

CONTAINERS PYTHON NA AWS

Lucas Borsatto



POR QUE PYTHON?

PORQUE SIM :)

PROJETO

server.py x

server.py ▸ ...

```
1  from flask import Flask
2
3  app = Flask(__name__)
4
5  @app.route('/user/<int:id>', methods=['GET'])
6  def user(id):
7      return str(id)
8
9  if __name__ == '__main__':
10     app.run(host='0.0.0.0')
```

POR QUE CONTAINERS?

SOLUÇÃO COM CONTAINERS

```
server.py  Dockerfile x
Dockerfile ▸ ...
1  FROM nitincipher/docker-ubuntu-python-pip:latest
2
3  WORKDIR /usr/src/app
4
5  COPY . /usr/src/app
6
7  RUN pip install Flask
8
9  EXPOSE 5000
10
11 CMD ["python3", "server.py"]
```

O QUE SÃO CONTAINERS

Características

- Ambientes independentes do SO
- Compartilham somente o kernel
- Podem funcionar como um SO independente

Estrutura

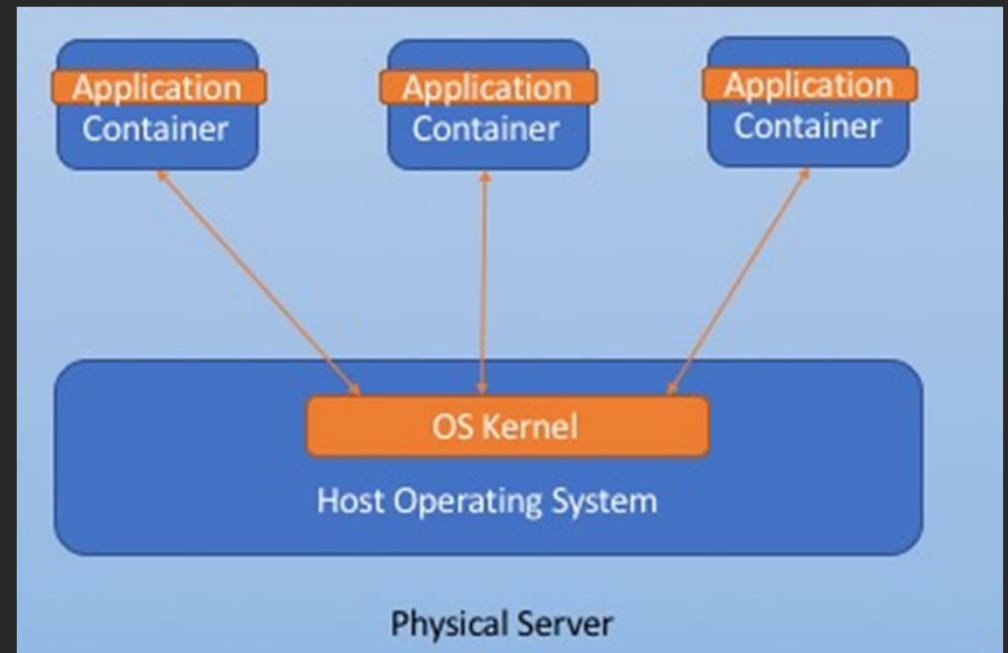


IMAGE BASE

server.py

Dockerfile x

Dockerfile ▸ ...

```
1 FROM nitincyper/docker-ubuntu-python-pip:latest
2
3 WORKDIR /usr/src/app
4
5 COPY . /usr/src/app
6
7 RUN pip install Flask
8
9 EXPOSE 5000
10
11 CMD ["python3", "server.py"]
```


DIRETÓRIO BASE

```
server.py  Dockerfile x
Dockerfile ▸ ...
1  FROM nitincyper/docker-ubuntu-python-pip:latest
2
3  WORKDIR /usr/src/app
4
5  COPY . /usr/src/app
6
7  RUN pip install Flask
8
9  EXPOSE 5000
10
11 CMD ["python3", "server.py"]
```

CÓPIA DO PROJETO

```
server.py  Dockerfile x
Dockerfile ▸ ...
1  FROM nitincyper/docker-ubuntu-python-pip:latest
2
3  WORKDIR /usr/src/app
4
5  COPY . /usr/src/app
6
7  RUN pip install Flask
8
9  EXPOSE 5000
10
11 CMD ["python3", "server.py"]
```

DEPENDÊNCIAS

server.py

Dockerfile x

Dockerfile ▸ ...

```
1 FROM nitincyper/docker-ubuntu-python-pip:latest
2
3 WORKDIR /usr/src/app
4
5 COPY . /usr/src/app
6
7 RUN pip install Flask
8
9 EXPOSE 5000
10
11 CMD ["python3", "server.py"]
```

EXPOSIÇÃO DE PORTA

server.py

Dockerfile x

Dockerfile ▸ ...

```
1 FROM nitincyper/docker-ubuntu-python-pip:latest
2
3 WORKDIR /usr/src/app
4
5 COPY . /usr/src/app
6
7 RUN pip install Flask
8
9 EXPOSE 5000
10
11 CMD ["python3", "server.py"]
```

INICIAR EXECUÇÃO

server.py

Dockerfile x

Dockerfile ▸ ...

```
1 FROM nitincyper/docker-ubuntu-python-pip:latest
2
3 WORKDIR /usr/src/app
4
5 COPY . /usr/src/app
6
7 RUN pip install Flask
8
9 EXPOSE 5000
10
11 CMD ["python3", "server.py"]
```

IMAGEM DO CONTAINER

```
lucas.simao@suportelocal-Inspiron-5457:~/Área de Trabalho/Meetup$ docker build --tag=meetup .
```

```
Sending build context to Docker daemon 5.632kB
```

```
Step 1/7 : FROM python:2.7-slim
```

```
---> ca96bab3e2aa
```

```
Step 2/7 : WORKDIR /usr/src/app
```

```
---> Running in bd37572b316d
```

```
Removing intermediate container bd37572b316d
```

```
---> 4e9076e08723
```

```
Step 3/7 : COPY . /usr/src/app
```

```
---> 7c7494694b57
```

```
Step 4/7 : RUN pip install Flask
```

```
---> Running in 07b0f5943f8c
```

```
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 won't be maintained after that date. A future version of pip will drop support for Python 2.7.
```

```
Collecting Flask
```

```
  Downloading https://files.pythonhosted.org/packages/9a/74/670ae9737d14114753b8c8fdf2e8bd212a05d3b361ab15b44937dfd40985/Flask-1.0.3-py2.py3-none-any.whl (92kB)
```

```
Step 5/7 : EXPOSE 5000
```

```
---> Running in abc670126415
```

```
Removing intermediate container abc670126415
```

```
---> f2135bec0afb
```

```
Step 6/7 : ENV NAME World
```

```
---> Running in f5a367af5bb1
```

```
Removing intermediate container f5a367af5bb1
```

```
---> bf0dfe963810
```

```
Step 7/7 : CMD ["python", "server.py"]
```

```
---> Running in f780f1251ce3
```

```
Removing intermediate container f780f1251ce3
```

```
---> 97bd7a17b4b0
```

```
Successfully built 97bd7a17b4b0
```

```
Successfully tagged meetup:latest
```

```
lucas.simao@suportelocal-Inspiron-5457:~/Área de Trabalho/Meetup$ █
```

CONTAINER

```
lucas.simao@suportelocal-Inspiron-5457:~/Área de Trabalho/Meetup$ docker run -p 5000:5000 meetup
```

POR QUE AWS?

DEPLOY

- Variedade de serviços:
 - EC2 (Elastic Compute Cloud)
 - ECS (Elastic Container Service)
 - EBS (Elastic Beanstalk)

INICIANDO BEANSTALK

- Dependências:
 - Criação da conta na AWS
 - Criação de um IAM
 - Criação de um grupo

CRIAÇÃO DO USUÁRIO

Definir detalhes do usuário

Você pode adicionar vários usuários de uma só vez com o mesmo tipo de acesso e permissões. [Saiba mais](#)

Nome de usuário*

[+](#) Adicionar outro usuário

Selecione o tipo de acesso à AWS

Selecione como esses usuários vão acessar a AWS. As chaves de acesso e as senhas geradas automaticamente são fornecidas na última etapa. [Saiba mais](#)

Tipo de acesso*

☐

Acesso programático

Habilita uma **ID da chave de acesso** e **chave de acesso secreta** para a API da AWS, CLI, SDK, e outras ferramentas de desenvolvimento.

☐

Acesso ao Console de Gerenciamento da AWS

Habilita uma **senha** que permite que os usuários façam login no Console de Gerenciamento da AWS.

CRIAÇÃO DO GRUPO

Criar um grupo



Crie um grupo e selecione as políticas a serem anexadas ao grupo. Usar grupos é uma prática recomendada para gerenciar as permissões dos usuários por função de trabalho, acesso ao serviço da AWS, ou suas permissões personalizadas. [Saiba mais](#)

Nome do grupo

Criar política

Atualizar

Filtrar políticas ▾

Pesquisar

Exibindo 452 resultados

		Nome da política ▾	Digite	Usado como	Descrição
<input checked="" type="checkbox"/>	▶	AdministratorAccess	Função de trabalho	Nenhum	Provides full access to AWS services and resources.
<input type="checkbox"/>	▶	AlexaForBusinessDe...	Gerenciado pela AWS	Nenhum	Provide device setup access to AlexaForBusiness ser...
<input type="checkbox"/>	▶	AlexaForBusinessFull...	Gerenciado pela AWS	Nenhum	Grants full access to AlexaForBusiness resources and...
<input type="checkbox"/>	▶	AlexaForBusinessGat...	Gerenciado pela AWS	Nenhum	Provide gateway execution access to AlexaForBusine...
<input type="checkbox"/>	▶	AlexaForBusinessRe...	Gerenciado pela AWS	Nenhum	Provide read only access to AlexaForBusiness services
<input type="checkbox"/>	▶	AmazonAPIGateway	Gerenciado pela AWS	Nenhum	Provides full access to create/edit/delete APIs in Ama

Cancelar

Criar um grupo

CAPTURA DE CREDENCIAIS

Adicionar usuario

2

4

5



Êxito

Você criou com êxito os usuários mostrados abaixo. Você pode visualizar e fazer download das credenciais de segurança do usuário. Você também pode enviar um e-mail aos usuários com as instruções para fazer login no Console de Gerenciamento da AWS. Esta é a última vez que essas credenciais estarão disponíveis para download. No entanto, você pode criar novas credenciais a qualquer momento.

Os usuários com acesso ao Console de Gerenciamento da AWS podem fazer login em:

[https://\[REDACTED\].signin.aws.amazon.com/console](https://[REDACTED].signin.aws.amazon.com/console)



 Fazer download .csv

	Usuário	ID da chave de acesso	Chave de acesso secreta	Senha	Instruções de login do e-mail
	meetup-user	[REDACTED]	[REDACTED] [REDACTED] Ocultar	***** Exibir	Enviar e-mail

CONFIGURAÇÃO DA AWS

Adicionar usuario

1

2

3

4

5



Êxito

Você criou com êxito os usuários mostrados abaixo. Você pode visualizar e fazer download das credenciais de segurança do usuário. Você também pode enviar um e-mail aos usuários com as instruções para fazer login no Console de Gerenciamento da AWS. Esta é a última vez que essas credenciais estarão disponíveis para download. No entanto, você pode criar novas credenciais a qualquer momento.

Os usuários com acesso ao Console de Gerenciamento da AWS podem fazer login em:

[https://\[REDACTED\].signin.aws.amazon.com/console](https://[REDACTED].signin.aws.amazon.com/console)



 Fazer download .csv

		Usuário	ID da chave de acesso	Chave de acesso secreta	Senha	Instruções de login do e-mail
▶		meetup-user	[REDACTED]	[REDACTED] [REDACTED] Ocultar	***** Exibir	Enviar e-mail ↗

CRIAÇÃO DO BEANSTALK

```
lucas.simao@suportelocal-Inspiron-5457:~/Area de Trabalho/Meetup$ eb init
```

```
Select a default region
1) us-east-1 : US East (N. Virginia)
2) us-west-1 : US West (N. California)
3) us-west-2 : US West (Oregon)
4) eu-west-1 : EU (Ireland)
5) eu-central-1 : EU (Frankfurt)
6) ap-south-1 : Asia Pacific (Mumbai)
7) ap-southeast-1 : Asia Pacific (Singapore)
8) ap-southeast-2 : Asia Pacific (Sydney)
9) ap-northeast-1 : Asia Pacific (Tokyo)
10) ap-northeast-2 : Asia Pacific (Seoul)
11) sa-east-1 : South America (Sao Paulo)
12) cn-north-1 : China (Beijing)
13) cn-northwest-1 : China (Ningxia)
14) us-east-2 : US East (Ohio)
15) ca-central-1 : Canada (Central)
16) eu-west-2 : EU (London)
17) eu-west-3 : EU (Paris)
18) eu-north-1 : EU (Stockholm)
19) ap-east-1 : Asia Pacific (Hong Kong)
(default is 3): 1
```

```
Enter Application Name
(default is "Meetup"):
Application Meetup has been created.
```

```
It appears you are using Docker. Is this correct?
(Y/n): y
Cannot setup CodeCommit because there is no Source Control setup, continuing with initialization
Do you want to set up SSH for your instances?
(Y/n): n
```


CRIAÇÃO DO AMBIENTE

```
lucas.simao@suportelocal-Inspiron-5457:~/Área de Trabalho/Meetup$ eb create Meetup
```

```
Creating application version archive "app-190629 120557".
```

```
Uploading Meetup/app-190629 120557.zip to S3. This may take a while.
```

```
Upload Complete.
```

```
Environment details for: Meetup
```

```
Application name: Meetup
```

```
Region: us-east-1
```

```
Deployed Version: app-190629 120557
```

```
Environment ID: e-pw4ehszpse
```

```
Platform: arn:aws:elasticbeanstalk:us-east-1::platform/Docker running on 64bit Amazon Linux/2.12.14
```

```
Tier: WebServer-Standard-1.0
```

```
CNAME: UNKNOWN
```

```
Updated: 2019-06-29 15:06:02.494000+00:00
```

```
Printing Status:
```

```
2019-06-29 15:06:01 INFO createEnvironment is starting.
```

```
2019-06-29 15:06:02 INFO Using elasticbeanstalk-us-east-1-547872292024 as Amazon S3 storage bucket for environment data.
```

```
2019-06-29 15:06:28 INFO Created security group named: sg-0162b9ecc3cff7892
```

```
2019-06-29 15:06:43 INFO Created load balancer named: awseb-e-p-AWSEBLoa-3PTMGF9DWL2Y
```

```
2019-06-29 15:06:43 INFO Created security group named: awseb-e-pw4ehszpse-stack-AWSEBSecurityGroup-1A20ZT4X1EQN7
```

```
2019-06-29 15:06:43 INFO Created Auto Scaling launch configuration named: awseb-e-pw4ehszpse-stack-AWSEBAutoScalingLaunchConfiguration-1B0G8ZJVPRIGE
```

```
2019-06-29 15:07:47 INFO Created Auto Scaling group named: awseb-e-pw4ehszpse-stack-AWSEBAutoScalingGroup-WB93789AHK0W
```

```
2019-06-29 15:07:47 INFO Waiting for EC2 instances to launch. This may take a few minutes.
```

```
2019-06-29 15:07:47 INFO Created Auto Scaling group policy named: arn:aws:autoscaling:us-east-1:547872292024:scalingPolicy:d60134fe-3e81-4409-92a8-67b5b3113fc0:autoScalingGroupName/awseb-e-pw4ehszpse-stack-AWSEBAutoScalingGroup-WB93789AHK0W:policyName/awseb-e-pw4ehszpse-stack-AWSEBAutoScalingScaleDownPolicy-1EG0URWQK7KC
```

```
2019-06-29 15:07:47 INFO Created Auto Scaling group policy named: arn:aws:autoscaling:us-east-1:547872292024:scalingPolicy:815318b3-2690-4ef1-a24f-1621853ae230:autoScalingGroupName/awseb-e-pw4ehszpse-stack-AWSEBAutoScalingGroup-WB93789AHK0W:policyName/awseb-e-pw4ehszpse-stack-AWSEBAutoScalingScaleUpPolicy-N8W15NEV8PTX
```

```
2019-06-29 15:07:47 INFO Created CloudWatch alarm named: awseb-e-pw4ehszpse-stack-AWSEBCloudwatchAlarmHigh-MV0HR5L34KTR
```

```
2019-06-29 15:08:02 INFO Created CloudWatch alarm named: awseb-e-pw4ehszpse-stack-AWSEBCloudwatchAlarmLow-1Y23AVKSTMW47
```

```
2019-06-29 15:08:21 INFO Successfully pulled python:2.7-slim
```

```
2019-06-29 15:08:33 INFO Successfully built aws beanstalk/staging-app
```

```
2019-06-29 15:08:42 INFO Docker container 8a1229ecabf8 is running aws beanstalk/current-app.
```

```
2019-06-29 15:09:18 INFO Application available at Meetup.rcuz2ezm5z.us-east-1.elasticbeanstalk.com.
```

```
2019-06-29 15:09:18 INFO Successfully launched environment: Meetup
```


CRIAÇÃO DO DEPLOY

```
lucas.simao@suportelocal-Inspiron-5457:~/Área de Trabalho/Meetup$ eb deploy
Creating application version archive "app-190629 121438".
Uploading Meetup/app-190629 121438.zip to S3. This may take a while.
Upload Complete.
2019-06-29 15:14:42      INFO      Environment update is starting.
2019-06-29 15:14:45      INFO      Deploying new version to instance(s).
2019-06-29 15:14:51      INFO      Successfully built aws beanstalk/staging-app
2019-06-29 15:14:51      INFO      Successfully pulled python:2.7-slim
2019-06-29 15:15:10      INFO      Docker container 1de4ef3d7e3b is running aws beanstalk/current-app.
2019-06-29 15:15:16      INFO      New application version was deployed to running EC2 instances.
2019-06-29 15:15:16      INFO      Environment update completed successfully.

lucas.simao@suportelocal-Inspiron-5457:~/Área de Trabalho/Meetup$ eb open
```

PERGUNTAS?