

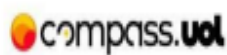
Documentação Atividade WordPress Docker
Aluno: Thiago Sandre Trevisan

Documento criado a partir da atividade prática,
Sobre Docker na AWS



Docker on AWS

Atividade AWS – Docker PB - FW - A - RG - SB - HA

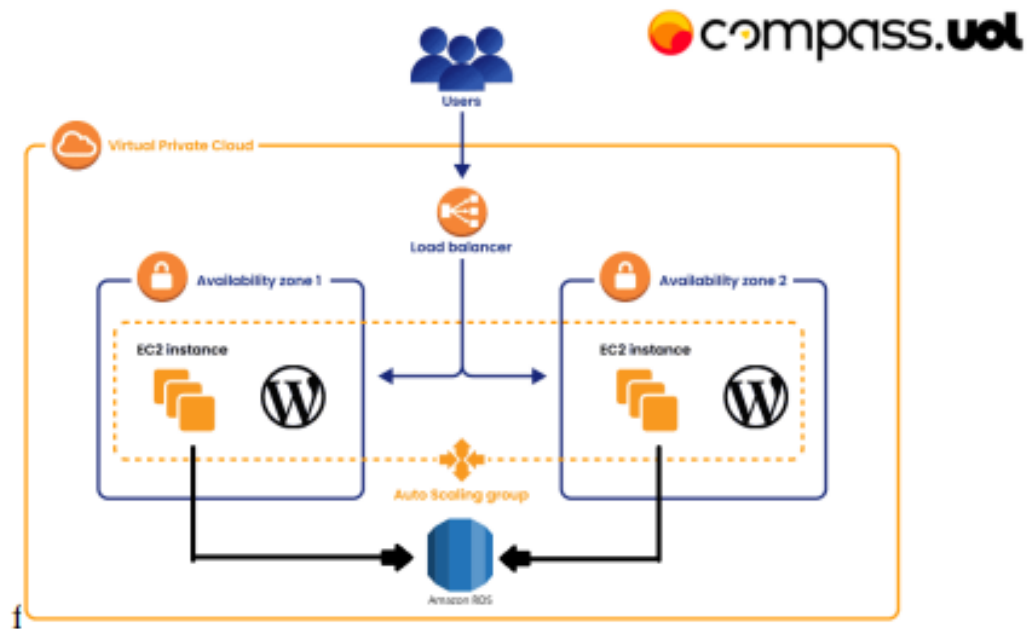


- [Assuntos:](#)

Data da entrega da atividade: * Repositório e documentação; Atenção: Enviar o LINK do repositório com código, versionamento e documentação até esse dia. Não serão permitidas modificações após a entrega.	30 de Outubro de 2023. Até as 14:00
Data da apresentação	31 de Outubro de 2023
Assunto	Atividade - DevSecOps - Docker
Participantes Compass	Fernando Meyer Marcos Antonio Soares Cassao Junior , Eduardo Lanius , Erick Mancz , Bruno Nicastro Ribeiro de Araujo , Paulo Junior Rothenbach

Arquitetura:

Arquitetura da Atividade proposta



Etapas a serem feitas dentro da atividade

Atividade - AWS - Docker	Docker	<p>Trabalho para fixar conhecimentos de DOCKER</p> <p>SEGUIR DESENHO TOPOLOGIA DISPOSTA.</p>	<ol style="list-style-type: none"> 1. instalação e configuração do DOCKER ou CONTAINERD no host EC2; 2. Efetuar Deploy de uma aplicação Wordpress com: container de aplicação RDS database Mysql 3. configuração da utilização do serviço EFS AWS para estáticos do container de aplicação Wordpress 4. configuração do serviço de Load Balancer AWS para a aplicação Wordpress <p>Pontos de atenção:</p> <ul style="list-style-type: none"> ○ não utilizar ip público para saída do serviços WP (Evitem publicar o serviço WP via IP Público) ○ sugestão para o tráfego de internet sair pelo LB (Load Balancer Classic) ○ pastas públicas e estáticos do wordpress sugestão de utilizar o EFS (Elastic File Sistem) ○ Fica a critério de cada integrante (ou dupla) usar Dockerfile ou Dockercompose; ○ Necessário demonstrar a aplicação wordpress funcionando (tela de login) ○ Aplicação Wordpress precisa estar rodando na porta 80 ou 8080; ○ Utilizar repositório git para versionamento; ○ Criar documentação.
---------------------------------	--------	---	--

VPC



VPC

ID da VPC vpc-0f93f6fc92c0a114a	Estado Available	Nomes de host DNS Habilitado	Resolução de DNS Habilitado
Localção Default	Conjunto de opções de DHCP dopt-0009f99fe0f820315	Tabela de rota principal rtb-03dfb625e03c2c912	Network ACL principal acl-0a5e196b0022beb21
VPC padrão Não	CIDR IPv4 10.0.0.0/24	Grupo IPv6 -	CIDR IPv6 (Grupo de borda de rede) -
Métricas de uso do endereço de rede Desabilitado	Grupos de regras do Firewall de DNS do resolvidor do Route 53 -	ID do proprietário 353703856889	

Vpc criada para a divisão de sub-nets e disponibilidade de internet para ambas, vinculada a uma Tabela de Rota e uma Network ACL.

Mapa Recursos



Mapa da VPC sobre como está configurada, onde mostra as sub-nets dentro da VPC (InstanciaDocker1, InstanciaDocker2) que estão vinculados a Tabela de Rotas para o trafego de rede para recursos que está vinculada a uma Network ACL.

CIDRs

Tipo de endereço	CIDR	Grupo de borda de rede	Grupo	Status
IPv4	10.0.0.0/24	-	-	Associated

IPv4 associado a VPC (10.0.0.0/24)

Subnets

InstanciaDocker1

Detalhes			
<div>ID da sub-rede</div> <div>subnet-0437c627748dcf173</div> <div>Endereços IPv4 disponíveis</div> <div>8</div> <div>Grupo de borda de rede</div> <div>us-east-1</div> <div>Sub-rede padrão</div> <div>Não</div> <div>Grupo de IPv4 de propriedade do cliente</div> <div>-</div> <div>Somente IPv6</div> <div>Não</div> <div>DNS64</div> <div>Desabilitado</div>	<div>ARN da sub-rede</div> <div>arn:aws:ec2:us-east-1:353703856889:subnet/subnet-0437c627748dcf173</div> <div>CIDR IPv6</div> <div>-</div> <div>VPC</div> <div>vpc-0f93f6fc92c0a114a AtividadeDocker</div> <div>Atribuir endereço IPv4 público automaticamente</div> <div>Não</div> <div>ID do Outpost</div> <div>-</div> <div>Tipo de nome de host</div> <div>Nome do IP</div> <div>Proprietário</div> <div>353703856889</div>	<div>Estado</div> <div>Available</div> <div>Zona de disponibilidade</div> <div>us-east-1a</div> <div>Tabela de rotas</div> <div>rtb-03dfb625e03c2c912 AtividadeDocker</div> <div>Atribuir endereço IPv6 automaticamente</div> <div>Não</div> <div>Reservas CIDR IPv4</div> <div>-</div> <div>Registro DNS A do nome do recurso</div> <div>Desabilitado</div>	<div>CIDR IPv4</div> <div>10.0.0.0/28</div> <div>ID de zona de disponibilidade</div> <div>use1-az6</div> <div>Network ACL</div> <div>acl-0a5e196b0022beb21</div> <div>Atribuir automaticamente endereço IPv4 de propriedade do cliente</div> <div>Não</div> <div>Reservas CIDR IPv6</div> <div>-</div> <div>Registro DNS AAAA do nome do recurso</div> <div>Desabilitado</div>

Subnet1 criada e vinculada a primeira instância criada na zona de disponibilidade us-east-1a com um CIDR (10.0.0.0/28), vinculada a Tabela de rotas para o trafego de rede para recursos e a VPC (AtividadeDocker).

InstânciaDocker2

Detalhes			
<div>ID da sub-rede</div> <div>subnet-0f0d73399669057aa</div> <div>Endereços IPv4 disponíveis</div> <div>7</div> <div>Grupo de borda de rede</div> <div>us-east-1</div> <div>Sub-rede padrão</div> <div>Não</div> <div>Grupo de IPv4 de propriedade do cliente</div> <div>-</div> <div>Somente IPv6</div> <div>Não</div> <div>DNS64</div> <div>Desabilitado</div>	<div>ARN da sub-rede</div> <div>arn:aws:ec2:us-east-1:353703856889:subnet/subnet-0f0d73399669057aa</div> <div>CIDR IPv6</div> <div>-</div> <div>VPC</div> <div>vpc-0f93f6fc92c0a114a AtividadeDocker</div> <div>Atribuir endereço IPv4 público automaticamente</div> <div>Não</div> <div>ID do Outpost</div> <div>-</div> <div>Tipo de nome de host</div> <div>Nome do IP</div> <div>Proprietário</div> <div>353703856889</div>	<div>Estado</div> <div>Available</div> <div>Zona de disponibilidade</div> <div>us-east-1b</div> <div>Tabela de rotas</div> <div>rtb-03dfb625e03c2c912 AtividadeDocker</div> <div>Atribuir endereço IPv6 automaticamente</div> <div>Não</div> <div>Reservas CIDR IPv4</div> <div>-</div> <div>Registro DNS A do nome do recurso</div> <div>Desabilitado</div>	<div>CIDR IPv4</div> <div>10.0.0.16/28</div> <div>ID de zona de disponibilidade</div> <div>use1-az1</div> <div>Network ACL</div> <div>acl-0a5e196b0022beb21</div> <div>Atribuir automaticamente endereço IPv4 de propriedade do cliente</div> <div>Não</div> <div>Reservas CIDR IPv6</div> <div>-</div> <div>Registro DNS AAAA do nome do recurso</div> <div>Desabilitado</div>

Subnet2 criada e vincula a segunda instância criada na zona de disponibilidade us-east-1B com um CIDR (10.0.0.16/28), vinculada a Tabela de rotas juntamente a primeira subnet para o trafego de rede para recursos e a VPC (AtividadeDocker).



Route Tables AtividadeDocker

Detalhes			
ID da tabela de rotas rtb-03dfb625e03c2c912	Principal Sim	Associações explícitas de sub-rede 2 sub-redes	Associações de borda -
VPC vpc-0f93f6fc92c0a114a AtividadeDocker	ID do proprietário 353703856889		

Vinculada a VPC (AtividadeDocker) mais associação das duas subnets existentes e a um Internet Gateway para acesso a internet.

Rotas (2)			
<input type="text" value="Filtrar rotas"/>			
Destino	Alvo	Status	Propagado
0.0.0.0/0	igw-03a174c427927cb40	Ativo	Não
10.0.0.0/24	local	Ativo	Não

Rotas vinculadas a Route Table, tendo um Gateway de Internet vinculado e um ip local.

Gatway de Internet

igw-03a174c427927cb40 / AtividadeDockerVPC			
Detalhes Tags			
Detalhes			
ID do gateway da Internet igw-03a174c427927cb40	Estado Attached	ID da VPC vpc-0f93f6fc92c0a114a AtividadeDocker	Proprietário 353703856889

Criado com a função de fornecer internet para VPC e outras funções que viriam ser vinculadas com a VPC.



Security Groups InstanciasDocker

Detalhes			
Nome do grupo de segurança InstanciasDocker	ID do grupo de segurança sg-0d9942f0b624d1359	Descrição Regras para Instancias se conectarem	ID da VPC vpc-0f93f6fc92c0a114a
Proprietário 353703856889	Número de regras de entrada 5 Entradas de permissão	Número de regras de saída 1 Entrada de permissão	

Criada com o objetivo de fornecer portas e acesso para as seguintes funções:

Regras de entrada (5)								
Filtrar regras de grupo de segurança								
<input type="checkbox"/>	Name	ID da regra do grup...	Versão do IP	Tipo	Protocolo	Intervalo de portas	Origem	
<input type="checkbox"/>	-	sgr-0baa06ef1a8c0a957	IPv4	NFS	TCP	2049	0.0.0.0/0	
<input type="checkbox"/>	-	sgr-00c9a7a9be3f70874	IPv4	SSH	TCP	22	0.0.0.0/0	
<input type="checkbox"/>	-	sgr-0d99cb54f05c7d8c4	IPv4	HTTPS	TCP	443	0.0.0.0/0	
<input type="checkbox"/>	-	sgr-017023e721d988...	IPv4	HTTP	TCP	80	0.0.0.0/0	
<input type="checkbox"/>	-	sgr-0979469a6174dcd...	IPv4	MYSQL/Aurora	TCP	3306	0.0.0.0/0	

SSH: Libera o acesso a instancia a partir do chave gerada (AtividadeDocker.pem).

NFS: Libera o acesso a função EFS, para que assim ambas as instâncias pudessem acessar um arquivo compartilhado pela função da própria AWS (EFS).

HTTPS: Libera o acesso para o navegador para que a conexão seja criptografada .

HTTP: Libera o acesso ao navegador para a comunicação da aplicação com o navegador.

MYSQL/Aurora: Libera o acesso ao banco de dados criado pelo RDS.



EFS

DockerEFS

Geral

Editar

Modo de desempenho

Uso geral

Modo de taxa de transferência

Elastic

Gerenciamento de ciclo de vida

Transição para IA: 30 dias desde o último acesso

Transição fora do IA: Nenhum

Zona de disponibilidade

Padrão

Backups automáticos

Habilitado

Criptografado

Não

Estado do sistema de arquivos

Disponível

Nome de DNS

fs-0192f5e1fd9b4b2f2.efs.us-east-1.amazonaws.com

Tamanho medido

Monitoramento

Tags

Política do sistema de arquivos

Pontos de acesso

Rede

Replicação

Rede

Gerenciar

Criado para a compartilhar arquivos dentro de uma pasta especificada pelo usuário, vinculado as duas subnets existentes para ter acesso a endereço de IPV4 de cada subnet fazendo com que assim possa ser gerado uma conexão para ambas as instâncias.

Modelo de Execução das instâncias (Auto Scaling)

Detalhes da versão do modelo de execução

Ações ▼Excluir versão de modelo

Versão 1 (padrão) ▼	Descrição Modelo Para Auto Scaling	Data de criação 2023-10-27T15:23:03.000Z	Criado por arn:aws:sts::353703856889:assumed-role/AWSReservedSSO_AdministratorAccess_49de29456505890b/tiago.trevisan.pb@compasso.com.br
------------------------	---------------------------------------	---	--

Detalhes da instância

Armazenamento

Tags de recurso

Interfaces de rede

Detalhes avançados


ID da AMI ami-0dbc3d7bc646e8516	Tipo de instância t3.small	Zona de disponibilidade -	Nome do par de chaves AtividadeDocker
Grupos de segurança -	IDs de grupo de segurança sg-0d9942f0b624d1359		

Modelo de execução criado com a função de iniciar as instâncias dentro do auto scaling, com o tipo de instância t3.small, 16G de armazenamento, Chave SSH, Security Group *Na imagem não possui um security group, mas ao criar a instância a partir do auto scaling ele atribui o security group configurado para o modelo de execução*.






Auto Scaling

Nome: DockerAutoScaling

Detalhes do grupo			Editar
Nome do grupo do Auto Scaling DockerAutoScaling	Capacidade desejada 2	Status -	Nome de recurso da Amazon (ARN)  <code>arn:aws:autoscaling:us-east-1:353703856889:autoScalingGroup:c1c40baf-649c-4d9e-8e76-bb018ace7c11:autoScalingGroupName/DockerAutoScaling</code>
Data de criação Fri Oct 27 2023 13:34:09 GMT-0400 (Hora padrão do Amazonas)	Capacidade mínima 2		
	Capacidade máxima 2		

Detalhe sobre a capacidade desejada sobre o auto Scaling de 2 para 2.

Modelo de Execução Auto Scaling

Modelo de execução  <code>lt-0cd9a456b163687c9</code> DockerEXEC	ID da AMI  <code>ami-0dbc3d7bc646e8516</code>	Tipo de instância t3.small	Proprietário <code>arn:aws:sts::353703856889:assumed-role/AWSReservedSSO_AdministratorAccess_49de29456505890b/thiago.trevisan.pb@compasso.com.br</code>
Versão Default	Grupos de segurança -	IDs de grupo de segurança  <code>sg-0d9942f0b624d1359</code>	Horário de criação Fri Oct 27 2023 11:23:03 GMT-0400 (Hora padrão do Amazonas)
Descrição Modelo Para Auto Scaling	Armazenamento (volumes) <code>/dev/xvda</code>	Nome do par de chaves AtividadeDocker	Solicitar instâncias spot Não

Rede Auto Scaling

Rede	
Zonas de disponibilidade us-east-1a, us-east-1b	ID da sub-rede <code>subnet-0f0d73399669057aa, subnet-0437c627748dcf173</code>

Duas zonas de disponibilidade 1a e 1b e duas subnets para cada zona, fazendo com que assim cada instância esteja em uma zona diferente assim como a atividade pediu.

Tags Auto Scaling

Etiquetas (3)		
Chave	Valor	Etiquetar novas instâncias
CostCenter	C092000004	Sim
Name	PB - FW - A - RG - SB - HA	Sim
Project	PB - FW - A - RG - SB - HA	Sim



Load Balancer Detalhes

Balanceador de carga: DockerLoadBalancer

Tipo de balanceador de carga Application	Status Ativo	VPC vpc-0f93f6fc92c0a114a	Tipo de endereço IP IPv4
Esquema Internet-facing	Zona hospedada Z35SXDOTRQ7X7K	Zonas de disponibilidade subnet-0f0d73399669057aa us-east-1b (use1-az1) subnet-0437c627748dcf173 us-east-1a (use1-az6)	Data de criação 27 de outubro de 2023, 13:34 (UTC-04:00)
ARN do load balancer <code>arn:aws:elasticloadbalancing:us-east-1:353703856889:loadbalancer/ap p/DockerLoadBalancer/4e7c3a2948ab9e55</code>		Nome do DNS Informações <code>DockerLoadBalancer-279261874.us-east-1.elb.amazonaws.com</code> (Registro A)	

Tipo de Load Balancer: Application.
Tipo de Endereço: IPV4.
Zonas de Disponibilidade: 1A e 1B.
Esquema: Internet-facing.
VPC: AtividadeDocker.

Mapeamento de Rede

[vpc-0f93f6fc92c0a114a](#)

IPv4: 10.0.0.0/24

IPv6 :-

IPv4

Mapeamentos

Incluir duas ou mais zonas de disponibilidade e sub-redes correspondentes aumenta a tolerância a falhas de suas aplicações.

Zona	Sub-rede	Endereço IPv4	Endereço IPv4 privado
us-east-1b (use1-az1)	subnet-0f0d73399669057aa	Atribuído pela AWS	Atribuído do CIDR 10.0.0.16/28
us-east-1a (use1-az6)	subnet-0437c627748dcf173	Atribuído pela AWS	Atribuído do CIDR 10.0.0.0/28

IPv4:10.0.0.0/24 (mesmo da VPC).

Zonas 1A (10.0.0.0/28) e 1B (10.0.0.16/28).

Onde ele redireciona para ambas as zonas o acesso do cliente na aplicação.

Security Groups Load Balancer

ID do grupo de segurança	Nome	Descrição
sg-0d9942f0b624d1359	InstanciasDocker	Regras para Instancias se conectarem

Mesmo Security Group que é usado para as instâncias

Tags Load Balancer

Chave	Valor
CostCenter	C092000004
Name	PB - FW - A - RG - SB - HA
Project	PB - FW - A - RG - SB - HA



RDS (Banco Dados Mysql)

Segurança e conexão

Endpoint e porta

Endpoint

[REDACTED]

Porta

3306

Redes

Zona de disponibilidade
us-east-1b

VPC

[AtividadeDocker \(vpc-0f93f6fc92c0a114a\)](#)

Grupo de sub-redes
default-vpc-0f93f6fc92c0a114a

Sub-redes
[subnet-0f0d73399669057aa](#)
[subnet-0437c627748dcf173](#)

Tipo de rede
IPv4

Segurança

Grupos de segurança da VPC

[InstanciasDocker \(sg-0d9942f0b624d1359\)](#)

✓ Ativo

Publicamente acessível

Não





Autoridade de
certificação [Informações](#)
rds-ca-2019

Data da autoridade de certificado
August 22, 2024, 13:08 (UTC-04:00)

Data de expiração do certificado
da instância de banco de dados
⚠ August 22, 2024, 13:08 (UTC-04:00)

Tipo de Rede: IPV4
Security Group: InstanciasDocker.
Porta: 3306.
Sub-nets: 1A e 1B .
Zona de Disponibilidade: 1B .
VPC: AtividadeDocker

Configuração

Instância			
Configuração ID da instância de banco de dados mysqldocker Versão do mecanismo 8.0.33 Nome do banco de dados mysqldocker Modelo de licença General Public License Grupos de opções default:mysql-8-0  Em sincronia Nome de recurso da Amazon (ARN)  arn:aws:rdsus-east-1:353703856889:db:mysqldocker ID do recurso db-TAPNKXZAKHAPIA2NGZCE4BAITQ Horário de criação October 27, 2023, 14:04 (UTC-04:00) Grupo de parâmetros de instância de banco de dados default:mysql8.0  Em sincronia Proteção contra exclusão Desabilitado	Classe de instância Classe de instância db.t3.micro vCPU 2 RAM 1 GB Disponibilidade Nome do usuário principal root Senha principal ***** Autenticação do banco de dados do IAM Não habilitado Multi-AZ Não Zona secundária -	Armazenamento Criptografia Habilitado Chave do AWS KMS aws/rds  Tipo de armazenamento SSD de uso geral (gp2) Armazenamento 20 GiB IOPS provisionadas - Taxa de throughput - Escalabilidade automática do armazenamento Habilitado Limite máximo de armazenamento 1000 GiB Configuração do sistema de arquivos de armazenamento Atual	Performance Insights Performance Insights habilitado Desativado

Criado no nível gratuito vinculado a VPC mais as duas subnets existentes em zonas de disponibilidade diferentes (1A e 1B), com um armazenamento de 20GB e com o intuito de servir como base para a conexão mais a instalação do WordPress.



Instalação Mysql Instancia

```
C:\ root@ip-10-0-0-5:/home/ec2-user
[root@ip-10-0-0-5 ec2-user]# ls
EFS  compose  mysql
[root@ip-10-0-0-5 ec2-user]#
```

Primeiro eu criei uma pasta pro mysql usando o comando “mkdir mysql”

```
C:\ root@ip-10-0-0-5:/home/ec2-user/mysql
[root@ip-10-0-0-5 mysql]# ls
mysql57-community-release-el7-8.noarch.rpm
[root@ip-10-0-0-5 mysql]#
```

Dentro dele eu executei o comando

“sudo rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2022
wget <http://dev.mysql.com/get/mysql57-community-release-el7-8.noarch.rpm>”.
Instalando assim um executavel do Mysql-community

Em seguida executo o comando

“sudo yum localinstall -y mysql57-community-release-el7-8.noarch.rpm”.

Para baixar o pacote

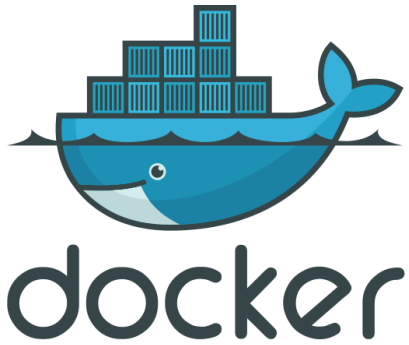
Logo depois executo a biblioteca instalada na maquina

“sudo yum install -y mysql-community-server”.

Depois eu só preciso iniciar o mysql e habilitar ele dentro da instância

“sudo systemctl start mysqld”.

“sudo systemctl enable mysqld”.



Instalação Docker Instancia

Executo o comando para a instalação do docker
“yum install docker -y”.

Depois eu starto a biblioteca instalada
“systemctl start docker”.

Então deixo ele ativo para que toda vez em que a instância for iniciada ele execute
o programa.
systemctl enable docker



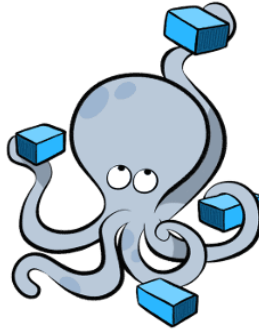
Montagem Do EFS

Executo o comando vim /etc/fstab para acessar as montagens automáticas e
adiciono a seguinte linha para deixar a mount automática.

“fs-0192f5e1fd9b4b2f2.efs.us-east-1.amazonaws.com:/ /home/ec2-user/EFS nfs4
nfsvers=4.1,rsz=1048576,wsz=1048576,hard,timeo=600,retrans=2,noresvport,_
netdev 0 0”.

```
UUID=b00088a-b0f2-46e6-be9a-8c6c16f40ce3 / xfs defaults,noatime 1 1
UUID=f415-3516 /boot/efi vfat defaults,noatime,uid=0,gid=0,umask=0077,shortname=winnt,x-systemd.automount 0 2
fs-0192f5e1fd9b4b2f2.efs.us-east-1.amazonaws.com:/ /home/ec2-user/EFS nfs4 nfsvers=4.1,rsz=1048576,wsz=1048576,hard,timeo=600,retrans=2,noresvport,_netdev 0 0
```

Onde eu determino id do EFS criado especifico o local da montagem



Docker-Compose
Executo o Comando

“sudo curl -L

[https://github.com/docker/compose/releases/download/1.22.0/docker-compose-\\$\(uname -s\)-\\$\(uname -m\) -o /usr/local/bin/docker-compose](https://github.com/docker/compose/releases/download/1.22.0/docker-compose-$(uname -s)-$(uname -m) -o /usr/local/bin/docker-compose)”.

Baixo o compose via git-hub e dou permissão para ele se tornar executável.

“sudo chmod +x /usr/local/bin/docker-compose”.

Arquivo docker-compose.yml

```
root@ip-10-0-0-5:/home/ec2-user/compose
version: '3.7'
services:
  wordpress:
    image: wordpress
    volumes:
      - /home/ec2-user/EFS/wordpress:/var/www/html
    ports:
      - 80:80
    restart: always
    environment:
      WORDPRESS_DB_HOST: [REDACTED]:3306
      WORDPRESS_DB_USER: root
      WORDPRESS_DB_PASSWORD: wordpress
      WORDPRESS_DB_NAME: mysqldocker
      WORDPRESS_TABLE_CONFIG: wp_
```

Especifico a versão do serviço

Image: wordpress (especifica a imagem da aplicação).

Volumes: /home/ec2-user/EFS/wordpress:/var/www/HTML (Falo pra ele jogar todos os arquivos do wordpress dentro do diretório especificado na montagem).

Ports: Especifico a porta onde aplicação será exposta.

Enviroment: Onde é especificado as variáveis para o acesso ao banco de dados (RDS)

Subindo aplicação

“docker-compose up -d”

```
[root@ip-10-0-0-26 compose]# docker-compose up -d
Creating network "compose_default" with the default driver
Pulling wordpress (wordpress:...)...
latest: Pulling from library/wordpress
e67fdae35593: Pull complete
22a19ba793cb: Pull complete
46e419350351: Pull complete
8de997c28946: Pull complete
1d7540f99b47: Pull complete
12a700ba0368: Pull complete
9b19829d9fc5: Pull complete
d44a7be08f84: Pull complete
84073d869176: Pull complete
6624316b1353: Pull complete
81ea0c762078: Pull complete
68680de1e42a: Pull complete
0c351c6a1d3f: Pull complete
3f6f46332f1b: Pull complete
91a39b0e19bd: Pull complete
9d17674da2dd: Pull complete
b6c1fd663115: Pull complete
076ed9c917d2: Pull complete
f43f1a40f0fe: Pull complete
c42c0b9ab38f: Pull complete
a9e97aff2b31: Pull complete
Digest: sha256:549987ef8d878e6a9f360ee0cc033f03ca68c1e86aa9af7a206bdae4a98b0486
Status: Downloaded newer image for wordpress:latest
Creating compose_wordpress_1 ... done
```

Ele irá buscar a versão mais recente do wordpress baixar ele e subir e deixar a aplicação exposta na porta 80

Depois execute o comando docker ps, para verificar o estado da aplicação

```
[root@ip-10-0-0-26 compose]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
2c0eeff0bbcb	wordpress	"docker-entrypoint.s..."	30 seconds ago	Up 28 seconds	0.0.0.0:80->80/tcp, :::80->80/tcp	compose_wordpress_1

Usando o link do Load Balancer

Executo o comando para se conectar com o banco de dados

"mysql -h mysqldocker.cyvuxrlwstdo.us-east-1.rds.amazonaws.com -u root -p"

```
[root@ip-10-0-0-26 compose]# mysql -h mysqldocker.cyvuxrlwstdo.us-east-1.rds.amazonaws.com -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 967
Server version: 8.0.33 Source distribution

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Dentro do banco de dados eu vou atrás da database do wordpress

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| mysqldocker |
| performance_schema |
| sys |
+-----+
5 rows in set (0.01 sec)
```

Achando a database eu uso o comando “use mysqldocker;” e entro na database

```
mysql> use mysqldocker;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
```

Dentro da database eu uso o comando “show tables;” para procurar uma tabela chamada wp_options

```
mysql> show tables;
+-----+
| Tables_in_mysqldocker |
+-----+
| wp_commentmeta |
| wp_comments |
| wp_links |
| wp_options |
| wp_postmeta |
| wp_posts |
| wp_term_relationships |
| wp_term_taxonomy |
| wp_termmeta |
| wp_terms |
| wp_usermeta |
| wp_users |
+-----+
12 rows in set (0.00 sec)
```

Achando a tabela eu executo o comando “SELECT * FROM wp_options WHERE option_name = 'siteurl';” para procurar a url do wordpress

```
mysql> SELECT * FROM wp_options WHERE option_name = 'siteurl';
```

option_id	option_name	option_value	autoload
1	siteurl	http://DockerLoadBalancer-279261874.us-east-1.elb.amazonaws.com	yes

```
1 row in set (0.00 sec)
```

Encontrando a url do site eu executo novamente outro comando para alterar para a URL do Load Balancer “UPDATE wp_options SET option_value = 'http://DockerLoadBalancer-279261874.us-east-1.elb.amazonaws.com' WHERE option_name = 'siteurl';”

```
mysql> UPDATE wp_options SET option_value = 'http://DockerLoadBalancer-279261874.us-east-1.elb.amazonaws.com' WHERE option_name = 'siteurl';
Query OK, 0 rows affected (0.01 sec)
Rows matched: 1  Changed: 0  Warnings: 0
```

```
mysql> SELECT * FROM wp_options WHERE option_name = 'siteurl';
```

option_id	option_name	option_value	autoload
1	siteurl	http://DockerLoadBalancer-279261874.us-east-1.elb.amazonaws.com	yes

```
1 row in set (0.00 sec)
```

Após fazer tudo isso é apenas jogar o link do Load Balancer no Google e verificar se tudo está funcionando corretamente



Mindblown: a blog about philosophy.

Olá, mundo!

Boas-vindas ao WordPress. Esse é o seu primeiro post. Edite-o ou exclua-o, e então comece a escrever!

27 de outubro de 2023

Tem alguma recomendação de livro?

Entre em contato