

Filipe Borba e Gabriel Moreira
R2 - Deployment Orchestration

Instalando Juju

1) Qual o S.O. utilizado na máquina Juju? Quem o instalou?

O Sistema Operacional utilizado na máquina Juju é o Ubuntu 18.04 LTS, instalado pelo MaaS.

2) O programa juju client roda aonde? E o juju service? Como eles interagem entre si?

O Juju client está rodando no MaaS, pois é nele que estamos inserindo os comandos e recebendo respostas. Ele poderia ser instalado em qualquer máquina, desde que fosse configurado corretamente para se comunicar com o Juju service. Por sua vez, o service está localizado na máquina do Juju no nosso rack. Essa instância é responsável por receber os comandos do Juju client e fazer as requisições para o MaaS a fim de fazer os deploys e outras atividades.

3) O que é LXC? e LXD?

LXC (ou *Linux Containers*) é uma virtualização em nível de sistema operacional que permite rodar vários sistemas Linux (containers) em apenas um Kernel. Com ele, é possível que usuários de Linux criem e administrem containers de aplicação ou sistema.

O LXD é um sistema de gerenciamento de containers construído em volta de uma REST API. Ele oferece uma experiência similar à de máquinas virtuais, mas utilizando LXCs.

Deploying Wordpress com Load Balancing

4) Explique o conceito por trás do HAProxy (Reverse Proxy). Vocês já fizeram algo parecido?

O HAProxy é um software open-source que funciona como proxy server e, ao mesmo tempo, balanceia requisições em diferentes servers. No roteiro 0, tivemos contato com o LoadBalancer da Amazon Web Services, que possui a mesma função de distribuição do

tráfego da aplicação em várias instâncias a fim de evitar o congestionamento de tais endereços.

5) Na instalação, o Juju alocou automaticamente 4 máquinas físicas, duas para o Wordpress, uma para o Mysql e uma para o HAProxy. Considerando que é um Hardware próprio, ao contrário do modelo Public Cloud, isso é uma característica boa ou ruim?

Visto que estamos num ambiente com um número limitado de máquinas e o Juju utilizou todas para que nossa aplicação funcionasse, podemos concluir que é uma característica ruim do Juju nesse sentido. Mesmo num ambiente de Public Cloud, a utilização de diversas máquinas acarretaria num custo maior, também sendo prejudicial.

6) Crie um roteiro de implantação do Wordpress no seu hardware sem utilizar o Juju.

1) Instalar apache web server

```
sudo apt-get install apache2 apache2-utils
```

```
sudo systemctl enable apache2
```

```
sudo systemctl start apache2
```

2) Instalar MySQL server

```
sudo apt-get install mysql-client mysql-server
```

```
sudo mysql_secure_installation
```

3) Instalar PHP e módulos

```
sudo apt-get install php7.0 php7.0-mysql libapache2-mod-php7.0
```

```
php7.0-cli php7.0-cgi php7.0-gd
```

Vá até o diretório do Apache (**/var/www/html**) e crie um arquivo **info.php**. Copie e cole o código abaixo para o arquivo.

```
<?php  
    phpinfo();  
?>
```

4) Instalar Wordpress CMS

```
swget -c http://wordpress.org/latest.tar.gz
```

```
tar -xzf latest.tar.gz
```

Mova os arquivos do diretório do wordpress (que foi baixado) para a pasta `/var/www/html/`:

```
sudo rsync -av wordpress/* /var/www/html/
```

Configure as permissões do diretório:

```
sudo chown -R www-data:www-data /var/www/html/
```

```
sudo chmod -R 755 /var/www/html/
```

5) Criar WordPress Database

```
mysql -u root -p
```

```
mysql> CREATE DATABASE wp_myblog;
```

```
mysql> GRANT ALL PRIVILEGES ON wp_myblog.* TO  
'your_username_here'@'localhost' IDENTIFIED BY  
'your_chosen_password_here';
```

```
mysql> FLUSH PRIVILEGES;
```

```
mysql> EXIT;
```

Renomeie o arquivo `wp-config-sample.php` (no diretório `/var/www/html/`) para `wp-config.php`:

```
sudo mv wp-config-sample.php wp-config.php
```

Complete o arquivo acima com as informações do seu database:

```
// ** MySQL settings - You can get this info from your web host ** //  
/** The name of the database for WordPress */  
define('DB_NAME', 'database_name_here'); /** MySQL database username */  
define('DB_USER', 'username_here'); /** MySQL database password */  
define('DB_PASSWORD', 'password_here'); /** MySQL hostname */  
define('DB_HOST', 'localhost'); /** Database Charset to use in creating  
database tables. */ define('DB_CHARSET', 'utf8'); /** The Database Collate type.  
Don't change this if in doubt. */ define('DB_COLLATE', '');
```

Reinicie o web server e o mysql:

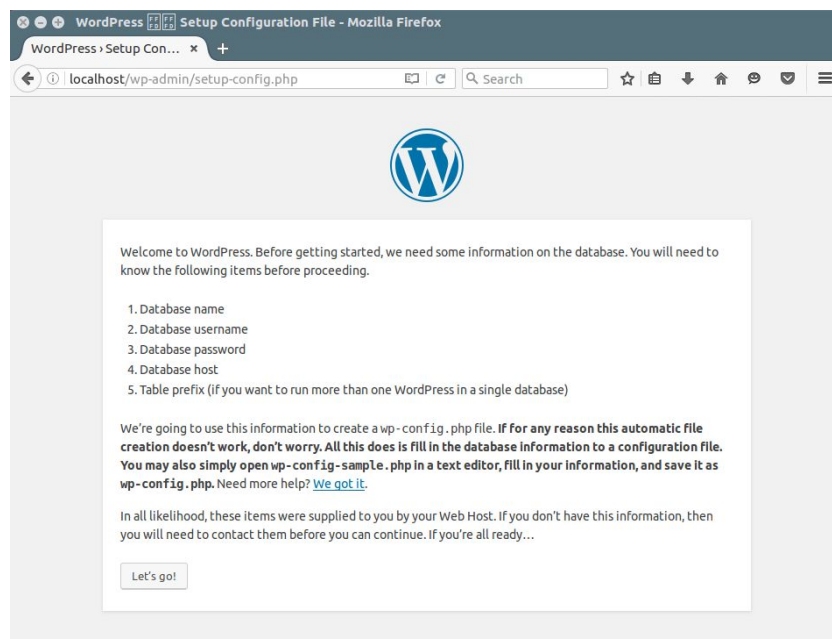
```
sudo systemctl restart apache2.service
```

```
sudo systemctl restart mysql.service
```

6) Últimos passos :

Abra seu browser na página <http://server-address:>

O WordPress deve estar instalado!



Questões Complementares

1) Juju é uma aplicação distribuída? E o MaaS?

O Juju, assim como o MaaS, não é uma aplicação distribuída, porque ele roda em apenas uma máquina na nossa rede.

2) Qual a diferença entre REST e RPC?

RPC (Remote Procedure Call) é uma tecnologia de comunicação entre processos que permite que um programa chame um procedimento em outro computador conectado por uma rede. Já o REST (Representational State Transfer) é um estilo de arquitetura que define um conjunto de restrições e propriedades baseados em HTTP. O RPC é uma

tecnologia popular para a implementação do modelo cliente-servidor, funcionando através de uma chamada de procedimento remoto. O cliente inicia a chamada enviando uma mensagem para um servidor remoto para executar um procedimento específico, assim, uma resposta é retornada ao cliente. O REST serve para especificar os princípios dessa comunicação entre cliente e servidor, indicando boas práticas, como operações bem definidas (GET, POST, PUT, DELETE), por exemplo.

3) O que é SOAP?

SOAP (ou Simple Object Access Protocol) é um protocolo que permite a troca de informação entre servidores via serialização. É baseado em XML e pode ser transacionado por diversos protocolos, mas geralmente o protocolo HTTP é utilizado. Este protocolo é independente de plataforma e foi projetado para ser independente de qualquer modelo de programação ou outra implementação específica; apesar disso, tem um overhead grande de informação devido ao formato XML (Pacote + Header + Body).

Concluindo

1. O que é e o que faz um Deployment Orchestrator? Cite alguns exemplos.

Um Deployment Orchestrator, como o Juju, é um software responsável por fazer deploys, monitorar, configurar e escalar infraestruturas de cloud de forma fácil e rápida. Com ele, é possível criar *charms*, ou seja, um conjunto de comandos para a realização do deploy, deixando a operação muito mais rápida para quem trabalha. Por sua vez, é possível construir um *bundle de charms*, que pode fazer o deploy de vários serviços de uma vez, construindo a infraestrutura de sua aplicação em apenas um comando. Ainda, é possível especificar o uso das máquinas para que o Juju aloque os recursos de infraestrutura automaticamente.

2. Como é o processo de interação entre o MaaS e o Juju?

O MaaS é responsável por fornecer a rede, as máquinas, o monitoramento das máquinas e as imagens de OS para que o Juju possa fazer seu trabalho. O Juju fará requisições ao MaaS para que o deploy seja possível, então ele vai pedir a máquina, pedir a instalação do Ubuntu, etc. Após isso, o Juju monitora o estado das máquinas através do MaaS e administra o estado das aplicações dentro de cada VM. Podemos dizer que o MaaS cuida da infraestrutura para que o Juju possa operar.

3. Defina Aplicação Distribuída, Alta Disponibilidade e Load Balancing?

Aplicação Distribuída: é uma aplicação modular que possui módulos em diferentes computadores porém rodam como se fossem apenas uma aplicação. Uma vantagem desse tipo de aplicação é seu poder de processamento distribuído.

Alta Disponibilidade: um sistema de alta disponibilidade é um sistema resistente a falhas de software e hardware. Quando uma máquina falha, por algum motivo, outra assume o seu lugar para que o sistema fique ativo o máximo tempo possível. Um exemplo disso é o produto da AWS chamado Amazon S3 que cuida de armazenamento dos dados e possui uma resiliência de 99,999999999%.

Load Balancing: é a capacidade de distribuição de carga em diferentes máquinas a fim de evitar o congestionamento de uma única ou até mesmo maximizar o desempenho da aplicação/sistema.

Conclusão: O Juju utilizou o MaaS como provedor de recursos. O MaaS por sua vez forneceu o que havia disponível no rack. Você acha que seria necessária uma máquina de 32Gb para rodar um Apache Webserver ou um Load Balancer? Extrapole a resposta para um Datacenter real, onde as máquinas possuem configurações muito superiores. Como resolver esse problema?

Com certeza não seria necessário uma máquina de 32Gb de RAM para rodar um Apache Webserver ou um Load Balancer, visto que são aplicações simples. Isso ocorreu porque o Juju apenas utilizou as máquinas disponíveis, sem se preocupar com divisões na mesma máquina ou com as especificações da mesma. Para resolver este problema, seria necessário especificar as máquinas para as quais queremos fazer deploys ou utilizar o Openstack para que ele possa gerenciar os recursos disponíveis em nossa infraestrutura (como por exemplo, mais de uma VM em uma máquina).