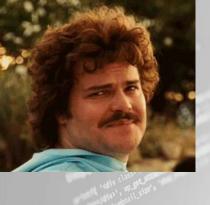


Dariusz Porowski



- Microsoft
 - Software Development Engineer
 - Commercial Software Engineering (CSE)
- Dariusz.Porowski@microsoft.com
- <u>DariuszPorowski.MS</u>
- GitHub.com/DariuszPorowski
- @DariuszPorowski







Source: https://twitter.com/dexhorthy/status/856639005462417409

Polkomtel



- Operator of Plus mobile network which was launched in 1996 and currently has around 13,5 million users.
- Plus network covers around 99,6% of the country's population.
- www.plus.pl



Polkomtel's problem statement



- Try Java Spring Boot application in Cloud environment
- App designed for Apache Tomcat on Linux (on-premise)
- Improve scalability and reliability
- Reduce administrative overhead (no OS management)
- Near zero-downtime deployments
- DevOps cycle
 - Improve time to market
 - Improve QA
 - Continuous show development progress to business

PoC with Microsoft





Outcome after Architecture Design Session



- VMs with Linux and Tomcat server (laaS)
 - Administrative overhead with OS and application server maintenance
- Managed Kubernetes cluster on Azure (AKS)
 - Application designed for on-premise environment
 - Monolith, not microservices/modern architecture
- Managed Container instances on Azure (ACI)
 - No "Out of the Box" web app related features
- Web App for Containers (App Service for Linux)
 - Great for monolith architecture
 - PaaS benefits designed for Web Apps e.g. slot swaps for rolling updates
 - Bring Your Own Code / Bring Your Own Container

Agenda / Lessons learned



- Azure App Service on Linux / Web App for Containers
- Own Docker image
- Troubleshooting SSH, Logs
- Generalize Docker image and push settings
- Docker image optimization
- DevOps pipeline

Azure App Service



- App Service
 - App Service -> Windows
 - App Service on Linux -> Linux
- Bring Your Own Code
 - Web App
 - Node.js
 - PHP
 - .NET Core
 - Ruby
- Bring Your Own Container
 - Web App for Containers







DEMO #1

Getting started with a Web App for Containers

Custom Docker image



Your own Dockerfile

```
FROM tomcat:8.5
```

EXPOSE 8080

CMD ["catalina.sh", "run"]

Docker Multi-Stage builds



- Great for local development
- Save time for setup development environment

FROM buildBaseImage AS build RUN ...

FROM finalBaseImage AS final
COPY --from=build <buildSource> <finalDestination>

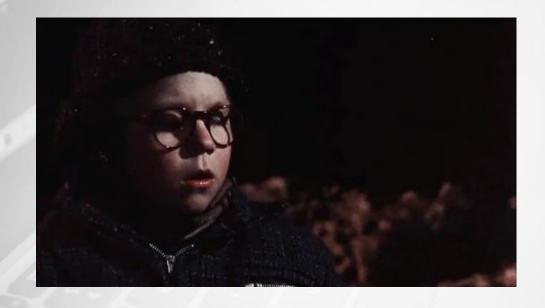


DEMO #2

Docker Multi-Stage builds

Azure, we have a problem!





Connect to container using SSH



- SSH Server
- Mandatory!
 - Username: root and password: Docker!
 - Port: 2222
 - sshd_config
 - Ciphers: aes128-cbc,3des-cbc,aes256-cbc
 - MACs: hmac-sha1,hmac-sha1-96
- https://<appName>.scm.azurewebsites.net/webssh/host

```
COPY sshd_config /etc/ssh/
RUN apt-get update \
     && apt-get install -y --no-install-recommends dialog \
     && apt-get update \
     && apt-get install -y --no-install-recommends openssh-server \
     && echo "root:Docker!" | chpasswd
EXPOSE 2222 <yourPort>
```

Docker Container logs



- API endpoint
 - https://<app>.scm.azurewebsites.net/api/logs/docker
- Logs
 - https://<app>.scm.azurewebsites.net/api/vfs/LogFiles/<logName>.log

```
2018-05-28 16:20:31.181 INFO - Recycling container because of AppFrameworkVersionChange and
                                                                             appFrameworkVersion = daporo.azurecr.io/javadockerwebapp:508
                                                                             2018-05-28 16:21:01.807 INFO - Starting container for site
2018-05-28T16:21:57.985612764Z
                                                                             2018-05-28 16:21:01.807 INFO - docker run -d -p 43752:8080 --name javadockerwebapp 1 -e
2018-05-28T16:21:58.018594116Z
                                                                             WEBSITES ENABLE APP SERVICE STORAGE=false -e WEBSITES PORT=8080 -e
2018-05-28T16:21:58.018921324Z
                                                                             DOCKER CUSTOM IMAGE NAME=daporo.azurecr.io/javadockerwebapp:502 -e WEBSITE SITE NAME=JavaDockerWebApp -e
2018-05-28T16:21:58.019208231Z ( (
                                                                             WEBSITE_AUTH_ENABLED=False -e WEBSITE_ROLE_INSTANCE_ID=0 -e
2018-05-28T16:21:58.023148733Z
                                                                             WEBSITE INSTANCE ID=ce1389f312f45210c378ab5dd90fe102bd111fb6d6c9292d318c00ae21073bae -e
2018-05-28T16:21:58.023406040Z
                                                                             HTTP LOGGING ENABLED=1 daporo.azurecr.io/javadockerwebapp:508
2018-05-28T16:21:58.023624245Z
                                                       (v1.5.10.RELEASE)
2018-05-28T16:21:58.064580203Z
                                :: Spring Boot ::
                                                                             2018-05-28 16:22:27.086 INFO - Container javadockerwebapp_1 for site javadockerwebapp initialized
2018-05-28T16:21:58.064899011Z
2018-05-28T16:21:59.534300945Z 2018-05-28 16:21:59.510 INFO 27 --- [ost-startStop-1] demo.HelloWorldApplication
                                                                                                                              : Starting HelloWorldApplication v0.0.1-SNAPSHOT on
4cf040d0f480 with PID 27 (/usr/local/tomcat/webapps/ROOT/WEB-INF/classes started by root in /usr/local/tomcat)
2018-05-28T16:21:59.538322749Z 2018-05-28 16:21:59.538 INFO 27 --- [ost-startStop-1] demo.HelloWorldApplication
                                                                                                                              : No active profile set, falling back to default
profiles: default
2018-05-28T16:22:00.214035093Z 2018-05-28 16:22:00.213 INFO 27 --- [ost-startStop-1] ationConfigEmbeddedWebApplicationContext : Refreshing
org.springframework.boot.context.embedded.AnnotationConfigEmbeddedWebApplicationContext@7e9b9eec: startup date [Mon May 28 16:22:00 GMT 2018]; root of context hierarchy
2018-05-28T16:22:09.466301449Z 2018-05-28 16:22:09.465 INFO 27 --- [ost-startStop-1] o.s.web.context.ContextLoader
                                                                                                                              : Root WebApplicationContext: initialization completed in
9252 ms
```





DEMO #3

Connect to container using SSH Get container logs

Generic Docker image and custom settings



- Avoid any hardcoded configuration!
- Use environment variables as much as possible!
 - WEBSITES_PORT for custom expose port inside container
 - Connection Strings
 - Configuration
 - etc...

Docker image optimization

- BASE image matter!
 - Alpine Linux is a good idea!
- Layers count matter!
 - Less is MORE!
- Avoid "latest" for tag
 - Use own tag, e.g., build id

PS	C: '	\>	doc	ker	ımage

tomcat

tomcat

maven

maven

REPOSITORY

daporo/javahellotomcat

daporo/javahelloalpine 0.0.1

8.5-jre8 3.5-jdk-8

TAG

0.0.1

8.5-alpine 3.5-jdk-8-alpine 955acc16db1e 108db0e7c85e 83d235b52940

91138c237846

4d2b29dd21e8

IMAGE ID

d881f0fedc24

CREATED

6 days ago

6 minutes ago

44 minutes ago

→ 558MB → 748MB → 106MB

SIZE

596MB

→ 121MB

9 days ago 6 weeks ago

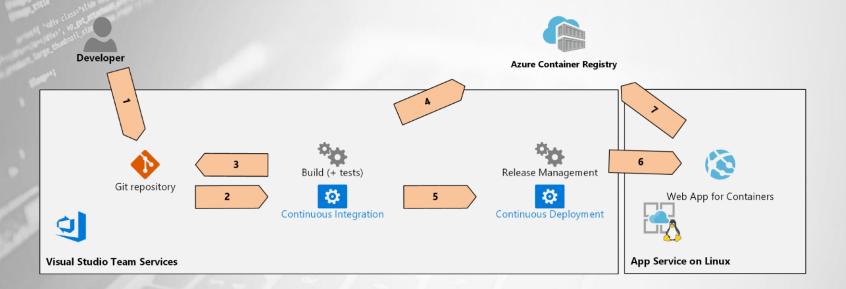
6 days ago

→ 118MB

DevOps with Visual Studio Team Services



Do not play with Docker manually!







DEMO #4

CI/CD in VSTS for Web App for Containers

Resources



- Technical Case Study Polkomtel
 - https://aka.ms/Polkomtel
- Web App for Containers
 - https://docs.microsoft.com/en-us/azure/app-service/containers
- Visual Studio Team Services
 - https://docs.microsoft.com/en-us/vsts

- GitHub.com/DariuszPorowski
- DariuszPorowski.MS







Source: https://twitter.com/dexhorthy/status/856639005462417409

BONUS! What's new for Web App for Containers



- Multi-Container App
- Docker Compose or Kubernetes deployment experience
- Persistent storage support
 - WEBSITES_ENABLE_APP_SERVICE_STORAGE: true
- Docs: https://docs.microsoft.com/en-us/azure/app-service/containers/tutorial-multi-container-app

BONUS! Tech Case Study



- K8s related Tech Case Study with Quadient
- "From long semi-manual VMs setup to quick standardized solution deployment based on Docker containers and Kubernetes for legacy monolithic Java application"
- https://aka.ms/Quadient



Tech-Acceleration.com

- Containers and Microservices
- Serverless
- Artificial Intelligence and Bots
- Internet of Things
- Machine Learning and Big Data













