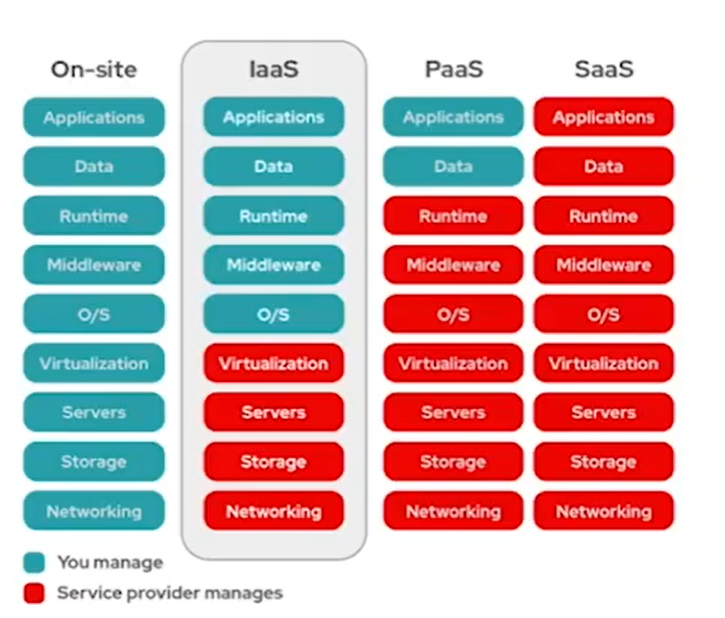
**VAGRANT: provisionando ambientes**



**Configurando ambiente**

**sudo apt install virtualbox: instalando o VirtualBox**

**sudo apt install virtualbox-dkms libelf-dev**

**sudo apt install vagrant: instala vagrant**

**vagrant --version ou -v: ver qual versão instalada**

**Criando Vagrantfile**

**vagrant init: cria vagrant file**

**vagrant init ubuntu/bionic64: cria vagrant file com a box do ubuntu/bionic64**

**vagrant status: ver status da máquina virtual**

**vagrant up: cria/liga máquina virtual**

**vagrant halt: desliga máquina virtual**

**vagrant destroy: desliga e remove a máquina do virtual box**

**vagrant ssh: entrar no terminal da máquina virtual**

**(no terminal da máquina virtual)**

**sudo apt update: atualiza ubuntu da vm**

**sudo apt install mysql-server: instala mysql serve na mv**

**telnet localhost 3306: verificar conexão com o banco de dados no endereço e porta passados**

**vagrant ssh-config: exibe configuracoes ssh**

**ssh-keygen -t rsa: cria chave keygen**

**Montando ambiente de desenvolvimento de um timo novo!**

* **Provisionar máquina que vai rodar a aplicação**
* **Provisionar máquina que vai rodar o banco de dados**

**Começamos pelo provisionamento da máquina que vai rodar o banco de dados**

* **Gerar arquivo vagrantfile usando uma box de ubuntu (vagrant init ubuntu/bionic64)**
* **Subir sua VM com vagrant up**
* **Verificando o status, deve estar running (vagrant status)**
* **Entrar no terminal da vm para instalar um database nela (vagrant ssh)**
* **Atualiza o ubuntu da vm (sudo apt update)**
* **Instala database mysql na vm (sudo apt install mysql-server)**
* **Verifica se o server tá rodando na vm (telnet localhost 3306)**
* **Sair do terminal da vm (exit)**

**Realizar configuração de rede na vm para poder acessar os serviços rodam dentro dela**

**Mapeamento de portas da vm com o nosso host no Vagrantfile**

* **config.vm.network "forwarded\_port", guest: 3306, host: 3306**

**Reiniciar máquina com vagrant halt e vagrant up apois realizar configuração**

* **Verifica se o server tá rodando na (telnet localhost 3306)**

**Configurando IP Estático para nossa máquina de banco de dados no Vagrantfile**

* **config.vm.network "private\_network", ip: "192.168.58.44"**

**Reiniciar máquina com vagrant reload após realizar configuração no vagrantfile**

**Configurar mysql para dar bind em endereços de fora do local host**

* **Entrar no terminal da vm (vagrant ssh)**
* **Abrir arquivo de configuração do mysql (sudo vim /etc/mysql/mysql.conf.d/mysqld.cnf)**
* **Mudar valor do bind-andress = 0.0.0.0**
* **Reiniciar servico de mysql na vm (sudo service mysql restart)**
* **Sair do terminal da vm (exit)**

**Testar conexão do ip estático e porta que definimos**

* **telnet 192.168.58.44 3306**

**Acesso das vms são feitos via chave privada**

**Verificando configurações da vm**

* **vagrant ssh-config**
* **Chave privada de acesso a vm que criamos**
  + **IdentityFile C:/Users/lilian.tavares/Documents/vagrant/ambiente\_dev/database/.vagrant/machines/default/virtualbox/private\_key**
* **Conectando com chave privada ssh –i {endereço da chave privada}**

**Criando chave especifica que será usada em todas as vms e compartilhada entre os devs**

* **Voltar um nível do diretorio atual (cd ..)**
* **Criar chave ssh-keygen –t rsa**
* **Enter file in which to save the key (C:\Users\lilian.tavares/.ssh/id\_rsa): C:\Users\lilian.tavares/.ssh/id\_rsa): C:\Users\lilian.tavares\Documents\vagrant\ambiente\_dev\ dev**
* **Colocar nome da chave no final do caminho onde ela será salva, nesse caso dev é o nome da chave**

**Configurar máquina virtual para utilizarem a chave que criamos (Vagrantfile do diretório database)**

**(provisionamento de file que copia arquivos entre host para máquina virtual)**

* **config.vm.provision "file", source: "C:/Users/lilian.tavares/Documents/vagrant/ambiente\_dev/dev", destination: "~/.ssh/dev.pub"**

**(provisionamento de SHELL que executo dentro da máquina virtual um** **comando de shell nesse caso usamos o cat)**

* **config.vm.provision "shell", inline: <<-SHELL**   
  **cat /home/vagrant/.ssh/dev.pub >> /home/vagrant/.ssh/authorized\_keys**

**SHELL**

**Depois de adicionar linhas de provisionamento é preciso provisionar a maquina**

* **vagrant provision**
* **ssh -i C:\Users\lilian.tavares\Documents\vagrant\ambiente\_dev\dev** [**vagrant@192.168.58.44**](mailto:vagrant@192.168.58.44) **(conectando com chave privada)**

**Fazendo provisionamento da máquina que vai rodar a aplicação (app)**

* **Gerar arquivo vagrantfile usando uma box de ubuntu (vagrant init ubuntu/bionic64)**
* **Configurar porta e IP estático no arquivo Vagrantfile**
* **Voltar um nível de diretório e criar a chave de acesso**
* **Configurar chave privada no Vagrantfile**
* **Provisionar maquina depois das configurações (vagrant provision)**