling from library/python
050382585609: Pull complete
dac2222ca532: Pull complete
29a7fe408caa: Pull complete
6ad337b9b53f: Pull complete
31d663a76478: Pull complete
Digest: sha256:d22196e0ced4a0fd44916e3ff4aea00565260f66a3d0d26f5551b8fdbd833423
Status: Downloaded newer image for python:alpine
docker.io/library/python:alpine
```

Rysunek 3.20. Pobranie pełnego obrazu oraz jego okrojonej wersji zawierającej środowisko Pythona i system operacyjny Alpine

```
1 FROM python:3-slim AS build-env
2 ADD . /app
3 WORKDIR /app
4
5 FROM gcr.io/distroless/python3
6 COPY --from=build-env /app /app
7 WORKDIR /app
```

Rysunek 3.22. Przykładowy plik Dockerfile wykorzystywany do tworzenia okrojonego obrazu zawierającego środowisko Pythona 3

```
$ docker images
REPOSITORY          TAG        IMAGE ID      CREATED       SIZE
python3              latest     f6d85470e9f6   21 minutes ago  52.5MB
<none>              <none>     801bdc56f2ed   23 minutes ago  52.5MB
python               3-slim    ca7f9e245002   2 weeks ago   143MB
python               latest     a4cc999cf2aa   2 weeks ago   929MB
gcr.io/distroless/python3  latest     b31fedb42763   49 years ago  50.9MB
```

```
$ docker run -it python bash
root@a4c6043a9fef:/# exit
exit
[node1] (local) root@192.168.0.8 ~
$ docker run -it python3 bash
/usr/bin/python3.5: can't open file 'bash': [Errno 2] No such file or directory
[node1] (local) root@192.168.0.8 ~
```

Rysunek 3.23. Uruchomienie okrojonego obrazu zawierającego środowisko Pythona 3

```
$ docker ps
CONTAINER ID        IMAGE           COMMAND       CREATED        STATUS
PORTS             NAMES
79b33d74c408      mysql           "docker-entrypoint.s..."   20 seconds ago   Up 19 seconds
3306/tcp, 33060/tcp  mysql
[node1] (local) root@192.168.0.38 ~
$ docker exec mysql touch /opt/filename
touch: cannot touch '/opt/filename': Read-only file system
```

Rysunek 4.4. Uruchomienie kontenera z bazą MySQL i woluminem

```
$ docker pull python@sha256:35ff9f44818f8850fld318aa69c2e7ba61d85e3b93283078c10e56e7d864c183
sha256:35ff9f44818f8850fld318aa69c2e7ba61d85e3b93283078c10e56e7d864c183: Pulling from library/python
c5e155d5ald1: Already exists
221d80d00ae9: Already exists
4250b3117dca: Already exists
3b7ca19181b2: Already exists
425d7b2a5bcc: Already exists
dc3049fcf3f44: Pull complete
472a6afc6332: Pull complete
6f79c90f8d7c: Pull complete
105lee813012: Pull complete
Digest: sha256:35ff9f44818f8850fld318aa69c2e7ba61d85e3b93283078c10e56e7d864c183
Status: Downloaded newer image for python@sha256:35ff9f44818f8850fld318aa69c2e7ba61d85e3b93283078c10e56e7d864c183
docker.io/library/python@sha256:35ff9f44818f8850fld318aa69c2e7ba61d85e3b93283078c10e56e7d864c183
[node1] (local) root@192.168.0.28 ~
$ docker pull python@sha256:35ff9f44818f8850fld318aa69c2e7ba61d85e3b93283078c10e56e7d864c183
invalid checksum digest length
```

Rysunek 4.19. Pobranie obrazu Docker'a zawierającego środowisko Pythona i sprawdzenie jego sumy kontrolnej

```
$ export DOCKER_CONTENT_TRUST=1
[node1] (local) root@192.168.0.28 ~
$ docker pull jmortegac/linux_tweet_app:1.0
Error: remote trust data does not exist for docker.io/jmortegac/linux_tweet_app: notary.docker.io does not have trust data for docker.io/jmortegac/linux_tweet_app
[node1] (local) root@192.168.0.28 ~
$ export DOCKER_CONTENT_TRUST=0
[node1] (local) root@192.168.0.28 ~
$ docker pull jmortegac/linux_tweet_app:1.0
1.0: Pulling from jmortegac/linux_tweet_app
bc95e04b23c0: Pull complete
a21d9ee25fc3: Pull complete
9bda7d5af39: Pull complete
e64d34b6ad71: Pull complete
a7e018a2b8ff: Pull complete
Digest: sha256:db9f2e75b91780c804c283081123fbelb2b4080fel24eeb040f0ad1bfd70fe30
Status: Downloaded newer image for jmortegac/linux_tweet_app:1.0
docker.io/jmortegac/linux_tweet_app:1.0
```

Rysunek 4.20. Pobieranie obrazu przy włączonej i wyłączonej funkcjonalności DCT

```

ENV GPG_KEY E3FF2839C048B25C084DEBE9B26995E310250568
ENV PYTHON_VERSION 3.8.0a4

RUN set -ex \
    \
    && wget -O python.tar.xz "https://www.python.org/ftp/python/${PYTHON_VERSION%*[a-z]}/Python-$PYTHON_VERSION.tar.xz" \
    && wget -O python.tar.xz.asc "https://www.python.org/ftp/python/${PYTHON_VERSION%*[a-z]}/Python-$PYTHON_VERSION.tar.xz.asc" \
    && export GNUPGHOME="$(mktemp -d)" \
    && gpg --batch --keyserver ha.pool.sks-keyservers.net --recv-keys "$GPG_KEY" \
    && gpg --batch --verify python.tar.xz.asc python.tar.xz \
    && { command -v gpgconf > /dev/null && gpgconf --kill all || :; } \
    && rm -rf "$GNUPGHOME" python.tar.xz.asc \
    && mkdir -p /usr/src/python \
    && tar -xJC /usr/src/python --strip-components=1 -f python.tar.xz \
    && rm python.tar.xz \
    \

```

Rysunek 4.21. Bezpieczne pobieranie obrazu z wykorzystaniem pliku Dockerfile

```

$ docker run -d -p 5000:5000 --restart=always --name registry registry:2
Unable to find image 'registry:2' locally
2: Pulling from library/registry
c87736221ed0: Pull complete
lcc8e0b4b4df: Pull complete
54d33bcb37f5: Pull complete
e8afc091c171: Pull complete
b4541f6d3db6: Pull complete
Digest: sha256:77a8fb00c00b99568772a70f0863f6192ff2635e4af4e22e4d9c622edeb5f2de
Status: Downloaded newer image for registry:2
d47e0b90b502870403e3f634632ffd4012c792b02aba11f4264cc5eed56c7089
(node) (local) root@192.168.0.28 ~
$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS
NAMES
d47e0b90b502        registry:2          "/entrypoint.sh /etc.."   2 minutes ago      Up 2 minutes       0.0.0.0
0:5000->5000/tcp   registry

```

Rysunek 4.23. Pobranie kontenera w celu utworzenia lokalnego rejestrów

```

audit.rules
/etc/audit/rules.d

## Rules
-W /usr/bin/docker -k docker
-W /var/lib/docker -k docker
-W /etc/docker -k docker
-W /usr/lib/systemd/system/docker.service -k docker
-W /usr/lib/systemd/system/docker.socket -k docker
-W /etc/default/docker -k docker
-W /etc/docker/daemon.json -k docker
-W /usr/bin/docker-containerd -k docker
-W /usr/bin/docker-runc -k docker

```

Rysunek 5.4. Definiowanie reguł audytu

» Ubuntu » Packages » bionic (18.04LTS) » admin » apparmor-profiles

[Source: apparmor] [xenial] [xenial-updates] [bionic] [bionic-updates] [cosmic] [disco] [disco-updates] [eonan]

Package: apparmor-profiles (2.12-4ubuntu5.1) [security]

experimental profiles for AppArmor security policies

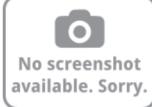
Other Packages Related to apparmor-profiles

depends	recommends	suggests	enhances
● apparmor (>= 2.8.96-2535-0ubuntu1~)			
user-space parser utility for AppArmor			

Download apparmor-profiles

Architecture	Package Size	Installed Size	Files
all	31.1 kB	360.0 kB	[list of files]

Links for apparmor-profiles



No screenshot available. Sorry.

Ubuntu Resources:

- [Bug Reports](#)

Rysunek 5.6. Pakiet modułu AppArmor dla systemu Ubuntu

Docker Bench for Security

```
# -----
# Docker Bench for Security v1.3.3
# Docker, Inc. (c) 2015-
# Checks for dozens of common best-practices around deploying Docker containers in production.
# Inspired by the CIS Docker Community Edition Benchmark v1.1.0.
# -----
Initialization Fri Jul 14 09:18:42 UTC 2017

[INFO] 1 - Host Configuration
[WARN] 1.1 - Ensure a separate partition for containers has been created
[NOTE] 1.2 - Ensure the container host has been Hardened
[PASS] 1.3 - Ensure Docker is up to date
[INFO] 1.4 - Check with the operating system vendor for support and security maintenance for Docker
[INFO] 1.4 - Ensure only trusted users are allowed to control Docker daemon
[INFO] 1.4 - docker:ix:992:vagrant
[WARN] 1.5 - Ensure auditing is configured for the Docker daemon
[WARN] 1.6 - Ensure auditing is configured for Docker files and directories - /var/lib/docker
[WARN] 1.7 - Ensure auditing is configured for Docker files and directories - /etc/docker
[WARN] 1.8 - Ensure auditing is configured for Docker files and directories - docker.service
[INFO] 1.9 - Ensure auditing is configured for Docker files and directories - docker.socket
[INFO] 1.9 - * File not found
[INFO] 1.10 - Ensure auditing is configured for Docker files and directories - /etc/default/docker
[INFO] 1.10 - * File not found
[INFO] 1.11 - Ensure auditing is configured for Docker files and directories - /etc/docker/daemon.json
```

Docker Pull Command

```
docker pull docker/docker-bench-security
```

Owner

 docker

Source Repository

 [GitHub](#)
[docker/docker-bench-security](#)

Rysunek 5.10. Obraz Docker Bench for Security w serwisie Docker Hub

Branch: master ▾ [docker-bench-security / tests /](#)

Create new file Upload files Find file History

konstruktoid use only year and month for version check #309 ... ✓ Latest commit 326e31f on 13 Apr

..

File	Description	Time
1_host_configuration.sh	use mountpoint and DockerRootDir #332	10 months ago
2_docker_daemon_configuration.sh	use only year and month for version check #309	4 months ago
3_docker_daemon_configuration_files.sh	linting	10 months ago
4_container_images.sh	revert grep thought fail	5 months ago
5_container_runtime.sh	linting	10 months ago
6_docker_security_operations.sh	Improve docker-bench-security json output	10 months ago
7_docker_swarm_configuration.sh	fix test 7.4 using 5.25 as a model	9 months ago
99_community_checks.sh	Improve docker-bench-security json output	10 months ago

Rysunek 5.17. Repozytorium GitHub

```

desc_5_1="Ensure AppArmor Profile is Enabled"
check_5_1="$id_5_1 - $desc_5_1"
starttestjson "$id_5_1" "$desc_5_1"

totalChecks=$((totalChecks + 1))

fail=0
no_apparmor_containers=""
for c in $containers; do
    policy=$(docker inspect --format 'AppArmorProfile={{ .AppArmorProfile }}' '$c')

    if [ "$policy" = "AppArmorProfile=" ] || [ "$policy" = "AppArmorProfile=[]" ] || [ "$policy" = "AppArmorProfile=<no value>" ]; then
        # If it's the first container, fail the test
        if [ $fail -eq 0 ]; then
            warn "$check_5_1"
            warn "* No AppArmorProfile Found: $c"
            no_apparmor_containers="$no_apparmor_containers $c"
            fail=1
        else
            warn "* No AppArmorProfile Found: $c"
            no_apparmor_containers="$no_apparmor_containers $c"
        fi
    fi

```

Rysunek 5.18. Skrypt 5_container_runtime.sh sprawdzający profil AppArmor

```

desc_5_2="Ensure SELinux security options are set, if applicable"
check_5_2="$id_5_2 - $desc_5_2"
starttestjson "$id_5_2" "$desc_5_2"

totalChecks=$((totalChecks + 1))

fail=0
no_securityoptions_containers=""
for c in $containers; do
    policy=$(docker inspect --format 'SecurityOpt={{ .HostConfig.SecurityOpt }}' '$c')

    if [ "$policy" = "SecurityOpt=" ] || [ "$policy" = "SecurityOpt=[]" ] || [ "$policy" = "SecurityOpt=<no value>" ]; then
        # If it's the first container, fail the test
        if [ $fail -eq 0 ]; then
            warn "$check_5_2"
            warn "* No SecurityOptions Found: $c"
            no_securityoptions_containers="$no_securityoptions_containers $c"
            fail=1
        else
            warn "* No SecurityOptions Found: $c"
            no_securityoptions_containers="$no_securityoptions_containers $c"
        fi
    fi

```

Rysunek 5.19. Filtrowanie SecurityOpt w skrypcie 5_container_runtime.sh

```

for c in $containers; do
    user=$(docker inspect --format 'User={{ .Config.User }}' '$c')

    if [ "$user" = "User=@0" ] || [ "$user" = "User=root" ] || [ "$user" = "User=" ] || [ "$user" = "User=[]" ] || [ "$user" = "User=<no v" ]
        # If it's the first container, fail the test
        if [ $fail -eq 0 ]; then
            warn "$check_4_1"
            warn "* Running as root: $c"
            root_containers="$root_containers $c"
            fail=1
        else
            warn "* Running as root: $c"
            root_containers="$root_containers $c"
        fi
    fi
done

```

Rysunek 5.20. Skrypt 4_container_images.sh sprawdzający konto wykorzystywane we wnętrzu kontenera

```

desc_2_6="Ensure TLS authentication for Docker daemon is configured"
check_2_6="$id_2_6 - $desc_2_6"
starttestjson "$id_2_6" "$desc_2_6"

totalChecks=$((totalchecks + 1))
if [ grep -i 'tcp://'$CONFIG_FILE' 2>/dev/null 1>&2 ] || \
[ $(get_docker_cumulative_command_line_args '-H' | grep -v '(unix|fd)://') >/dev/null 2>&1 ]; then
  if [ $(get_docker_configuration_file_args '"tlsverify":' | grep 'true') ] || \
  [ $(get_docker_cumulative_command_line_args '--tlsverify' | grep 'tlsverify') >/dev/null 2>&1 ]; then
    pass "$check_2_6"
    resulttestjson "PASS"
    currentScore=$((currentScore + 1))
  elif [ $(get_docker_configuration_file_args 'tls:' | grep 'true') ] || \
  [ $(get_docker_cumulative_command_line_args '--tls' | grep 'tls$') >/dev/null 2>&1 ]; then
    warn "$check_2_6"
    warn "      * Docker daemon currently listening on TCP with TLS, but no verification"
    resulttestjson "WARN" "Docker daemon currently listening on TCP with TLS, but no verification"
    currentScore=$((currentScore - 1))
  fi
fi

```

Rysunek 5.21. Fragment skryptu sprawdzajacy szyfrowanie TLS

```

FIND=$(grep "FROM" ${AUDIT_FILE} | sed 's/ /:space:/g')
for I in ${FIND}; do
  IMAGE=$(echo ${I} | sed 's/:space:/ /g' | awk '{ if ($1=="FROM") { print $2 } }')
  TAG=$(echo ${IMAGE} | cut -d':' -f2)
  Display --indent 2 --text "Found image:" --result "${IMAGE}"

  IS_DEBIAN=$(echo ${IMAGE} | grep -i debian)
  IS_FEDORA=$(echo ${IMAGE} | grep -i fedora)
  IS_UBUNTU=$(echo ${IMAGE} | grep -i ubuntu)
  IS_ALPINE=$(echo ${IMAGE} | grep -i alpine)
  IS_LATEST=$(echo ${TAG} | grep -i latest)

  if [ -n "${IS_DEBIAN}" ]; then IMAGE="debian"; fi
  if [ -n "${IS_FEDORA}" ]; then IMAGE="fedora"; fi
  if [ -n "${IS_UBUNTU}" ]; then IMAGE="ubuntu"; fi
  if [ -n "${IS_ALPINE}" ]; then IMAGE="alpine"; fi
  if [ -n "${IS_LATEST}" ]; then
    ReportWarning "dockerfile" "latest TAG used. Specifying a targeted OS image and version is better for reproducible results."
  fi

```

Rysunek 5.29. Kod uzyskujacy typ obrazu

```

if [ ${FILE_DOWNLOAD} -eq 1 ]; then
  SSL_USED_FIND=$(egrep "(https)" ${AUDIT_FILE})

  if HasData "${SSL_USED_FIND}"; then
    SSL_USED="YES"
    COLOR="GREEN"
  else
    SSL_USED="NO"
    COLOR="RED"
  fi
  Display --indent 2 --text "Integrity testing performed" --result "${SSL_USED}" --color ${COLOR}
  HASHING_USED=$(egrep "(sha1sum|sha256sum|sha512sum)" ${AUDIT_FILE})
  Display --indent 2 --text "Hashing" --result "${HASHING_USED}"
  KEYS_USED=$(egrep "(apt-key adv)" ${AUDIT_FILE}| sed 's/RUN apt-key adv//g' | sed 's/-keyserver/Key Server:/g' | sed 's/-recv/Key'
  Display --indent 2 --text "Signing keys used" --result "${KEYS_USED}"
  Display --indent 2 --text "All downloads properly checked" --result "?"
else
  Display --indent 2 --text "No files seems to be downloaded in this Dockerfile"
fi

```

Rysunek 5.30. Sprawdzenie pobranych pakietow

```
[root@dockerlab001 ~]# more myreport.txt
Docksan Report

-[ Medium ]-
=Docker running with IPv4 forwarding enabled=
Description:
Docker daemon reports it is running daemon with IPv4 forwarding enabled.
This is not recommended for production as it forwards network packets without rules.
Output:
Docker daemon reports it is running with automatic IPv4 forwarding.
Solution:
It is recommended to disable IPv4 forwarding by default.

-[ Low ]-
=Container have higher number of changed files-
Description:
Container have high number of changed files which is not recommended practice.
This is not recommended for production as data can be lost. It can also mean successful break in attempt.
Output:
4792d6df02c3124e9602e5992a2d546e1e80aab08364184440a389893703505e (/traefik_proxy_1) with IP: has more than 5 file changes: 9
/etc
/etc/traefik
/etc/traefik/traefik.toml
/run
/run/secrets
/tmp
/var
/var/run
/var/run/docker.sock

Solution:
It is recommended to have minimal number of changed files inside container and do not store data inside container. It is recommended to use volumes.

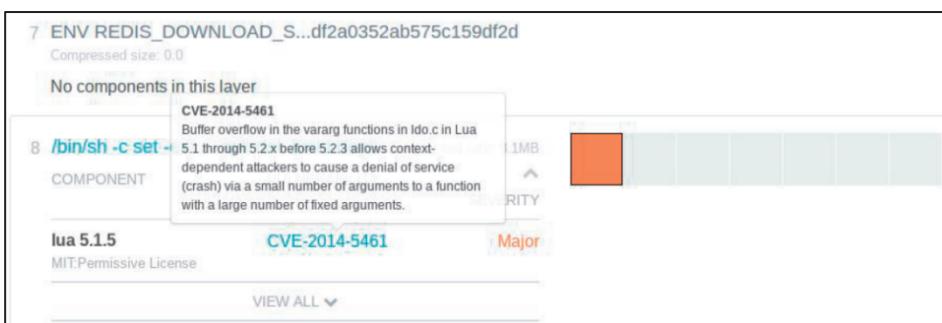
=Docker registries are not mirrored=
Description:
Docker daemon reports it is running configuration without registry mirrors.
If you set up local mirror, your docker host does not have to go directly to internet if not needed.
Output:
Docker daemon reports it does not have mirror registries.
Offending registry indexes:
docker.io

Solution:
it is recommended to setup mirror registry.
```

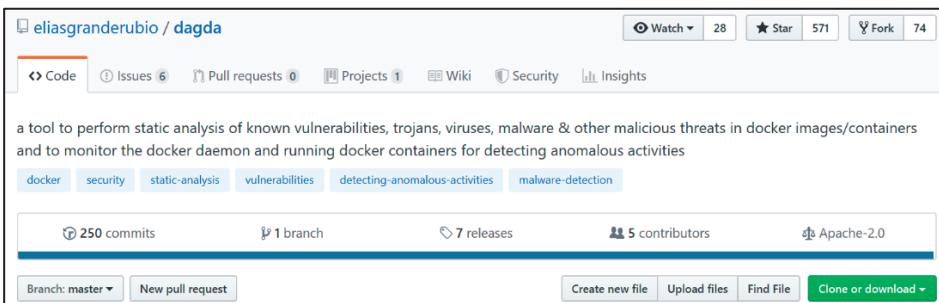
Rysunek 5.33. Raport opisujący luki w bezpieczeństwie hosta platformy Docker



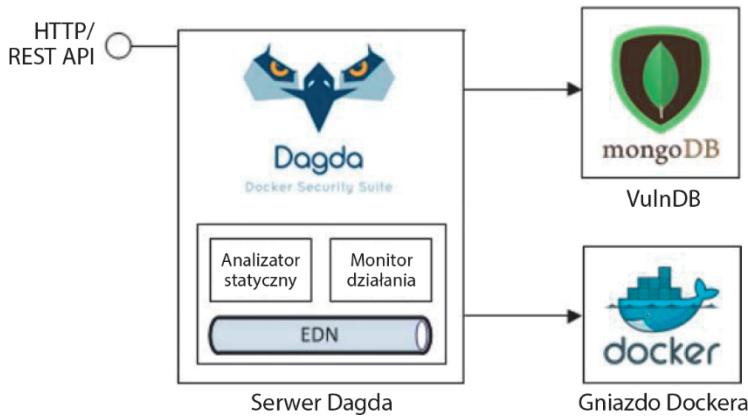
Rysunek 6.1. Analiza warstw obrazu Dockera



Rysunek 6.2. Szczegółowe informacje na temat zagrożenia CVE obrazu Dockera



Rysunek 6.3. Narzędzie Dagda w repozytorium GitHub



Rysunek 6.4. Architektura narzędzia Dagda

Aqua Security's MicroScanner lets you check your container images for vulnerabilities. If your image has any known high-severity issue, MicroScanner can fail the image build, making it easy to include as a step in your CI/CD pipeline.

Registering for a token

Rysunek 6.15. Strona narzędzia MicroScanner, na której można uzyskać token

Data Source	Data Collected	Format	License
Debian Security Bug Tracker	Debian 6, 7, 8, unstable namespaces	dpkg	Debian
Ubuntu CVE Tracker	Ubuntu 12.04, 12.10, 13.04, 14.04, 14.10, 15.04, 15.10, 16.04 namespaces	dpkg	GPLv2
Red Hat Security Data	CentOS 5, 6, 7 namespaces	rpm	CVRF
Oracle Linux Security Data	Oracle Linux 5, 6, 7 namespaces	rpm	CVRF
Alpine SecDB	Alpine 3.3, Alpine 3.4, Alpine 3.5 namespaces	apk	MIT
NIST NVD	Generic Vulnerability Metadata	N/A	Public Domain

Rysunek 6.17. Lista dystrybucji systemu Linux obsługiwanych przez narzędzie Clair

clair_config	clair	2 years ago
Dockerfile	clair	2 years ago
README.MD	clair	2 years ago
docker-compose.yml	clair	2 years ago
README.MD		

Rysunek 6.18. Repozytorium GitHub zawierające skaner Clair

```
1 version: '2'
2 services:
3   postgres:
4     container_name: clair_postgres
5     image: postgres:latest
6     environment:
7       POSTGRES_PASSWORD: password
8
9   clair:
10    container_name: clair_clair
11    image: hxquangnhat/clair:latest
12    depends_on:
13      - postgres
14    ports:
15      - "6060-6061:6060-6061"
16    links:
17      - postgres
18    volumes:
19      - /tmp:/tmp
20      - ./clair_config:/config
21      - /var/run/docker.sock:/var/run/docker.sock
22    command: [-config, /config/config.yaml]
```

Rysunek 6.19. Plik docker-compose.yml skanera Clair

```

dockerserverclair@DockerServerClair:~$ sudo mkdir $HOME/clair_config
dockerserverclair@DockerServerClair:~$ sudo curl -L https://raw.githubusercontent.com/coreos/clair/master/config.yaml
sample -o $PWD/clair_config/config.yaml
  % Total    % Received % Xferd  Average Speed   Time     Time      Current
     0          0        0      0   17300      0  --:--:--  --:--:--  --:--:-- 17402
100  2941  100  2941     0      0  17300      0  --:--:--  --:--:--  --:--:-- 17402
dockerserverclair@DockerServerClair:~$ ls
clair_config
dockerserverclair@DockerServerClair:~$ 

```

Rysunek 6.23. Drugi krok instalacji

```

dockerserverclair@DockerServerClair:~/clairscanner$ sudo mv clair-scanner_linux_amd64 clair-scanner
dockerserverclair@DockerServerClair:~/clairscanner$ ls
clair-scanner
dockerserverclair@DockerServerClair:~/clairscanner$ sudo chmod +x clair-scanner
dockerserverclair@DockerServerClair:~/clairscanner$ ls
clair-scanner
dockerserverclair@DockerServerClair:~/clairscanner$ 

```

Rysunek 6.26. Nadanie skanerowi Clair uprawnień do działania

```

2018/11/25 16:28:21 [INFO] □ Analyzing dde0e59ffbf1df7fc5f6d4a60dcaa2a7d936eab1fe455a9f8dd96b64b706b37
2018/11/25 16:28:24 [INFO] □ Analyzing ffcc629ef32caddb37507405b90fe425e1bdaf77d087832ea8e3feee5d079516e
2018/11/25 16:28:24 [INFO] □ Analyzing e90cc14beb2b09c943dd6b4a55fb3c2c14fb06aec3bc5308f24f87f2faf45a4b
2018/11/25 16:28:24 [INFO] □ Analyzing f7e0e4d4e102e919c5e29ee3402d36c87368888ec16a16398957d2674019c3ec
2018/11/25 16:28:24 [INFO] □ Analyzing e1c827f65ff4ff62f553edfd8749a594c5ib92f13485f45ddc37d7c348e144
2018/11/25 16:28:24 [INFO] □ Analyzing 512148149fabf6ca5ca1bcfcfeefc3c8bd2b3076746d5b74719011a208eb796b
2018/11/25 16:28:24 [INFO] □ Analyzing 70d3771c19ba4c1b4ec59092a35eeccc79187793b5680e4b3e4667579e5cfcd6
2018/11/25 16:28:24 [WARNING] □ Image [vulnerables/cve-2016-10033] contains 233 total vulnerabilities
2018/11/25 16:28:24 [ERROR] □ Image [vulnerables/cve-2016-10033] contains 233 unapproved vulnerabilities
+-----+
| STATUS | CVE SEVERITY | PACKAGE NAME | PACKAGE VERSION | CVE DESCRIPTION
+-----+
| Unapproved | High CVE-2016-2779 | util-linux | 2.25.2-6 | runuser in util-linux allows local users to escape to the parent session via a crafted TIOCSTI ioctl call, to the terminal's input buffer.
|           |           |           |           |
+-----+

```

Rysunek 6.28. Szósty krok instalacji

```

|           |           |           |           | local users to exploit side-channel timing attack on 'port contention'. | via a side-channel timing attack on 'port contention'. | https://security-tracker.debian.org/tracker/CVE-2018-5407 | 
+-----+
| Unapproved | Unknown CVE-2018-14647 | python2.7 | 2.7.9-2+deb8u1 | Python's elementtree C acceleration failed to initialise | Expat's hash salt during initialization. This could make it easy to conduct denial-of-service attacks against XML document that would cause pathological hash collisions in Expat's internal data structures, consuming large amounts CPU and RAM. Python 2.7 are believed to be vulnerable. | https://security-tracker.debian.org/tracker/CVE-2018-14647 | 
+-----+
| Unapproved | Unknown CVE-2018-19518 | php5 | 5.6.30+dfsg-0+deb8u1 | https://security-tracker.debian.org/tracker/CVE-2018-19518 | 
+-----+

```

Rysunek 6.29. Znalezione zagrożenia obrazu vulnerables/cve-2016-10033

Create a new container

First we'll create a container with a single new file based off of the `ubuntu` base image:

```
$ docker run ubuntu echo "fun" > myfile
```

The container will immediately terminate (because its one command is `echo`), so we'll use `docker ps -l` to list it:

CONTAINER ID	IMAGE	COMMAND	CREATED
0f72065197ef	ubuntu:12.04	echo fun	31 seconds ago

Make note of the `container id`; we'll need it for the commit command.

Rysunek 6.32. Identyfikator zwrócony przez nowo utworzony kontener

Tag the container to an image

We next need to tag the container to a known image name

Note that the `username` must be your Quay.io username and `reponame` is the new name of your repository.

```
$ docker commit 0f72065197ef quay.io/username/reponame  
e7050e5a288f9f3498cccd2847fee966d701867bc671b02abf03a6629dc921bb
```

Rysunek 6.33. Oznaczenie kontenera nazwą obrazu

Repository Tags

TAG	LAST MODIFIED	SECURITY SCAN	SIZE	EXPIRES	MANIFEST
latest	5 minutes ago	Unsupported	20.4 MB	Never	SHA256 0fe333c46ff4

1 - 1 of 1 Filter Tags... Compact Expanded

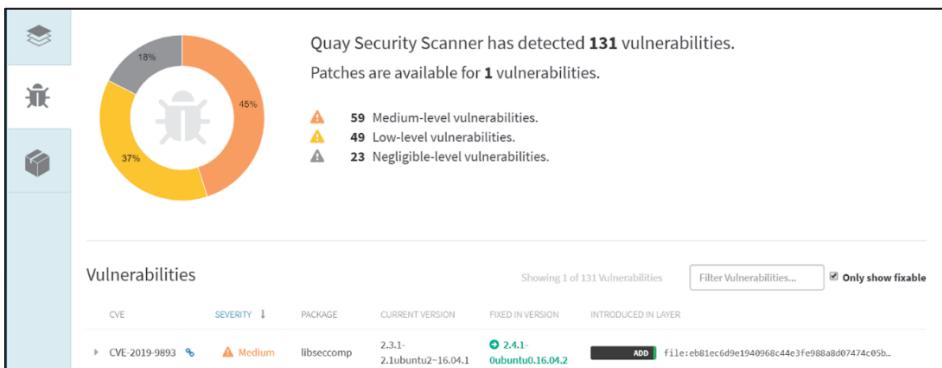
+ Add New Tag
Edit Labels
Delete Tag
Change Expiration

Rysunek 6.36. Operacje na oznaczeniach obrazu

Manifest Layers

```
>_ /bin/bash
CMD ["python3"]
RUN set -ex; wget -O get-pip.py 'https://bootstrap.pypa.io/get-pip.py'; python get-pip.py --disable-pip-version-check --no-cache-dir "pip=$PYTHON_PIP_VERSION"; pip --version; find /usr/local -depth 1 -type d -a ! -name test -o -name tests -o ! -type f -a ! -name *.pyc -o -name *.pyo -exec rm -rf '{}'; rm -f get-pip.py
ENV PYTHON_PIP_VERSION=9.0.1
RUN cd /usr/local/bin && ln -s idle3 idle && ln -s pydoc3 pydoc && ln -s python3 python && ln -s python3-config python-config
RUN set -ex && buildDeps=' dpkg-dev tcl-dev tk-dev ' && apt-get update && apt-get install -y $buildDeps --no-install-recommends && rm -rf /var/lib/apt/lists/* && wget -O python.tar.xz "https://www.python.org/ftp/python/$PYTHON_VERSION%[a-z]*$Python-$PYTHON_VERSION.tar.xz.asc" && export GNUPGHOME=$(mktemp -d) && gpg --keyserver ha.pool.sks-keyservers.net --recv-keys "$GPG_KEY" && gpg --batch --verify python.tar.xz.asc python.tar.xz && rm -rf "$GNUPGHOME" python.tar.xz.asc && mkdir -p /usr/src/python && tar -xJC /usr/src/python --strip-components=1 -f python.tar.xz && rm python.tar.xz && cd /usr/src/python && gnuArch="$(dpkg-architecture --query DEB_BUILD_GNU_TYPE)" && ./configure --build="$gnuArch" --enable-loadable-sqlite-extensions --enable-shared --with-system-expat --without-ensurepip && make -j $(nproc) && make install && ldconfig && apt-get purge -y --auto-remove $buildDeps && find /usr/local -depth 1 -type d -a ! -name test -o -name tests -o ! -type f -a ! -name *.pyc -o -name *.pyo -exec rm -rf '{}' + && rm -rf /usr/src/python
```

Rysunek 6.37. Warstwy manifestu



Rysunek 6.38. Zagrożenia znalezione przez skaner Quay.io

Image Vulnerabilities					
CVE	SEVERITY	PACKAGE	CURRENT VERSION	FIXED IN VERSION	INTRODUCED IN IMAGE
» CVE-2017-17805	⚠️ High	linux	3.16.51-2	3.16.51-3+deb8u1	RUN set -ex; apt-get update; apt...
» CVE-2017-17558	⚠️ High	linux	3.16.51-2	3.16.51-3+deb8u1	RUN set -ex; apt-get update; apt...
» CVE-2018-2562	⚠️ High	mysql-5.5	5.5.59-0+deb8u1	5.5.59-0+deb8u1	RUN set -ex; apt-get update; apt...
» CVE-2017-16538	⚠️ High	linux	3.16.51-2	3.16.51-3+deb8u1	RUN set -ex; apt-get update; apt...
» CVE-2017-8824	⚠️ High	linux	3.16.51-2	3.16.51-3+deb8u1	RUN set -ex; apt-get update; apt...
» CVE-2017-17806	⚠️ High	linux	3.16.51-2	3.16.51-3+deb8u1	RUN set -ex; apt-get update; apt...
» CVE-2017-16939	⚠️ High	linux	3.16.51-2	3.16.51-3+deb8u1	RUN set -ex; apt-get update; apt...
» CVE-2017-15968	⚠️ High	linux	3.16.51-2	3.16.51-3+deb8u1	RUN set -ex; apt-get update; apt...

Rysunek 6.39. Zagrożenia wraz z odpowiadającymi im numerami CVE

Vulnerabilities					
CVE	SEVERITY	PACKAGE	CURRENT VERSION	FIXED IN VERSION	INTRODUCED IN LAYER
» CVE-2016-4448	10 / 10	libxml2	2.9.1+dfsg1-5+deb8u5	(None)	RUN set -ex; apt-get upda...
» CVE-2017-17458	10 / 10	mercurial	3.1.2-2+deb8u4	3.1.2-2+deb8u6	RUN apt-get update && apt
» CVE-2017-18017	10 / 10	linux	3.16.51-2	3.16.56-1	RUN set -ex; apt-get upda...

VECTORS

Access Vector	Access Complexity	Authentication	Confidentiality Impact	Integrity Impact
⚠️ Network	⚠️ Low	⚠️ None	⚠️ Complete	⚠️ Complete
● Adjacent Network	● Medium	● Single	● Partial	● Partial
● Local	● High	● Multiple	● None	● None

DESCRIPTION

The tcpmss_mangle_packet function in net/netfilter/xt_TCMSS.c in the Linux kernel before 4.11, and 4.9.x before 4.9.36, allows remote attackers to cause a denial of service (use-after-free and memory corruption) or possibly have unspecified other impact by leveraging the presence of xt_TCMSS in an iptables action.

Rysunek 6.40. Szczegóły wykrytego zagrożenia

```

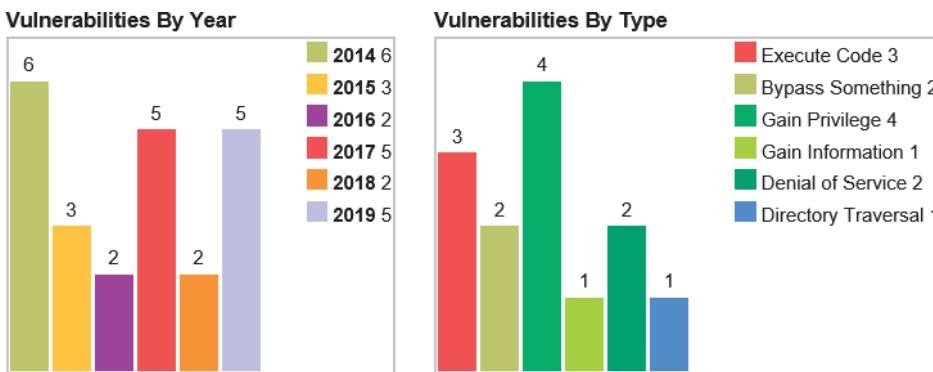
$ git clone https://github.com/anchore/anchore-engine
Cloning into 'anchore-engine'...
remote: Enumerating objects: 121, done.
remote: Counting objects: 100% (121/121), done.
remote: Compressing objects: 100% (82/82), done.
remote: Total 12804 (delta 51), reused 80 (delta 35), pack-reused 12683
Receiving objects: 100% (12804/12804), 20.92 MiB | 11.25 MiB/s, done.
Resolving deltas: 100% (8515/8515), done.
Checking out files: 100% (985/985), done.
[node1] (local) root@192.168.0.53 ~
$ cd anchore-engine
[node1] (local) root@192.168.0.53 ~/anchore-engine
$ ls
bash: LS: command not found
[node1] (local) root@192.168.0.53 ~/anchore-engine
$ ls
CHANGELOG.md          anchore_manager      requirements-test.txt
CONTRIBUTING.rst       conf                  requirements.txt
Dockerfile              docker-compose-dev.yaml scripts
LICENSE                 docker-compose.yaml    setup.py
MANIFEST.in             docker-entrypoint.sh test

```

Rysunek 6.42. Pobranie kodu źródłowego silnika Anchore

Docker » Docker : Vulnerability Statistics															
Vulnerability Trends Over Time															
Year	# of Vulnerabilities	Dos	Code Execution	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	Http Response Splitting	Bypass something	Gain Information	Gain Privileges	CSRF	File Inclusion	# of exploits
2014	6		2							1		1			
2015	3										1	1			
2016	2									1		1			
2017	5	2													
2018	2														
2019	5		1						1			1			
Total	23	2	3						1		2	1	4		
% Of All		8.7	13.0	0.0	0.0	0.0	0.0	4.3	0.0	8.7	4.3	17.4	0.0	0.0	

Rysunek 7.1. Kategorie zagrożeń kontenerów Dockera



Rysunek 7.2. Najczęściej przeprowadzane ataki na kontenery Dockera

Vulnerable and fixed packages					
The table below lists information on source packages.					
Source Package	Release	Version	Status		
linux (PTS)	jessie	3.16.56-1+deb8u1	fixed		
	jessie (security)	3.16.57-2	fixed		
	stretch	4.9.110-1	fixed		
	stretch (security)	4.9.110-3+deb9u5	fixed		
	buster	4.18.6-1	fixed		
	sid	4.18.8-1	fixed		

The information below is based on the following data on fixed versions.						
Package	Type	Release	Fixed Version	Urgency	Origin	Debian Bugs
linux	source	(unstable)	4.7.8-1	high		
linux	source	jessie	3.16.36-1+deb8u2	high	DSA-3698-1	
linux	source	wheezy	3.2.82-1	high	DLA-670-1	

Rysunek 7.7. Wersje systemu Linux, których dotyczy zagrożenie Dirty COW

Link	Usage	Description	Family
dirtycow.c	<code>./dirtycow file content</code>	Read-only write	/proc/self/mem
cowroot.c	<code>./cowroot</code>	SUID-based root	/proc/self/mem
dirtycow-mem.c	<code>./dirtycow-mem</code>	libc-based root	/proc/self/mem
pokemon.c	<code>./d file content</code>	Read-only write	PTRACE_POKEDATA
dirtycow.cr	<code>dirtycow --target --string -- offset</code>	Read-only write	/proc/self/mem
dirtycow.c	<code>./dirtycow file content</code>	Read-only write (Android)	/proc/self/mem

Rysunek 7.8. Pliki wykorzystujące podatność Dirty COW

Dockerfile	Minor changes (docker -> x86)	5 months ago
README.md	README.md	5 months ago
dirtycow.c	README.md	5 months ago
run.sh	README.md	5 months ago
safe_run.sh	README.md	5 months ago

Rysunek 7.9. Repozytorium zawierające kontener do testowania zagrożenia Dirty COW

```

165     def is_layer_safe(layer_file):
166         results = []
167         try:
168             tar = tarfile.open(layer_file, mode='r:gz')
169         except tarfile.ReadError:
170             tar = tarfile.open(layer_file, mode='r')
171
172         while True:
173             next_block = tar.next()
174             if not next_block:
175                 break
176
177             filename = next_block.name
178             link_destination = next_block.linkname
179
180             if not os.path.relpath(filename).find('..\\"') or not os.path.relpath(filename).find('..\\'):
181                 results.append((filename, 0, layer_file))
182
183             if link_destination:
184                 if link_destination[0] not in ['/','\\']:
185                     full_path = os.path.dirname(filename) + "/" + link_destination
186                     if not os.path.relpath(full_path).find('..\\"') or not os.path.relpath(full_path).find('..\\'):
187                         results.append((full_path, 1, layer_file))

```

Rysunek 7.13. Fragment skryptu w języku Python wykrywającego zagrożenie
CVE-2018-8115

Vuln ID	Summary	CVSS Severity
CVE-2018-15664	In Docker through 18.06.1-ce-rc2, the API endpoints behind the 'docker cp' command are vulnerable to a symlink-exchange attack with Directory Traversal, giving attackers arbitrary read-write access to the host filesystem with root privileges, because daemon/archive.go does not do archive operations on a frozen filesystem (or from within a chroot).	V3: 8.7 HIGH V2: 8.3 HIGH
CVE-2019-5021	Versions of the Official Alpine Linux Docker images (since v3.3) contain a NULL password for the 'root' user. This vulnerability appears to be the result of a regression introduced in December of 2015. Due to the nature of this issue, systems deployed using affected versions of the Alpine Linux container which utilize Linux PAM, or some other mechanism which uses the system shadow file as an authentication database, may accept a NULL password for the 'root' user.	V3: 9.8 CRITICAL V2: 10.0 HIGH
CVE-2019-1003065	Jenkins CloudShare Docker-Machine Plugin stores credentials unencrypted in its global configuration file on the Jenkins master where they can be viewed by users with access to the master file system.	V3: 8.8 HIGH V2: 8.0 MEDIUM

Rysunek 7.15. Strona bazy NVD

Vuln ID	Summary	CVSS Severity
CVE-2018-15664	In Docker through 18.06.1-ce-rc2, the API endpoints behind the 'docker cp' command are vulnerable to a symlink-exchange attack with Directory Traversal, giving attackers arbitrary read-write access to the host filesystem with root privileges, because daemon/archive.go does not do archive operations on a frozen filesystem (or from within a chroot).	V3: 8.7 HIGH V2: 8.3 HIGH
CVE-2019-5021	Versions of the Official Alpine Linux Docker images (since v3.3) contain a NULL password for the 'root' user. This vulnerability appears to be the result of a regression introduced in December of 2015. Due to the nature of this issue, systems deployed using affected versions of the Alpine Linux container which utilize Linux PAM, or some other mechanism which uses the system shadow file as an authentication database, may accept a NULL password for the 'root' user.	V3: 9.8 CRITICAL V2: 10.0 HIGH
CVE-2019-1003065	Jenkins CloudShare Docker-Machine Plugin stores credentials unencrypted in its global configuration file on the Jenkins master where they can be viewed by users with access to the master file system.	V3: 8.8 HIGH V2: 8.0 MEDIUM

Rysunek 7.16. Zagrożenia w bazie NVD

CVE	Package	Ubuntu 12.04 ESM (Precise Pangolin)	Ubuntu 14.04 ESM (Trusty Tahr)	Ubuntu 16.04 LTS (Xenial Xerus)	Ubuntu 18.04 LTS (Bionic Beaver)	Ubuntu 18.10 (Cosmic Cuttlefish)	Ubuntu 19.04 (Disco Dingo)	Ubuntu 19.10 (Eoan)	Links
CVE-2002-2439	gcc-4.6	needs-triage*	DNE	DNE	DNE	DNE	DNE	DNE	Mitre LP Debian
CVE-2005-4890	shadow	needed*	not-affected*	not-affected*	not-affected*	not-affected*	not-affected*	not-affected*	Mitre LP Debian
CVE-2008-7320	seahorse	DNE	DNE	needs-triage*	needs-triage*	needs-triage*	needs-triage*	needs-triage*	Mitre LP Debian
CVE-2009-1384	libpam-krb5	needed*	needed*	needed*	needed*	needed*	needed*	needed*	Mitre LP Debian
CVE-2009-5080	groff	needed*	needed*	needed*	needed*	needed*	needed*	needed*	Mitre LP Debian
CVE-2009-5155	eglibc	needed*	needed*	DNE	DNE	DNE	DNE	DNE	Mitre LP Debian
CVE-2009-5155	glibc	DNE	DNE	needed*	needed*	not-affected*	not-affected*	not-affected*	Mitre LP Debian

Rysunek 7.17. Lista zagrożeń dystrybucji Ubuntu

Red Hat Customer Portal		Products & Services	Tools	Security	Community
Security Updates	> Red Hat CVE Database				
Security Advisories		Red Hat CVE Database		Security Labs	
Keyword	GO	All	Low	Moderate	Important
		Critical			
				Filter By Year	All
	CVE	Synopsis	Impact	Publish Date	
?	CVE-2019-10156	** RESERVED ** This candidate has been reserved by an organization or individual that will use it when announcing a new security problem. When the candidate has been publicized, the details for this candidate will be provided.	Moderate	04 Jun 2019	
?	CVE-2019-10149	A flaw was found in Exim versions 4.87 to 4.91 (inclusive). Improper validation of recipient address in <code>deliver_message()</code> function in <code>/src/deliver.c</code> may lead to remote	Critical	04 Jun 2019	

Rysunek 7.18. Lista zagrożeń CVE firmy Red Hat

CVE-ID	
CVE-2019-5021	Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description	Versions of the Official Alpine Linux Docker images (since v3.3) contain a NULL password for the 'root' user. This vulnerability appears to be the result of a regression introduced in December of 2015. Due to the nature of this issue, systems deployed using affected versions of the Alpine Linux container which utilize Linux PAM, or some other mechanism which uses the system shadow file as an authentication database, may accept a NULL password for the 'root' user.
References	<p>Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.</p> <ul style="list-style-type: none">• BID:108288• URL:http://www.securityfocus.com/bid/108288• CONFIRM:https://security.netapp.com/advisory/ntap-20190510-0001/• MISC:https://alpinelinux.org/posts/Docker-image-vulnerability-CVE-2019-5021.html• MISC:https://talosintelligence.com/vulnerability_reports/TALOS-2019-0782• SUSE:openSUSE-SU-2019:1495• URL:https://lists.opensuse.org/opensuse-security-announce/2019-06/msg00004.html

Rysunek 7.19. Zagrożenie systemu Linux Alpine

```

Diffstat
-rwxr-xr-x scripts/genrootsfs.sh 3
1 files changed, 3 insertions, 0 deletions

diff --git a/scripts/genrootsfs.sh b/scripts/genrootsfs.sh
index ac760e6e0d..5118027632 100755
--- a/scripts/genrootsfs.sh
+++ b/scripts/genrootsfs.sh
@@ -39,6 +39,9 @@ ${APK:-apk} fetch --keys-dir "$keys_dir" --no-cache \
    --repositories-file "$repositories_file" \
    --stdout --quiet alpine-base | tar -zx -C "$tmp" etc/
 

+## make sure root login is disabled
+sed -i -e 's/^root:::/root:!:/' "$tmp"/etc/shadow
+
branch=edge
VERSION_ID=$(awk -F= '$1=="VERSION_ID" {print $2}' "$tmp"/etc/os-release)
case $VERSION_ID in

```

Rysunek 7.20. Zagrożony skrypt obrazu Alpine

Docker : Security Vulnerabilities														
#	CVE ID	CWE ID	# of Exploits	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication	Conf.	Integ.	Avail.
1	CVE-2019-1020014	415			2019-07-29	2019-08-19	2.1	None	Local	Low	Not required	Partial	None	None
docker-credential-helpers before 0.6.3 has a double free in the List functions.														
2	CVE-2019-16884	863		Bypass	2019-09-25	2019-10-07	5.0	None	Remote	Low	Not required	None	Partial	None
runc through 1.0.0-r08, as used in Docker through 19.03.2-ce and other products, allows AppArmor restriction bypass because libcontainer/rootfs_linux.go incorrectly checks mount targets, and thus a malicious Docker image can mount over a /proc directory.														
3	CVE-2019-15752	264		+Priv	2019-08-28	2019-09-04	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
Docker Desktop Community Edition before 2.1.0.1 allows local users to gain privileges by placing a Trojan horse docker-credential-wincred.exe file in %PROGRAMDATA%\DockerDesktop\Version-bin\ as a low-privilege user, and then waiting for an admin or service user to authenticate with Docker, restart Docker, or run 'docker login' to force the command.														
4	CVE-2019-14271	94			2019-07-29	2019-08-28	7.5	None	Remote	Low	Not required	Partial	Partial	Partial
In Docker 19.03.x before 19.03.1 linked against the GNU C Library (aka glibc), code injection can occur when the nswitch facility dynamically loads a library inside a chroot that contains the contents of the container.														
5	CVE-2019-13509	532			2019-07-18	2019-08-27	5.0	None	Remote	Low	Not required	Partial	None	None
In Docker CE and EE before 18.09.8 (as well as Docker EE before 17.06.2-ee-23 and 18.x before 18.03.1-ee-10), Docker Engine in debug mode may sometimes add secrets to the debug log. This applies to a scenario where docker stack deploy is run to redeploy a stack that includes (non external) secrets. It potentially applies to other API users of the stack API if they read the secret.														
6	CVE-2019-5736	216		Exec Code	2019-09-21	2019-09-03	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
runc through 1.0-r06, as used in Docker before 18.09.2 and other products, allows attackers to overwrite the host runc binary (and consequently obtain host root access) by leveraging the ability to execute a command as root within one of these types of containers: (1) a new container with an attacker-controlled image, or (2) an existing container, to which the attacker previously had write access, that can be attached with docker exec. This occurs because of file-descriptor mishandling, related to /proc/self/exe.														
7	CVE-2019-15664	362		Dir. Trav.	2019-05-23	2019-06-25	6.2	None	Local	High	Not required	Complete	Complete	Complete
In Docker through 18.06.1-ce-r02, the API endpoints behind the 'docker cp' command are vulnerable to a symlink-exchange attack with Directory Traversal, giving attackers arbitrary read/write access to the host filesystem with root privileges, because 'daemon/archive.go' does not do archive operations on a frozen filesystem (or from within a chroot).														
8	CVE-2019-15514	502			2018-08-31	2018-11-09	6.5	None	Remote	Low	Single system	Partial	Partial	Partial
HandleRequestAsync in Docker for Windows before 18.06.0-ce-rc3-win68 (edge) and before 18.06.0-ce-win72 (stable) serialized requests over the \\pipe\dockerBackend named pipe without verifying the validity of the serialized .NET objects. This would allow a malicious user in the "docker-users" group (who may not otherwise have administrator access) to escalate to administrator privileges.														
9	CVE-2017-14992	20		DoS	2017-11-01	2017-11-22	4.3	None	Remote	Medium	Not required	None	None	Partial
Lack of content verification in Docker-CE (Also known as Moby) versions 1.12.6-0, 1.10.3, 17.05.0, 17.03.1, 17.03.2, 17.06.0, 17.06.1, 17.06.2, 17.09.0, and earlier allows a remote attacker to cause a Denial of Service via a crafted image layer payload, aka gzip bombing.														
10	CVE-2017-11468	770		DoS	2017-07-20	2019-10-02	5.0	None	Remote	Low	Not required	None	None	Partial
Docker Registry before 2.6.2 in Docker Distribution does not properly restrict the amount of content accepted from a user, which allows remote attackers to cause a denial of service (memory consumption) via the manifest endpoint.														
11	CVE-2017-7297				2017-03-28	2019-10-02	6.5	None	Remote	Low	Single system	Partial	Partial	Partial

Rysunek 7.23. Baza zagrożeń CVE platformy Docker

CVE	Vendors	Products	Updated	CVSS
CVE-2019-14271	1 Docker	1 Docker	2019-08-28	7.5
		In Docker 19.03.x before 19.03.1 linked against the GNU C Library (aka glibc), code injection can occur when the nsswitch facility dynamically loads a library inside a chroot that contains the contents of the container.		
CVE-2019-13509	1 Docker	1 Docker	2019-08-27	5.0
		In Docker CE and EE before 18.09.8 (as well as Docker EE before 17.06.2-ee-23 and 18.x before 18.03.1-ee-10), Docker Engine in debug mode may sometimes add secrets to the debug log. This applies to a scenario where docker stack deploy is run to...		
CVE-2019-1020014	1 Docker	1 Credential Helpers	2019-08-19	2.1
		docker-credential-helpers before 0.6.3 has a double free in the List functions.		
CVE-2018-15664	1 Docker	1 Docker	2019-06-25	6.2
		In Docker through 18.06.1-ce-rc2, the API endpoints behind the 'docker cp' command are vulnerable to a symlink-exchange attack with Directory Traversal, giving attackers arbitrary read-write access to the host filesystem with root privileges....		
CVE-2019-5736	10 Docker, Google, Linuxcontainers and 7 more	12 Docker, Kubernetes Engine, Lxc and 9 more	2019-06-03	9.3

Rysunek 7.24. Baza saus zawierająca zagrożenia CVE

```
~$ kubectl logs kube-bench-node
[INFO] 2 Worker Node Security Configuration
[INFO] 2.1 Kubelet
[FAIL] 2.1.1 Ensure that the --allow-privileged argument is set to false (Scored)
[PASS] 2.1.2 Ensure that the --anonymous-auth argument is set to false (Scored)
[PASS] 2.1.3 Ensure that the --authorization-mode argument is not set to AlwaysAllow (Scored)
[PASS] 2.1.4 Ensure that the --client-ca-file argument is set as appropriate (Scored)
[PASS] 2.1.5 Ensure that the --read-only-port argument is set to 0 (Scored)
[FAIL] 2.1.6 Ensure that the --streaming-connection-idle-timeout argument is not set to 0 (Scored)
[FAIL] 2.1.7 Ensure that the --protect-kernel-defaults argument is set to true (Scored)
[FAIL] 2.1.8 Ensure that the --make-iptables-util-chains argument is set to true (Scored)
[FAIL] 2.1.9 Ensure that the --keep-terminated-pod-volumes argument is set to false (Scored)
[FAIL] 2.1.10 Ensure that the --hostname-override argument is not set (Scored)
[FAIL] 2.1.11 Ensure that the --event-qps argument is set to 0 (Scored)
[PASS] 2.1.12 Ensure that the --tls-cert-file and --tls-private-key-file arguments are set as appropriate (Scored)
[PASS] 2.1.13 Ensure that the --advisor-port argument is set to 0 (Scored)
[FAIL] 2.1.14 Ensure that the RotateKubeletClientCertificate argument is set to true
[FAIL] 2.1.15 Ensure that the RotateKubeletServerCertificate argument is set to true
[INFO] 2.2 Configuration Files
[FAIL] 2.2.1 Ensure that the kubelet.conf file permissions are set to 644 or more restrictive (Scored)
[FAIL] 2.2.2 Ensure that the kubelet.conf file ownership is set to root:root (Scored)
[FAIL] 2.2.3 Ensure that the kubelet service file permissions are set to 644 or more restrictive (Scored)
[FAIL] 2.2.4 2.2.4 Ensure that the kubelet service file ownership is set to root:root (Scored)
[FAIL] 2.2.5 Ensure that the proxy kubeconfig file permissions are set to 644 or more restrictive (Scored)
[FAIL] 2.2.6 Ensure that the proxy kubeconfig file ownership is set to root:root (Scored)
[WARN] 2.2.7 Ensure that the certificate authorities file permissions are set to 644 or more restrictive (Scored)
[WARN] 2.2.8 Ensure that the client certificate authorities file ownership is set to root:root
```

Rysunek 8.2. Uruchomienie narzędzia Kube Bench w węźle roboczym

```
[root@master ~]# docker run --pid=host -t aquasec/kube-bench:latest master --version 1.8
[INFO] 1 Master Node Security Configuration
[INFO] 1.1 API Server
[FAIL] 1.1.1 Ensure that the --anonymous-auth argument is set to false (Scored)
[PASS] 1.1.2 Ensure that the --basic-auth-file argument is not set (Scored)
[PASS] 1.1.3 Ensure that the --insecure-allow-any-token argument is not set (Scored)
[PASS] 1.1.4 Ensure that the --kubelet-https argument is set to true (Scored)
[PASS] 1.1.5 Ensure that the --insecure-bind-address argument is not set (Scored)
[PASS] 1.1.6 Ensure that the --insecure-port argument is set to 0 (Scored)
[PASS] 1.1.7 Ensure that the --secure-port argument is not set to 0 (Scored)
[FAIL] 1.1.8 Ensure that the --profiling argument is set to false (Scored)
[FAIL] 1.1.9 Ensure that the --repair-malformed-updates argument is set to false (Scored)
[FAIL] 1.1.10 Ensure that the admission control policy is not set to AlwaysAdmit (Scored)
[FAIL] 1.1.11 Ensure that the admission control policy is set to AlwaysPullImages (Scored)
[FAIL] 1.1.12 Ensure that the admission control policy is set to DenyEscalatingExec (Scored)
[FAIL] 1.1.13 Ensure that the admission control policy is set to SecurityContextDeny (Scored)
[FAIL] 1.1.14 Ensure that the admission control policy is set to NamespaceLifecycle (Scored)
[FAIL] 1.1.15 Ensure that the --audit-log-path argument is set as appropriate (Scored)
[FAIL] 1.1.16 Ensure that the --audit-log-maxage argument is set to 30 or as appropriate (Scored)
```

Rysunek 8.3. Uruchomienie narzędzia w węźle głównym

```
→ ~ kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
hello-minikube-7c77b68cff-qhq5b  1/1     Running   0          29m
→ ~ kubectl plugin sniff hello-minikube-7c77b68cff-qhq5b
[+] Sniffing on pod: hello-minikube-7c77b68cff-qhq5b container: namespace:
[+] Verifying pod status
NAME           I READY   STATUS    RESTARTS   AGE
hello-minikube-7c77b68cff-qhq5b  1/1     Running   0          29m
[+] checking if tcpdump already exist
-rwxrwxr-x 1 1000 1000 2700408 Jun 22 14:08 /static-tcpdump
[+] static tcpdump is already installed on container!
[+] Starting remote sniffing!
```

Rysunek 8.5. Weryfikacja stanu podów za pomocą wtyczki ksniff

```

[15] 1:kubectl-dig:          "l13o1" 12:21 20-May-19
Viewing: Processes For: whole machine
Source: Live System Filter: evt.type!=switch
Select View      Containers
Connections      List all the containers running on this machine, and the resources that each of them uses.
Containers      Containers
Containers Errors  Tips
Directories      Select a container and click enter to drill down into it. At that point, you will be able to
Errors          access several views that will show you the details of the selected container.
File Opens List
Files           Columns
I/O by Type     CPU: Amount of CPU used by the container.
K8s Controllers PROCS: Number of processes currently running inside the container.
K8s Deployments THREADS: Number of threads currently running inside the container.
K8s Namespaces  VIRT: Total virtual memory for the process.
K8s Pods        RES: Resident non-swapped memory for the process.
K8s ReplicaSets FILE: Total (input+output) file I/O bandwidth generated by the container, in bytes per second
K8s Services    .
K8s Services    NET: Total (input+output) network bandwidth generated by the container, in bytes per second.
Marathon Apps   ENGINE: Container type.
Marathon Groups IMAGE: Container image name.
Mesos Frameworks ID: Container ID. The format of this column depends on the containerization technology. For e
Mesos Tasks    xample, Docker ID are 12 characters hexadecimal digit strings.
New Connections NAME: Name of the container.
Page Faults
Processes
Processes CPU
Processes Errors
Processes FD Usage
Server Ports
Slow File I/O
Socket Queues
Spectrogram-File Action Hotkeys
Spy Systlog  a: docker attach (docker attach %container.id)
Spy Users    b: bash shell (docker exec -i -t %container.id /bin/bash)
System Calls f: follow logs (docker logs -f %container.id)
Threads      h: image history (docker history %container.image)
Traces List  i: docker inspect (docker inspect %container.id)
Traces Spectrogram k: docker kill (docker kill %container.id)
Traces Summary l: docker logs (docker logs %container.id)
                s: docker stop (docker stop %container.id)
                z: docker pause (docker pause %container.id)
                u: docker unpause (docker unpause %container.id)
F1Help F2Views F4Filter F5Echo F6Dig F7Legend F8Actions F9Sort F12Spectro CTRL+FSearch p Pause  11/78(14.1%)

```

Rysunek 8.6. Przykład użycia wtyczki kubectl-dig

Use "rakkess [command] --help" for more information about a command.				
NAME	LIST	CREATE	UPDATE	DELETE
bindings		✓		
configmaps	✓	✓	✓	✓
controllerrevisions.apps	✓	✓	✓	✓
cronjobs.batch	✓	✓	✓	
daemonsets.apps	✓	✓	✓	✓
daemonsets.extensions	✓	✓	✓	✓
deployments.apps	✓	✓	✓	✓
deployments.extensions	✓	✓	✓	✓
endpoints	✓	✓	✓	✓
events	✓	✓	✓	✓
events.events.k8s.io	✓	✓	✓	✓
horizontalpodautoscalers.autoscaling	✓	✓	✓	✓
ingresses.extensions	✓	✓	✓	✓
ingresses.networking.k8s.io	✓	✓	✓	✓
jobs.batch	✓	✓	✓	✓
leases.coordination.k8s.io	✓	✓	✓	✓
limitranges	✓	✓	✓	✓
localsubjectaccessreviews.authorization.k8s.io	✓	✓	✓	✓
networkpolicies.extensions	✓	✓	✓	✓

Rysunek 8.7. Przykład użycia wtyczki rakkess

```
$ docker network create my-network
79896668b483d38ca83642f3197afdc9188116d079b70203c065c053dfe8980
[node1] (local) root@192.168.0.28 ~
$ docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
80302a34d531   bridge    bridge      local
21276d934921   host      host      local
79896668b483   my-network  bridge      local
03bf600bbfec   none      null      local
```

Rysunek 9.16. Tworzenie sieci

```
root@a05699ed73ab:/# set | grep -i elasticSearch
ELASTICSEARCH_ENV_CA_CERTIFICATES_JAVA_VERSION=20140324
ELASTICSEARCH_ENV_ES_VERSION=2.2.2
ELASTICSEARCH_ENV_GOSU_VERSION=1.7
ELASTICSEARCH_ENV_JAVA_DEBIAN_VERSION=8u111-b14-2~bpo8+1
ELASTICSEARCH_ENV_JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre
ELASTICSEARCH_ENV_JAVA_VERSION=8u111
ELASTICSEARCH_ENV_LANG=C.UTF-8
ELASTICSEARCH_NAME=/ubuntu/elasticsearch
ELASTICSEARCH_PORT=tcp://172.17.0.2:9200
ELASTICSEARCH_PORT_9200_TCP=tcp://172.17.0.2:9200
ELASTICSEARCH_PORT_9200_TCP_ADDR=172.17.0.2
ELASTICSEARCH_PORT_9200_TCP_PORT=9200
ELASTICSEARCH_PORT_9200_TCP_PROTO=tcp
ELASTICSEARCH_PORT_9300_TCP=tcp://172.17.0.2:9300
ELASTICSEARCH_PORT_9300_TCP_ADDR=172.17.0.2
ELASTICSEARCH_PORT_9300_TCP_PORT=9300
ELASTICSEARCH_PORT_9300_TCP_PROTO=tcp
```

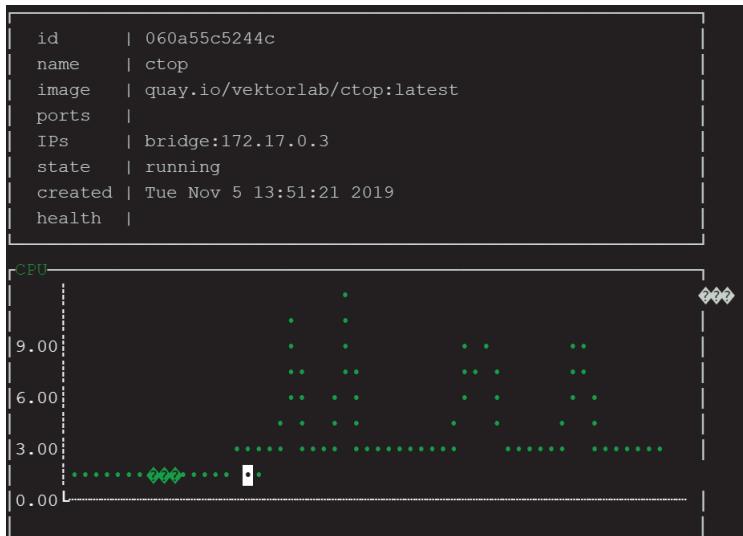
Rysunek 9.19. Zmienne środowiskowe kontenera Ubuntu zawierające informacje o kontenerze Elasticsearch

ctop - 13:51:59 UTC 3 containers							
NAME	CID	CPU	MEM	NET RX/TX	IO R/W	PIDS	
ctop	060a55c5244c	1%	9M / 31.4G	0B / 0B	0B / 0B	13	
laughing_greib..	c92da791b0af	0%	5M / 31.4G	0B / 0B	32K / 0B	2	
sad_bouman	db2de5b34fc2	-	-	-	-	-	

Rysunek 10.9. Przykład użycia narzędzia ctop

3 containers							
		CPU	MEM	NET RX/TX	IO R/W	PIDS	
[o]	single view						
[l]	log view						
[s]	stop	0a55c5244c	1%	9M / 31.4G	0B / 0B	0B / 0B	13
[p]	pause	2da791b0af	0%	5M / 31.4G	0B / 0B	32K / 0B	2
[r]	restart	2de5b34fc2	-	-	-	-	-
[e]	exec shell						
[c]	cancel						

Rysunek 10.10. Opcje wizualizacji zdarzeń



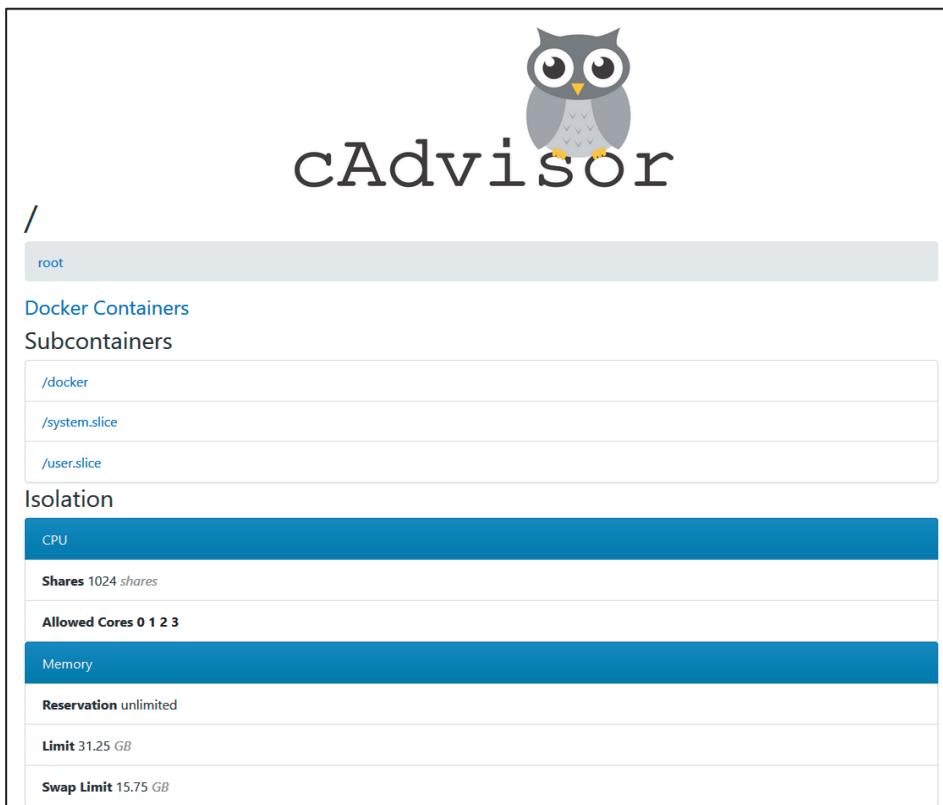
Rysunek 10.11. Szczegółowe informacje o wykorzystaniu procesora w kontenerze

```

Project- /          config- Name:    lazyteam/lazydocker
          |           ID:      sha256:02b45falbe32a3ae8f7b3bdb79930cd6e301f606cf0f2f5ea17
Containers- running  sweet_torvalds  lazyte  Tags:    lazyteam/lazydocker:latest
          |           Size:   69.94MB
          |           Created: Tue, 05 Nov 2019 08:18:57 UTC
Images- <none>       <none>
        |<none>       <none>       ID      TAG          SIZE      COMMAND
        |<none>       <none>       02b45falbe  lazyteam/lazydocker:latest  10.19MiB  COPY file:3836b2a7104
lazyteam/lazydocker latest  |<missing>          56.52MiB  COPY file:6048005b542
arm32v7/golang  1.12.6-alpine3 |<missing>          0B        ENTRYPOINT ["/bin/laz
arm32v6/golang  1.12.6-alpine3 |<missing>          0B        LABEL org.opencontain
arm64v8/golang  1.12.6-alpine3 |<missing>          0B        ARG VERSION
                                |<missing>          0B        ARG VCS_REF
                                |<missing>          0B        ARG BUILD_DATE
Volumes-

```

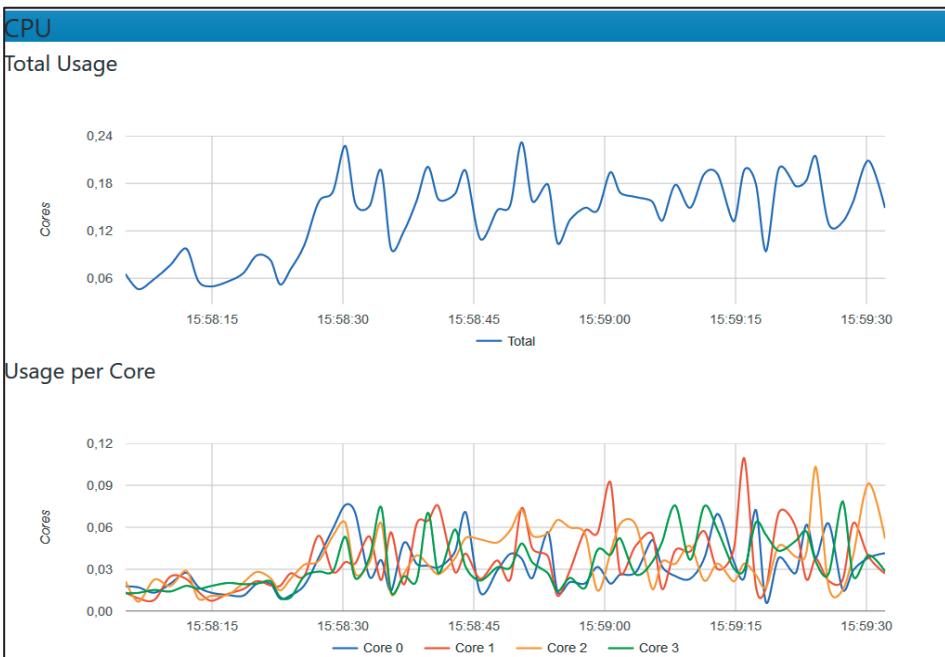
Rysunek 10.12. Szczegółowe informacje o kontenerze uzyskane za pomocą narzędzia LazyDocker



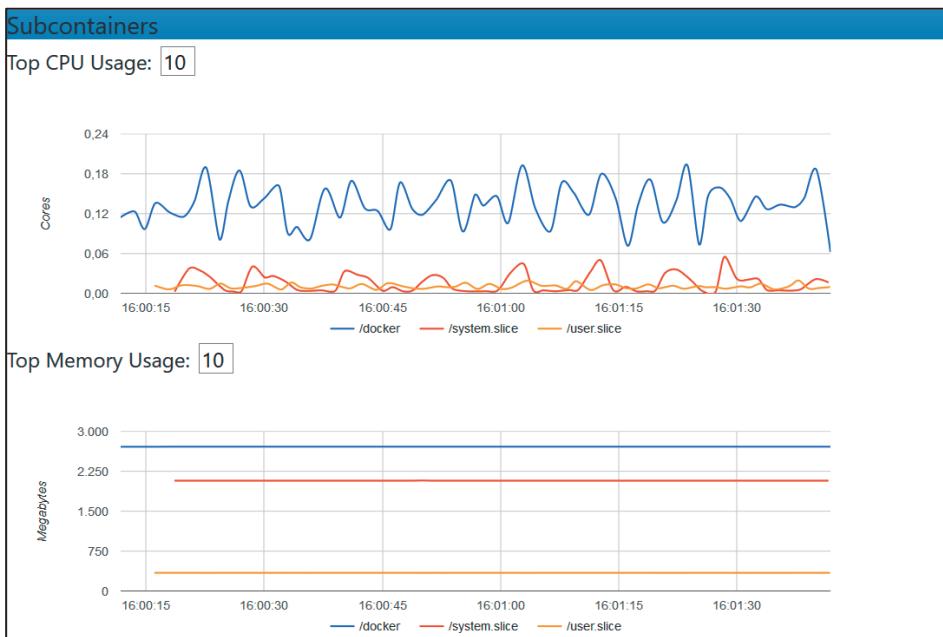
Rysunek 10.14. Informacje o uruchomionych kontenerach

Usage Overview																
Processes																
User	PID	PPID	Start Time	CPU %	MEM %	RSS	Virtual Size	Status	Running Time	Command	Container					
root	21.799	21.781	15:52	3.30	0.20	79.45 MiB	1.19 GiB	Rsl	00:00:12	cadvisor	/docker					
root	5.166	1	Feb25	0.90	0.30	115.30 MiB	1.59 GiB	Ssl	00:30:17	dockerd	/system.slice/docker.service					
root	5.236	1	Feb25	0.30	0.10	42.13 MiB	1.12 GiB	Ssl	00:10:15	containerd	/system.slice/containerd.servi					
root	9.787	9.701	Feb25	0.30	0.10	38.20 MiB	732.30 MiB	Ssl	00:09:39	telegraf	/dfb008f7351e6447bcfad5					
root	23.689	23.686	10:57	0.20	0.90	293.71 MiB	8.46 GiB	S1	00:00:40	java	/ba32db3e25d30b389564aa					
root	23.833	23.830	10:57	0.20	0.90	291.14 MiB	8.46 GiB	S1	00:00:41	java	/cb534d308dc18d2a6462E5					
root	29.421	29.420	10:12	0.20	0.80	278.55 MiB	10.33 GiB	S1	00:00:43	java	/c8dbe3c1d899c5658310bb					
root	29.579	29.578	10:12	0.20	0.80	273.49 MiB	10.33 GiB	S1	00:00:43	java	/aef16f0b2253a1a729b688					
root	1.566	1.536	15:14	0.10	0.00	1.97 MiB	111.04 MiB	Ss	00:00:04	bash	/user.slice					
root	25.741	25.697	10:06	0.10	0.80	286.04 MiB	10.33 GiB	S1	00:00:38	java	/757db37c3fe051d0779286					
root	25.891	25.842	10:06	0.10	0.90	292.18 MiB	10.33 GiB	S1	00:00:38	java	/4f8a0664245c4a97ef4255					
root	30.840	30.790	10:13	0.10	1.00	321.45 MiB	8.47 GiB	S1	00:00:41	java	/5d65cec329c52416d15651					
root	30.979	30.933	10:13	0.10	1.00	321.80 MiB	8.47 GiB	S1	00:00:41	java	/1c74596fcbad96fa966af3					
root	1	0	Feb25	0.00	0.00	4.03 MiB	186.63 MiB	Ss	00:00:38	systemd	/					
root	2	0	Feb25	0.00	0.00	0.00 B	0.00 B	S	00:00:00	kthreadd	/					
root	3	2	Feb25	0.00	0.00	0.00 B	0.00 B	S	00:00:01	ksoftirqd/0	/					
root	5	2	Feb25	0.00	0.00	0.00 B	0.00 B	S<	00:00:00	kwworker/0:OH	/					
root	7	2	Feb25	0.00	0.00	0.00 B	0.00 B	S	00:00:01	migration/0	/					
root	8	2	Feb25	0.00	0.00	0.00 B	0.00 B	S	00:00:00	rcu_bh	/					
root	9	2	Feb25	0.00	0.00	0.00 B	0.00 B	S	00:01:43	rcu_sched	/					
root	10	2	Feb25	0.00	0.00	0.00 B	0.00 B	S<	00:00:00	lru-add-drain	/					

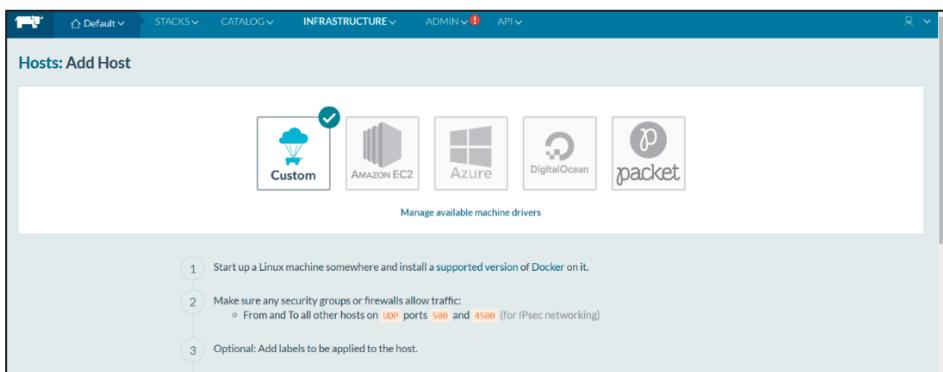
Rysunek 10.15. Informacje o procesach



Rysunek 10.16. Wykorzystanie rdzeni procesora



Rysunek 10.17. Wykorzystanie procesora i pamięci przez poszczególne kontenery



Rysunek 11.1. Dodawanie hosta za pomocą narzędzia Rancher



Rysunek 11.2. Rejestrowanie hosta za pomocą narzędzia Rancher

Hosts **Add Host**

ACTIVE

master

- 195.154.79.10 | 18.09.7
- Ubuntu 16.04.6 LTS (4.4.0)
- 2x2.1 GHz | 1.95 GiB | 44.1 GiB

Stack: healthcheck

- healthcheck-1 10.42.76.44

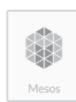
Stack: ipsec

- cni-driver-1 None
- ipsec-1 10.42.135.94

Sidekicks

Rysunek 11.3. Informacje o hoście w interfejsie narzędzia Rancher

Add Environment

Name	Description
<input type="text" value="h2g.lab"/>	e.g. Environment for developer experimentation
Environment Template	
 Cattle  Kubernetes  Mesos  Swarm  Windows	
Orchestration: Cattle Framework: Network Services, Scheduler, Healthcheck Service Networking: Rancher IPsec	

Rysunek 11.4. Szablon środowiska w narzędziu Rancher

Environment Templates [Add Template](#)

An environment template allows users to define a different combination of infrastructure services to be deployed.
The infrastructure services includes but not limited to container orchestration (i.e. cattle, kubernetes, mesos, swarm, networking) or rancher services (i.e healthcheck, dns, metadata, scheduling, service discovery and storage)

Name	Description	Stacks	Public
Cattle	Default Cattle template	network-services, ipsec, scheduler, healthcheck	✓  
Kubernetes	Default Kubernetes template	kubernetes, network-services, ipsec, healthcheck	✓  
Mesos	Default Mesos template	mesos, network-services, ipsec, scheduler, healthcheck	✓  
Swarm	Default Swarm template	portainer, swarm, network-services, ipsec, scheduler, healthcheck	✓  
Windows	Experimental Windows template	windows, windows-network-services	✓  

Rysunek 11.5. Szablony środowiska w narzędziu Rancher

Default  CATALOG  ADMIN 

Catalog: All

 Alfresco An ECM and BPM platform. View Details	 Alibaba Cloud DNS Rancher External DNS service powered by Alibaba Cloud View Details	 Apache Guacamole Apache Guacamole is a clientless remote desktop gateway. It supports standard protocols like VNC, RDP, and SSH. View Details	 Apache Kafka Apache Kafka cluster View Details
---	---	--	---

Rysunek 11.6. Katalog aplikacji w interfejsie narzędzia Rancher

Containers Add Container Show System

Actions Set Search: < > 25 Items

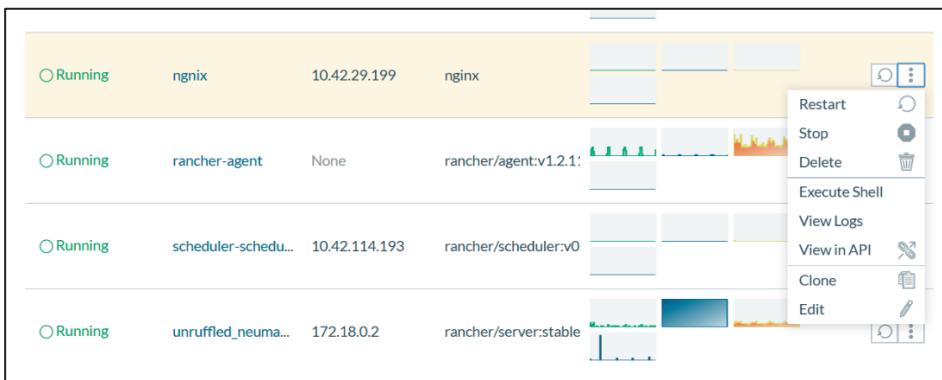
State	Name	IP Address	Host	Image	Command
Starting	coredns	None	master	sha256:eb516548c180f8a6e023	None
Starting	coredns	None	master	sha256:eb516548c180f8a6e023	None
Stopped	coredns-fb8b8dcf-ndkx	10.42.226.169	master	k8s.gcr.io/pause:3.1	None
Stopped	coredns-fb8b8dcf-zcf7x	10.42.7.135	master	k8s.gcr.io/pause:3.1	None
Stopped	etcd	None	master	sha256:2c4adeb21b4ff8ed3309c	None
Stopped	etcd-master	None	master	k8s.gcr.io/pause:3.1	None
Running	healthcheck-healthcheck-1	10.42.76.44	master	rancher/healthcheck:v0.3.8	healthcheck--metadata-addr...
Running	ipsec-cni-driver-1	None	master	rancher/netv0.13.17	start-cni-driversh
Burnin	ipsec-ipsec-1	10.42.135.94	master	rancher/netholder	sh-c-echo Refer to router side

Rysunek 11.7. Panel kontenerów w narzędziu Rancher

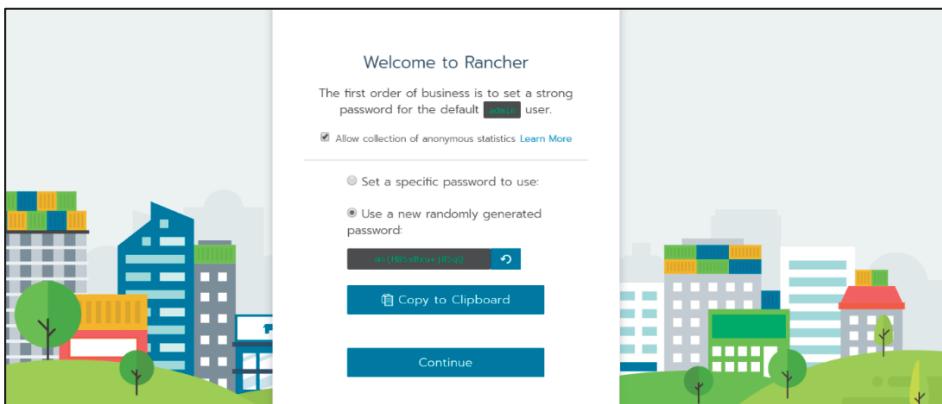
Add Container

Name	Description	
nginx	nginx	
Select Image*	<input checked="" type="checkbox"/> Always pull image before creating	
nginx		
+ Port Map		
Public Host Port	Private Container Port	Protocol
80	80	TCP
Show Host IP field		

Rysunek 11.8. Dodawanie kontenera za pomocą narzędzia Rancher



Rysunek 11.9. Uruchomione kontenery widoczne w interfejsie narzędzia Rancher



Rysunek 11.10. Strona powitalna narzędzia Rancher

This screenshot shows the "Users" section of the Rancher interface. At the top, there's a navigation bar with links for Global, Clusters, Apps, Users, Settings, Security, and Tools. Below the navigation is a search bar and a "Refresh Group Memberships" button. The main area is titled "Users" and contains a table with columns for State, Name, ID, Local Username, and Actions. One row is visible, showing "Active" status, the name "Default Admin (admin)", the ID "user-xcnpm", the local username "admin", and an "Edit" icon. There are also "Deactivate" and "Delete" buttons at the top of the table.

Rysunek 11.11. Sekcja Users w narzędziu Rancher

This screenshot shows the "Add Cluster - Select Cluster Type" screen. At the top, there's a navigation bar with links for Global, Clusters, Apps, Users, Settings, Security, and Tools. The main area has two main sections: "From existing nodes (Custom)" and "Import an existing cluster". Under "From existing nodes (Custom)", there's a "Create a new Kubernetes cluster using RKE, out of existing bare-metal servers or virtual machines." link. Under "Import an existing cluster", there's a "Import an existing Kubernetes cluster. The provider that created it will continue to manage the provisioning and configuration of the cluster." link. Below these are four boxes for infrastructure providers: "Amazon EC2", "Azure", "DigitalOcean", "Linode", and "vSphere".

Rysunek 11.12. Tworzenie klastra

This screenshot shows the "Add Cluster - Custom" screen. At the top, there's a navigation bar with links for Global, Clusters, Apps, Users, Settings, Security, and Tools. The main area starts with a "Cluster Name" field containing "k8s-katacoda-staging" and an "Add a Description" link. Below this is a "Member Roles" section with a note about controlling access and permissions. Then comes a "Labels & Annotations" section with a "None" link. At the bottom, there's a "Cluster Options" section and an "Edit as a Form" button.

Rysunek 11.13. Wybór typu klastra

```

1 #
2 # Cluster Config
3 #
4 docker_root_dir: /var/lib/docker
5 enable_cluster_alerting: false
6 enable_cluster_monitoring: false
7 enable_network_policy: false
8 local_cluster_auth_endpoint:
9   enabled: true
10 name: k8s-katacoda-staging
11 #
12 # Rancher Config
13 #
14 rancher_kubernetes_engine_config:
15   addon_job_timeout: 30
16   authentication:
17     strategy: x509
18   ignore_docker_version: true
19 #

```

Rysunek 11.14. Plik konfiguracyjny klastra

1 Node Options
Choose what roles the node will have in the cluster
Node Role
 etcd Control Plane Worker
[Show advanced options](#)

2 Run this command on one or more existing machines already running a supported version of Docker.

```
sudo docker run -d --privileged --restart=unless-stopped --net=host -v /etc/kubernetes:/etc/kubernetes -v /var/run:/var/run rancher/rancher-agent:v2.3.2 --server https://288679532-80-cykor1a01.environments.katacoda.com --token hgj5qlvlgmggn124j1jsqslvtdkdhnx67j2wzpz6vxj95krmmvt --ca-checksum c82cc0b5a72a3f0e0cb29bcd14af55ccalc2ed73949c9b974746ecc3397a6cd3 --etcd --controlplane --worker
```

Rysunek 11.15. Konfiguracja węzła klastra

This cluster is currently **Provisioning**: areas that interact directly with it will not be available until the API is ready.
Waiting for etcd and controlplane nodes to be registered

Nodes [Edit Cluster](#)

<input type="button" value="Delete"/>	<input type="button" value="Search"/>					
State	Name	Roles	Version	CPU	RAM	Pods
<input checked="" type="checkbox"/> registering	master 172.17.0.68	<input type="checkbox"/> Worker	n/a	n/a	n/a	n/a

Waiting to register with Kubernetes

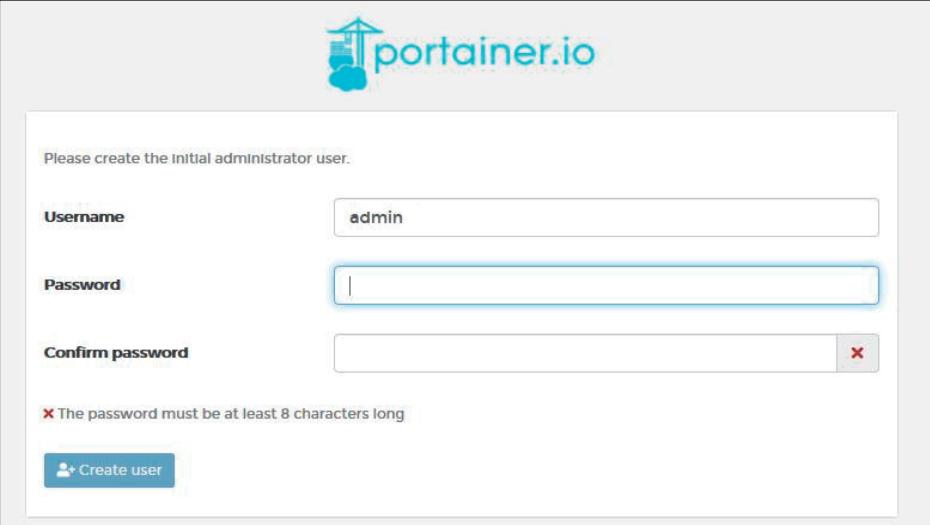
Rysunek 11.16. Rejestracja głównego węzła klastra

Clusters [Add Cluster](#)

<input type="button" value="Delete"/>	<input type="button" value="Search"/>				
State	Cluster Name	Provider	Nodes	CPU	RAM
<input checked="" type="checkbox"/> Provisioning	k8s-katacoda-staging	Custom	1	n/a	n/a

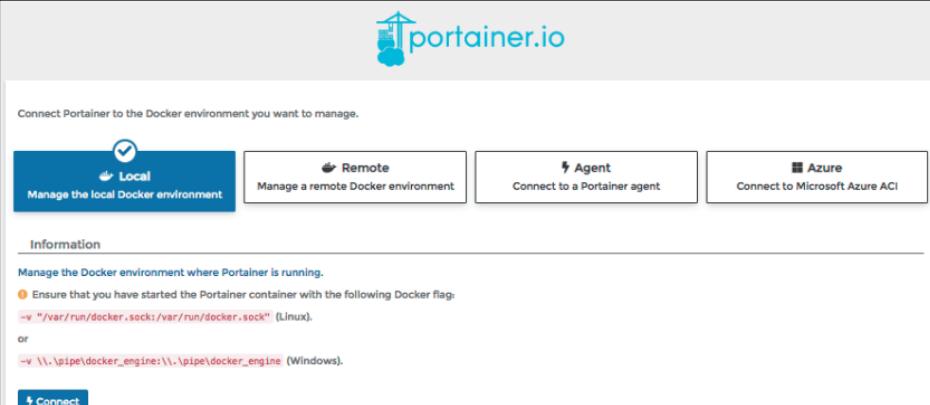
Waiting for etcd and controlplane nodes to be registered

Rysunek 11.17. Stan klastra w interfejsie narzędzia Rancher



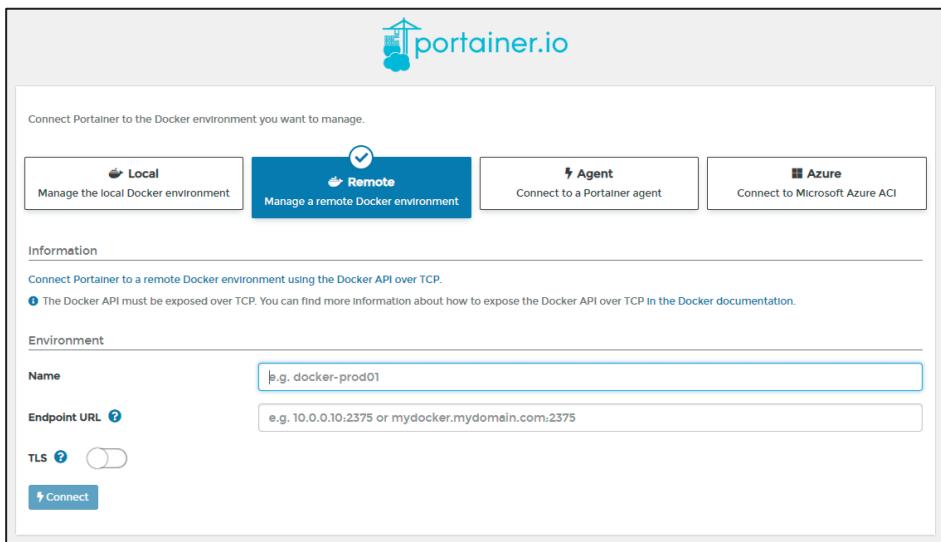
The screenshot shows the initial configuration screen for creating an administrator user. At the top, the Portainer logo is displayed. Below it, a message reads: "Please create the initial administrator user." There are three input fields: "Username" (containing "admin"), "Password" (empty), and "Confirm password" (empty). A red error message below the password field states: "The password must be at least 8 characters long". At the bottom left is a blue "Create user" button.

Rysunek 11.19. Definiowanie konta administratora

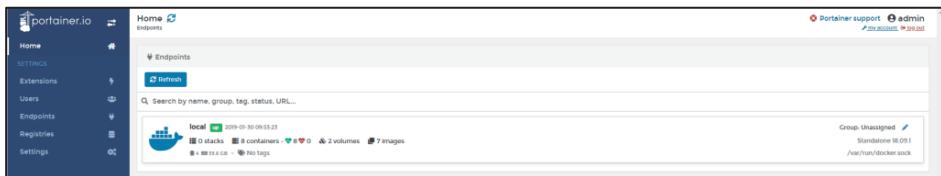


The screenshot shows the "Connect Portainer to the Docker environment you want to manage" screen. It features four options: "Local" (selected, highlighted in blue), "Remote", "Agent", and "Azure". The "Local" option has a sub-section titled "Information" with instructions for managing the Docker environment where Portainer is running. It includes a note about starting the Portainer container with the "-v "/var/run/docker.sock:/var/run/docker.sock" flag (Linux) or "\\.\pipe\docker_engine:\\.\pipe\docker_engine" (Windows). At the bottom is a blue "Connect" button.

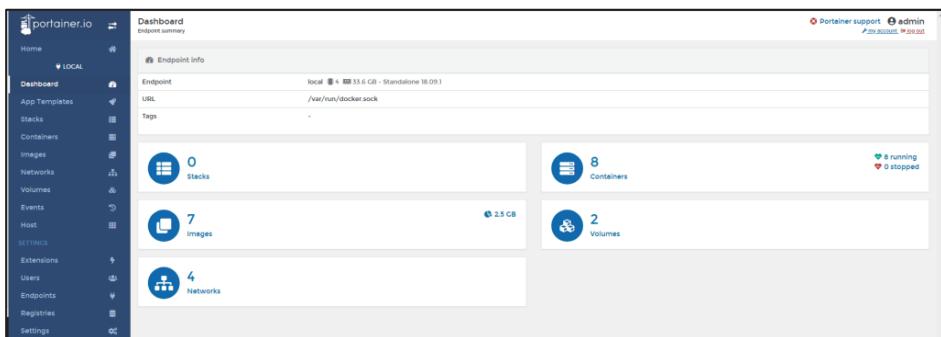
Rysunek 11.20. Lokalna instalacja narzędzia Portainer



Rysunek 11.21. Zarządzanie zewnętrznym środowiskiem za pomocą narzędzia Portainer



Rysunek 11.22. Interfejs graficzny narzędzia Portainer



Rysunek 11.23. Główna strona narzędzia Portainer

Name	State	Quick actions	Stack	Image	Created	IP Address	Published Ports	Ownership
portainer	running		-	portainer/portainer	2019-01-30 09:51:42	172.17.0.8	9000:9000	administrators
traefik_proxy	running		-	traefik	2019-01-28 14:20:18	172.18.0.2	80:80 8000:8080	administrators
telegraf	running		-	svttelegraf/latest	2019-01-24 17:14:34	172.17.0.7	-	administrators
[REDACTED]	running		-	tomcat7war	2019-01-23 16:03:07	172.17.0.5	9003:8009 9003:8080	administrators
[REDACTED]	running		-	tomcat7war	2019-01-23 16:02:07	172.17.0.2	9004:8009 9004:8080	administrators
[REDACTED]	running		-	tomcat7war	2019-01-23 15:56:06	172.17.0.6	9005:8009 9005:8080	administrators
[REDACTED]	running		-	tomcat7war	2019-01-23 15:49:07	172.17.0.5	9002:8009 9002:8080	administrators
[REDACTED]	running		-	tomcat7war	2019-01-23 15:40:07	172.17.0.4	9001:8080 9101:8009	administrators

Rysunek 11.24. Lista kontenerów w interfejsie narzędzia Portainer

Container details	
Containers > nginx-web	
Container status	
ID	e2b1b62ee25363a264af65eef72f68cb416066d5de219b1d8c3ea830957c335e
Name	nginx-web
IP address	172.17.0.2
Status	Running for 6 minutes
Created	2019-11-13 16:15:20
Start time	2019-11-13 16:15:20

Rysunek 11.25. Szczegóły kontenera w interfejsie narzędzia Portainer

```

Log viewer settings

Auto-refresh logs  (on)
Wrap lines  (on)
Display timestamps  (off)

Fetch All logs
Search Filter...
Lines 100
Actions           

```

```

* directory[/opt/gitlab/sv/grafana/control] action create (up to date)
* link[/opt/gitlab/init/grafana] action create (up to date)
* file[/opt/gitlab/sv/grafana/down] action delete (up to date)
* directory[/opt/gitlab/service] action create (up to date)
* link[/opt/gitlab/service/grafana] action create (up to date)
* ruby_block[wait for grafana service socket] action run (skipped due to not_if)
  execute the ruby block reload log service

```

Rysunek 11.26. Szczegółowe informacje o dziennikach kontenera

Container inspect	
Containers > nginx > Inspect	Portainer support admin
	my account log out
Inspect	Tree Text
▼ 9c52919fc46953034a7b54ba15520a816b62e875f4c1b66967841066c9e550d7:	
AppArmorProfile: docker-default	
Args: [-g, daemon off;]	
Config: { ArgsEscaped: true, AttachStderr: false, AttachStdin: false, AttachStdout: false }	
Created: 2019-11-13T16:15:34.455100998Z	
Driver: overlay2	
ExecIDs:	
GraphDriver: { Data: [object Object], Name: overlay2 }	
HostConfig: { AutoRemove: false, Binds: "/var/lib/docker/containers/9c52919fc46953034a7b54ba15520a816b62e875f4c1b66967841066c9e550d7:/var/lib/docker/containers/9c52919fc46953034a7b54ba15520a816b62e875f4c1b66967841066c9e550d7/hostname", HostnamePath: "/var/lib/docker/containers/9c52919fc46953034a7b54ba15520a816b62e875f4c1b66967841066c9e550d7/hosts", HostsPath: "/var/lib/docker/containers/9c52919fc46953034a7b54ba15520a816b62e875f4c1b66967841066c9e550d7/hosts", Id: "9c52919fc46953034a7b54ba15520a816b62e875f4c1b66967841066c9e550d7", Image: "sha256:540a289bab6ccbb80086a9b803cf0c4cefe3bc85cdefa199b69614525199f", LogPath: "/var/lib/docker/containers/9c52919fc46953034a7b54ba15520a816b62e875f4c1b66967841066c9e550d7/9c52919fc46953034a7b54ba15520a816b62e875f4c1b66967841066c9e550d7-nginx.log", MountLabel: "", Mounts: [{ Object: { ArgsEscaped: true, AttachStderr: false, AttachStdin: false, AttachStdout: false, Cmd: ["/usr/sbin/nginx -g daemon off;"], DockerFile: null, Environment: ["HTTP_PROXY=http://127.0.0.1:8888"], Labels: {}, Name: "nginx", Labels: {} }, Target: "/var/lib/docker/containers/9c52919fc46953034a7b54ba15520a816b62e875f4c1b66967841066c9e550d7/nginx" }] }	

Rysunek 11.27. Szczegółowe informacje o kontenerze

The screenshot shows the Portainer interface for the container 'nginx'. The top navigation bar includes 'Container statistics' (selected), 'Containers > nginx > Stats', 'Portainer support' (with a red exclamation icon), 'my account' (with a blue gear icon), and 'admin' (with a blue user icon). Below the navigation is a section titled 'About statistics' with a note: 'This view displays real-time statistics about the container nginx as well as a list of the running processes inside this container.' A 'Refresh rate' dropdown is set to '5s'. Three charts are displayed: 'Memory usage' (blue bar chart, mostly at 0.0B), 'CPU usage' (blue bar chart, mostly at 0.0%), and 'Network usage (aggregate)' (blue bar chart, mostly at 0.0B). Below the charts is a 'Processes' section with a search bar.

Rysunek 11.28. Wskaźniki opisujące pracę kontenera

Images					Settings
		Actions			
	Remove	+ Build a new image	Import	Export	
<input type="text"/> Search...					
Id	Tags		Size	Created	
<input type="checkbox"/> sha256:f6e8af4562c14ab06a2c9f3698e39e...	Unused	dockersamples/examplevotingapp_vote:<none>	83.6 MB	2017-01-11 02:54:06	
<input type="checkbox"/> sha256:2b1e6048c5398e19b011fc1b67c2d1...	Unused	dockersamples/examplevotingapp_worker:<none>	961.9 MB	2017-04-07 21:31:15	
<input type="checkbox"/> sha256:540a289bab6cb1bf880086a9b803cf...	nginx:latest		126.2 MB	2019-10-23 02:26:03	
<input type="checkbox"/> sha256:36726735dc3c2c86ff47a937c72d53...	Unused	postgres:<none>	206.3 MB	2019-10-17 06:40:54	

Rysunek 11.29. Strona Images w narzędziu Portainer

Network list

Networks

Remove Add network

Search...

Name	Stack	Scope	Driver	Attachable	Internal	IPAM Driver	IPAM Subnet	IPAM Gateway	Ownership
bridge	-	local	bridge	false	false	default	172.17.0.0/16	-	administrators
docker_gwbridge	-	local	bridge	false	false	default	172.19.0.0/16	172.19.0.1	administrators
host	-	local	host	false	false	default	-	-	administrators
Ingress	-	swarm	overlay	false	false	default	10.255.0.0/16	10.255.0.1	administrators
none	-	local	null	false	false	default	-	-	administrators

Items per page: 10

Rysunek 11.30. Lista sieci w interfejsie narzędzia Portainer

Volume list

Volumes

Remove Add volume

Search...

Name	Stack	Driver	Mount point	Created
21e33b5cd62877c370d2a7a897257142f8b1e...	-	local	/var/lib/docker/volumes/2[...]fabebc6a46bd598e8db/_data	2019-11-13 17:15:34
3f5caacc6318d6eee36a9a4690fecbc25b376...	-	local	/var/lib/docker/volumes/3[...]9d6a7ab47de42d4b9df/_data	2019-11-13 17:15:34

Items per page: 10

Rysunek 11.31. Lista woluminów w interfejsie narzędzia Portainer

Application templates list

Templates

Add template

Show container templates

Select a category

Search...

Portainer Agent stack	Manage all the resources in your Swarm cluster	<input checked="" type="checkbox"/> Update	<input type="checkbox"/> Delete	portainer
OpenFaaS stack	Serverless functions made simple	<input checked="" type="checkbox"/> Update	<input type="checkbox"/> Delete	serverless
IronFunctions stack	Open-source serverless computing platform	<input checked="" type="checkbox"/> Update	<input type="checkbox"/> Delete	serverless

Rysunek 11.32. Szablony aplikacji w interfejsie narzędzia Portainer

Show container templates

Search...

 Registry <small>container</small> Docker image registry	<input type="button" value="Update"/> <input type="button" value="Delete"/> docker
 Nginx <small>container</small> High performance web server	<input type="button" value="Update"/> <input type="button" value="Delete"/> webserver
 Httpd <small>container</small> Open-source HTTP server	<input type="button" value="Update"/> <input type="button" value="Delete"/> webserver
 Caddy <small>container</small> HTTP/2 web server with automatic HTTPS	<input type="button" value="Update"/> <input type="button" value="Delete"/> webserver
 MySQL <small>container</small> The most popular open-source database	<input type="button" value="Update"/> <input type="button" value="Delete"/> database

Rysunek 11.33. Szablony kontenerów w interfejsie narzędzia Portainer

 **CrateDB** container
An open-source distributed SQL database

 **Elasticsearch** container
Open-source search and analytics engine

 **Gitlab CE** container
Open-source end-to-end software development platform

 **Minio** container
A distributed object storage server built for cloud applications and devops

Rysunek 11.34. Szablony narzędzia GitLab w interfejsie narzędzia Portainer

Port mapping

Portainer will automatically assign a port if you leave the host port empty.

host	9080	→	container	80	TCP	UDP	<input type="button" value="Delete"/>
host	9443	→	container	443	TCP	UDP	<input type="button" value="Delete"/>
host	9022	→	container	22	TCP	UDP	<input type="button" value="Delete"/>

Rysunek 11.35. Wiązanie portów w interfejsie narzędzia Portainer

Container list

Containers

Columns

Search...

State	Quick actions	Stack	Image	Created	IP Address	Published Ports	Ownership
healthy	   	-	gitlab/gitlab-ce:latest	2019-11-13 18:11:19	172.17.0.2	<input type="button" value="80"/> <input type="button" value="443"/> <input type="button" value="8080"/> <input type="button" value="9080"/>	administrators
running	   	test	portainer/agent:latest	2019-11-13 18:06:54	10.0.0.10	<input type="button" value="9001"/> <input type="button" value="9001"/>	administrators

Rysunek 11.36. Lista kontenerów w interfejsie narzędzia Portainer

Deploying Portainer to Docker Swarm Cluster

Step 1 of 5 >

Step 1 - Deploy Swarm Cluster

This scenario deploys Portainer to a Swarm cluster. A two-node cluster has been created for you. If you would like more information on how this was created, complete the Katacoda scenario on [Getting Started With Swarm Mode](#).

You can view the status of the Swarm cluster with the command `docker node ls ✓`

CONTINUE

```
$ docker node ls
Error response from daemon: This node is not a swarm manager. Use "docker swarm init" or "docker swarm join" to connect this node to swarm and try again.
$ docker swarm init
Swarm initialized: current node (o1jqayqqnncyub5ewrvf0ymb) is now a manager.

To add a worker to this swarm, run the following command:

  docker swarm join --token SWMTK-1-2z8m2oz1d1kyse84s78r6idqgw19yc3qqd27t8h3dy0vd92ub-3mcifnw6ld
  pltubkslzemaku 172.17.0.77:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

$ docker node ls
ID          HOSTNAME   STATUS    AVAILABILITY  MANAGER  ST
ATUS        ENGINE VERSION
o1jqayqqnncyub5ewrvf0ymb * host01     Ready     Active      Leader
18.09.7
$
```

Rysunek 11.37. Wdrożenie narzędzia Portainer w klastrze Docker Swarm

Container list

Containers

Containers **Columns** **Settings**

Start **Stop** **Kill** **Restart** **Pause** **Resume** **Remove** **+ Add container**

Search...

<input type="checkbox"/> Name	State	Quick actions	Stack	Image	Create
<code>portainer.1.9rl1kq9j3oknva9q5...</code>	running		-	<code>portainer/portainer:latest</code>	2019-11-

Items per page: 10

Rysunek 11.38. Uruchomiony kontener Portainer

Deploying Portainer to Docker Swarm Cluster

Step 4 of 5 >

Step 4 - Deploy Template

One of the many features of Portainer is it's available to deploy services based on pre-defined containers.

Application templates list

Templates

+ Add template

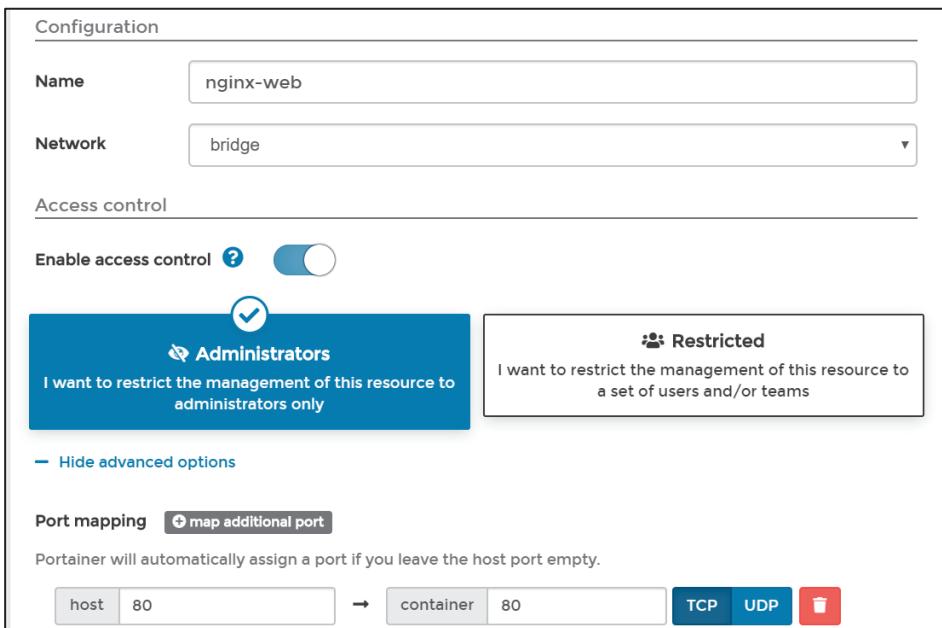
Show container templates

Search...

Registry container Docker image registry	Update Delete
Nginx container High performance web server	Update Delete
Httpd container Open-source HTTP server	Update Delete

Task - Deploy nginx

Rysunek 11.39. Wdrożenie szablonu serwera Nginx



Rysunek 11.40. Konfiguracja kontenera z serwerem Nginx

Container details										
Image	<code>nginx:latest@sha256:540a289bab6cb1bf880086a9b803cf0c4cefe38cbb5cdefa199b69614525199f</code>									
Port configuration	<code>0.0.0.0:80 → 80/tcp</code>									
CMD	<code>nginx -g daemon off;</code>									
ENTRYPOINT	<code>null</code>									
ENV	<table border="1"> <tr> <td>PATH</td><td>/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin</td></tr> <tr> <td>NGINX_VERSION</td><td>1.17.5</td></tr> <tr> <td>NJS_VERSION</td><td>0.3.6</td></tr> <tr> <td>PKG_RELEASE</td><td>1~buster</td></tr> </table>		PATH	/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin	NGINX_VERSION	1.17.5	NJS_VERSION	0.3.6	PKG_RELEASE	1~buster
PATH	/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin									
NGINX_VERSION	1.17.5									
NJS_VERSION	0.3.6									
PKG_RELEASE	1~buster									
Labels	<table border="1"> <tr> <td>maintainer</td><td>NGINX Docker Maintainers <docker-maint@nginx.com></td></tr> </table>		maintainer	NGINX Docker Maintainers <docker-maint@nginx.com>						
maintainer	NGINX Docker Maintainers <docker-maint@nginx.com>									

Rysunek 11.41. Szczegóły kontenera z serwerem Nginx

ID	HOSTNAME	STATUS	AVAILABILITY	MANAGER STATUS
olw40ii82gs1xgfjtmy5h2nrc*	docker-master1	Ready	Active	Leader
1vvjz6c4bfe4ez16kzkyuc31o	docker-master2	Ready	Active	Reachable
q8em4kafthkmlphkuuab48eau	docker-worker1	Ready	Active	
hb4ylcxthc6kevimap7wxr9e	docker-worker2	Ready	Active	

Rysunek 11.42. Lista węzłów klastra Docker Swarm

ID	NAME	MODE	REPLICAS	IMAGE
tpmlzwsun4xt	httpd_httpd	global	4/4	httpd:2.4
9040dr4bjqtf	jboss_jboss	global	4/4	jboss/wildfly:latest
rhjjxlrqimj7	portainer_agent	global	4/4	portainer/agent:latest
rs5wrgflhmax	portainer_portainer	replicated	1/1	portainer/portainer:lates
oxj7bf6e36kj	tomcat_tomcat	global	4/4	tomcat:8.0

Rysunek 11.43. Lista usług w klastrze Docker Swarm

Services				
	Name	Stack	Image	Scheduling Mode
<input type="checkbox"/>	httpd_httpd	httpd	httpd:2.4	global 4 / 4
<input type="checkbox"/>	jboss_jboss	jboss	jboss/wildfly:latest	global 4 / 4
<input type="checkbox"/>	portainer_agent	portainer	portainer/agent:latest	global 4 / 4
<input type="checkbox"/>	portainer_portainer	portainer	portainer/portainer:latest	replicated 1 / 1 Scale
<input type="checkbox"/>	tomcat_tomcat	tomcat	tomcat:8.0	global 4 / 4

Rysunek 11.44. Aktywne usługi w klastrze Docker Swarm

Cluster status											
Nodes		Resources									
Nodes		4									
Docker API version		1.39									
Total CPU		4									
Total memory		6.27 GB									
Go to cluster visualizer											
Nodes											
Search...											
Name	Role	CPU	Memory	Engine	IP Address	Status					
docker-master1	manager	1	2.1 GB	18.09.1	192.168.1.131	ready					
docker-master2	manager	1	2.1 GB	18.09.1	192.168.1.132	ready					
docker-worker1	worker	1	1 GB	18.09.1	192.168.1.133	ready					
docker-worker2	worker	1	1 GB	18.09.1	192.168.1.134	ready					

Rysunek 11.45. Węzły tworzące klaszt Docker Swarm