

Handout 2 - Deployment Orchestration

Computação em Nuvem

Isabella Rocha de Oliveira

Lucas Chen Alba

Inspire 2018.2

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

O sistema operacional é o Linux Ubuntu 18. Nós demos o comando para a maas e ela instalou.

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

O juju client roda na maas, enquanto o juju service roda na própria máquina do juju. Os serviços se comunicam com o client e vice-versa através de uma API REST, o juju server envia os requerimentos para o mass, e este responde com os parâmetros solicitados.

3. O que é LXC? e LXD?

LXC é um container que roda no Kernel do Linux. Ele possibilita a execução de diversos softwares no ambiente virtual, como uma máquina virtual porém ele roda sob o nível de sistema operacional do Linux e tem como vantagem ser expansivo (tanto a memória quanto o poder de processamento são utilizados sob demanda) enquanto as máquinas virtuais rodam sobre o metal e tem suas especificações fixas uma vez que iniciadas.

LXD é uma extensão do LXC, como se fosse uma REST API da biblioteca do código do LXC. Ela cria um sistema com processos em plano de fundo que podem ser acessados por apps usando um socket Unix ou pela rede via HTTPS.

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

O HAProxy é um load balancer com alta disponibilidade e um proxy server que pode ser usado como reverse proxy para aplicações baseadas em TCP e HTTP e ele distribui para diversos servidores de forma rápida e eficiente em termos de processamento e utilização de memória. Um servidor proxy tem clients associados e pode entrar em contato com qualquer server, já o reverse proxy tem servers associados e pode ser contatado por qualquer client. Já fizemos algo parecido quando usamos o load balancer da Amazon, ele distribuía tarefas diferentes a diversas instâncias/máquinas, sendo que qualquer uma poderia realizar qualquer tarefa.

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?

Dependendo da aplicação pode ser uma característica boa ou ruim. Caso as máquinas já tiverem sido compradas, elas sejam realmente muito boas e não esteja nos planos expandir a utilização, por bastante tempo não será necessário ter gastos com infraestrutura então seria uma característica boa. Porém após alguns anos seria necessário migrar todo o sistema para novas máquinas pois as atuais poderiam se tornar obsoletas. Caso esteja no planejamento expandir a utilização, será uma característica ruim pois se mais máquinas forem necessárias será um alto gasto com infraestrutura que, como já mencionado, precisará ser substituída no futuro e pode ficar por um período sendo parcialmente utilizada.

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

É bastante comum realizar a instalação do Wordpress utilizando a “stack” LAMP (Linux, Apache, MySQL, PHP), por isso, segue um guia sobre como instalar Wordpress localmente utilizando a LAMP.

-Pode-se começar instalando o MySQL, repare que o comando contém um pacote adicional, servirá de suporte para o PHP5:

```
sudo apt-get install mysql-server php5-mysql
```

Siga os passos após a execução do comando no terminal Linux.

-Após a instalação do MySQL pode-se instalar o Apache, segue o comando para Ubuntu:

```
sudo apt-get install apache2 libapache2-mod-php5
```

-Agora é necessário instalar o Wordpress em si. Para isso é necessário baixar o pacote de: <https://wordpress.org/download/>. Em seguida descompactá-lo e mover os arquivos para a pasta /var/www/html. Execute os seguintes comandos após feitos os passos acima:

```
cd /var/www/html
sudo wget http://wordpress.org/latest.tar.gz
sudo tar zxvf latest.tar.gz -strip-components=1
```

-Execute os seguintes comando para permitir a visualização/edição dos arquivos pelo servidor web.

```
sudo chown -R www-data *
sudo chgrp -R www-data *
```

-Agora é necessário criar o banco de dados que o site utilizará:

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

```
mysql> CREATE DATABASE wordpress;
mysql> GRANT ALL PRIVILEGES ON wordpress.* TO "wordpress"@"localhost"
IDENTIFIED BY "foopassword";
mysql> FLUSH PRIVILEGES;
mysql> EXIT
```

-Renomeie o arquivo wp-config-sample.php para wp-config.php

-Agora a interface de instalação do Wordpress deveria estar acessível através do link:
http://server_domain_or_IP

Protótipo I

- Usuários/Repositórios:

IsabellaRO/CharmCloud

chends888/juju-charm-tutorial

Questões Complementares

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

Em nosso caso, nenhum dos dois são aplicações distribuídas, ambos estão sendo servidos a partir de uma máquina apenas. Eles controlam mais de uma máquina porém não rodam em mais de uma, fazendo com que eles sejam classificados como aplicações não distribuídas.

2. Qual a diferença entre REST e RPC?

REST e RPC são ambas formas de se comunicar com um servidor remoto tanto para enviar dados à ele, ou para recebê-los, sendo a principal diferença entre REST e RPC é que o REST irá apenas realizar um requerimento de dados de acordo com as informações passadas, enquanto o RPC irá realizar uma chamada de função no servidor. Resultando no REST sendo utilizado majoritariamente para manipulação de dados, e RPC quando o assunto é tomada de ações.

3. O que é SOAP?

SOAP (Simple Object Access Protocol) é um protocolo de transferência de dados inteiramente baseada em XML, todas as mensagens, sejam elas de chamada de função, envio e recebimento de informações são escritas em XML. O SOAP pode ser utilizado através de chamadas HTTP bem como SMTP entre outras.

Concluindo

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

Um Deployment Orchestrator administra o ciclo de vida dos deployments em uma aplicação em nuvem. Ele comanda os deployments, un-deployments, redeployments e confirmações do usuário. Com um desses é possível reduzir o tempo e o esforço ao realizar deploys de várias instâncias em uma mesma aplicação, e no caso de ser recebido um requerimento solicitando mais recursos ou uma nova aplicação, ferramentas automatizadas podem executar tarefas que antes só poderiam ser realizadas por vários administradores que operam em suas próprias partes individuais da pilha física. Alguns exemplos são o próprio Juju, o Puppet e o ADAPT Deployment Orchestrator.

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

O Juju solicita uma máquina para o MaaS para poder subir a aplicação desejada. No nosso caso subimos o Vanilla mas também poderia ser um WordPress, por exemplo.

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

Aplicação Distribuída: Uma aplicação distribuída é qualquer aplicação projetada para executar em várias máquinas simultaneamente ou um grupo de processos que executa em mais de um computador trabalhando de forma coordenada para concluir uma tarefa ou atingir algum objetivo.

Alta Disponibilidade: Alta disponibilidade é uma aplicação ficar ativa o maior tempo possível e sem danos mesmo que uma falha de hardware, software, energia, entre outras aconteça, evitando perdas de tempo e dinheiro.

Load Balancing: O Load Balancing serve para evitar que um hardware alcance o seu limite. Ele funciona distribuindo a carga de trabalho de forma uniforme entre computadores, discos rígidos e etc. Desta forma é possível além de evitar a sobrecarga, minimizar o tempo de resposta, otimizar a utilização dos recursos disponíveis e maximizar o desempenho dos hardwares.

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?

Em nosso caso o MAAS forneceu uma máquina de 32Gb pois é o que temos disponível e não sentimos falta de espaço. Não seria necessário utilizar uma máquina de 32Gb, poderia ter sido feita uma instância, com o intuito de dividir a máquina em partes menores e trabalhar em paralelo, podendo realizar tarefas menores mais eficientemente. Isso se aplica muito bem em datacenters reais, onde temos máquinas gigantes, que nenhuma aplicação seria capaz de consumir todo o poder de processamento, assim podemos realizar a proposta acima para melhorar a produtividade como um todo.