

## Devops

### Sumário

1.	Virtualização Conceito .....	2
2.	Virtualização prática – Criar VM / on-premisse.....	6
3.	Comandos básicos Linux.....	6
4.	Cloud Computing.....	9
5.	Conhecendo a Azure .....	11
6.	Criando uma VM Linux Azure.....	12

## 1. Virtualização Conceito

- **História da virtualização**

A virtualização começou nos anos 1960 com a IBM, que dividia grandes computadores (mainframes) para vários usuários ao mesmo tempo. Nos anos 1990 e 2000, a VMware revolucionou os servidores, permitindo rodar vários sistemas operacionais em uma única máquina. Com o tempo, a virtualização se expandiu para redes, armazenamento e nuvem, tornando-se essencial para tornar a TI mais ágil e eficiente.

- **Virtualização, o que é?**

Virtualização é a tecnologia que permite criar versões virtuais de sistemas operacionais, servidores, redes e até armazenamento. Em vez de depender de hardware físico dedicado, um único computador pode rodar vários sistemas ao mesmo tempo, aumentando a eficiência e reduzindo custos.

- **Máquina Virtual: o que é?**

Uma máquina virtual (VM) é um ambiente simulado dentro de um computador físico. Ela funciona como um computador independente, com seu próprio sistema operacional e aplicativos, mas compartilha os recursos do hardware real. Isso permite testar sistemas, isolar aplicações e otimizar o uso dos servidores.

- **Hypervisors : o que é e por que usar?**

O hypervisor é um software que permite criar e gerenciar máquinas virtuais (VMs). Ele age como um "chefe do tráfego", dividindo e distribuindo os recursos do hardware físico (processador, memória, disco) entre várias VMs. Arquitetura

### Por que usar um Hypervisor?

- **Melhor aproveitamento do hardware** – Permite rodar vários sistemas no mesmo servidor.
- **Mais segurança** – Isola as VMs, evitando que um problema em uma afete as outras.
- **Flexibilidade** – Facilita a migração de sistemas sem precisar de um novo computador.
- **Redução de custos** – Menos servidores físicos = menos gastos com energia e manutenção.

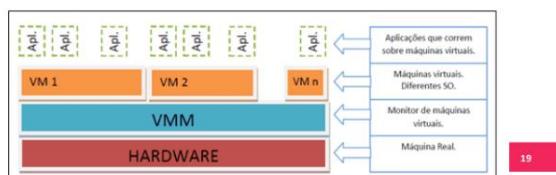
- **Propriedades básicas**

- **Particionamento** – Permite dividir um único hardware físico em várias máquinas virtuais, cada uma funcionando de forma independente.
- **Isolamento** – Cada VM opera separadamente, garantindo que falhas ou ameaças em uma não afetem as outras.
- **Encapsulamento** – A VM é um "pacote" completo, incluindo sistema operacional, aplicativos e configurações, podendo ser movida ou copiada facilmente.

- Objetivo da virtualização
  - Centralizar tarefas administrativas
  - Diminuir custos (energia, pessoal, manutenção etc)
  - Escalar aplicações
  - Agilidade na resposta das necessidades dos negócios
  - Os recursos virtualizados são representados por software
  - Recursos virtuais são criados a partir de um conjunto de recursos físicos
- Principais utilizações da VM
  - Hardware
  - Servidores
  - Armazenamento (Storage)
- Virtualização de hardware

Permite rodar vários sistemas operacionais em um único computador físico, criando máquinas virtuais que compartilham os recursos do processador, memória e armazenamento. Exemplo: Hypervisores como VMware e VirtualBox.

- Possui três componentes principais
  - **Hospedeiro**: Um servidor (computador) físico que será virtualizado
  - **Camada de Virtualização**: O Software que permite realizar e gerenciar a virtualização
  - **Convidado**: O componente virtualizado



- Virtualização de servidores

Divide um único servidor físico em vários servidores virtuais, otimizando o uso dos recursos, reduzindo custos e facilitando a escalabilidade.

Exemplo: Servidores cloud como AWS e Microsoft Azure.

### I VIRTUALIZAÇÃO DE SERVIDORES FIAP BENEFÍCIOS

- Diminuição espaço físico (número de máquinas)
- Economia de energia elétrica
- Isolamento e segurança da aplicação ou serviço
- Simplificação para disponibilizar novos servidores
- Múltiplos sistemas operacionais



- Virtualização de storage

A virtualização de storage combina vários dispositivos de armazenamento físico (HDs, SSDs, NAS, SAN) em um único sistema virtual. Isso facilita a gestão e otimiza o desempenho.

### I VIRTUALIZAÇÃO DE STORAGE FIAP BENEFÍCIOS

- Melhorar a eficiência, reduzindo custos administrativos e liberando recursos de TI
- Fornecer um sistema de gerenciamento de armazenamento de dados centralizado
- Alcançar uma maior escalabilidade e flexibilidade para o crescimento futuro



23

### I VIRTUALIZAÇÃO FIAP

- Utilização
  - Criar ambientes seguros (Máquinas separadas)
  - Construir sistemas escaláveis
  - Otimizar armazenamento
  - Automatizar processos (Clonagem, Disponibilização de máquinas etc)
  - Emular ambientes de Homologação
  - Cloud Computing



- Desvantagens x Vantagens (virtualização)

### | BENEFÍCIOS DA VIRTUALIZAÇÃO FIAP

- Todas as máquinas VM tem os recursos ideais para obter o desempenho máximo
- Não é necessário tantos servidores para atender as demandas de negócio
- O cálculo e a sobrecarga de máquina diminuem muito e a melhora dos aplicativos acaba sendo acentuada
- Resultado: maior valor com menor custo

### | DESVANTAGENS DA VIRTUALIZAÇÃO FIAP

- Grande uso de espaço em disco, já que é preciso de todos os arquivos para cada sistema operacional instalado em cada máquina virtual
- Dificuldade no acesso direto a hardware, como por exemplo placas específicas, sendo necessário uma especialização para administração de servidores
- Desempenho: A introdução de uma camada extra de software entre o sistema operacional e o hardware, o Hypervisor, gera um custo de processamento superior ao que se teria sem a virtualização

- Virtualizar máquina pessoal x profissional

**PORQUE VIRTUALIZAR MÁQUINA PESSOAL? FIAP**

- Para estudar várias versões de sistemas operacionais
- Testar vários softwares
- Instalar vários sistemas, evitando ter questões de mal funcionamento
- Testar ambiente de redes, realizando comunicações entre servidores (máquinas virtuais)
- Segurança: ambiente específico para acesso ao ambiente virtual (internet)



**PORQUE VIRTUALIZAR MÁQUINA PROFISSIONAL ? (SERVIDORES) FIAP**

- Gerenciamento centralizado
- Instalações simplificadas
- Facilidade para execução de backups
- Independência de hardware
- Disponibilizar novos servidores em pouco tempo
- Economia de espaço físico
- Economia de energia elétrica
- Segurança: melhor ambiente para cada tipo de aplicação (requerimentos de segurança)
- Confiança e disponibilidade (a falha de uma não prejudica as demais)

## 2. Virtualização prática – Criar VM / on-premisse

- Softwares necessários
  - Hypervisors (utilizado para gerenciamento das máquinas virtuais): Oracle Vm VirtualBox
- Passo a passo da criação da VM no Oracle VirtualBox (Linux)
  - com a imagem do sistema operacional crie a VM
- Update kernel e pacotes auxiliares
- Extra: habilitar ícones no Desktop

## 3. Comandos básicos Linux

- Criar novo usuário no Linux

**Criando um novo usuário no Linux**

- ◆ Pelas Configurações (Interface Gráfica)
  1. Abra as Configurações do Sistema.
  2. Vá até a seção Usuários (ou "Contas de Usuário").
  3. Clique em Adicionar Usuário.
  4. Escolha o tipo de conta (Administrador ou Padrão).
  5. Defina um nome de usuário e senha.
  6. Clique em Criar para finalizar.

• Pelo Terminal

1. Criar o usuário:

```
bash
```

Copiar Editar

```
sudo adduser nome_do_usuario
```

- Ele pedirá uma senha e informações opcionais (nome, telefone, etc.).

2. Dar permissões de administrador (opcional):

```
bash
```

Copiar Editar

```
sudo usermod -aG sudo nome_do_usuario
```

3. Verificar se o usuário foi criado:

```
bash
```

Copiar Editar

```
cat /etc/passwd | grep nome_do_usuario
```

- Estrutura de diretórios

- Criar um diretório: **mkdir nome\_dir**

- O comando **pwd** exibe o caminho completo do diretório atual no terminal do Linux.

- Ir até a home do user. Ex: **cd /home/admdimdim**

- Comando: **mkdir -p Dimdim/Financeiro/ContasAPagar** cria uma estrutura de diretórios de forma recursiva. Então Dimdim e dentro dele as pastas Financeiro e Contas a pagar

- **ls** lista os arquivos/pastas do diretório atual

- Curingas

- \* Atua como um substituto para qualquer sequência de caracteres

- ? Atua como substituto para qualquer caractere único

- **touch** esse comando cria arquivos

- listar o conteúdo de arquivos com: **cat** ou **more** (permite a paginação do conteúdo se ele for mto extenso)

O ALIAS funciona como um apelido para um comando no Linux

**alias seuAtalho='Comando --OPCOES'**

Exemplos:

```
alias c='clear'
```

```
alias ls='ls -la'
```

**Atenção – Sem espaços**

- copiando arquivos e diretórios com **cp**: **cp (arquivo ou diretório de origem) (arquivo ou diretório destino)**

Exemplo: vamos criar um arquivo dentro de `/home/admdimdim` e copiar esse arquivo para dentro de um subdiretório:

1. Vá ao diretório: `/home/admdimdim`
2. Crie o arquivo `texto1` a partir do arquivo `passwd` em `/etc`  
`cat /etc/passwd > texto1`
3. Copie o arquivo criado para o diretório criado anteriormente `ex1`  
`cp texto1 ex1`
4. Isso feito, insira o seguinte comando  
`ls -l ex1`

23

- Copiando o conteúdo de um diretório para outro

Vamos copiar todo o conteúdo do diretório `/home/admdimdim` para dentro do subdiretório `ex3`:

1. Vá ao diretório `/home/admdimdim`
2. Entre no subdiretório `ex3` criado e digite: `cp /home/admdimdim/* .`
3. Agora, escreva: `ls -l`
4. Veja o conteúdo do diretório original: `ls -l /home/admdimdim/`



Observe que os subdiretórios de `/home/admdimdim` não foram copiados, somente os arquivos

Para copiar os subdiretórios, use a opção "r" do comando

Vamos, agora, fazer o mesmo, só que, desta vez, copiando os subdiretórios e arquivos de `/home/admdimdim`

1. Primeiro, digite: `cp -r /home/admdimdim /* .`
2. Depois, digite o seguinte comando: `ls -l`

24

### - movendo arquivos ou diretórios com `mv`: **mv (arquivo ou diretório origem) (arquivo ou diretório destino)**

Exemplo: vamos criar um arquivo dentro de `/home/admdimdim` e copiar esse arquivo para dentro de um subdiretório:

1. Vá ao diretório: `/home/admdimdim`
2. Crie o arquivo `exemplo1.txt`  
`touch exemplo1.txt`
3. Crie o subdiretório `dir1` e move o arquivo criado para esse subdiretório  
Crie o diretório com o comando `mkdir dir1`  
`mv exemplo1.txt dir1`
4. Isso feito, insira o seguinte comando  
`ls -l dir1`

Se a origem e o destino forem os mesmos o comando renomeia o arquivo/diretório

25

### - Apagando arquivos e diretórios utilizando `rm`

Exemplos:

Apaga o arquivo especificado: `rm texto1`

Apaga o arquivo especificado forçadamente (-f de force): `rm -f texto3`

Apaga o diretório especificado: `rm -r ex1`

Apaga o diretório especificado forçadamente: `rm -rf ex2`

1. Vá ao diretório: `/home/admdimdim/ex`
2. Apague os arquivos: `rm texto*` (confirmando)
3. Apague os arquivos: `rm -f doc*` (sem solicitar confirmação)
4. Apague o diretório: `rm -r ex1` (confirmando)
5. Apague o diretório: `rm -rf ex2` (sem solicitar confirmação)

26

### - desde pág 29 – monitoramento de recursos

### - desde pág 33 – comandos utilitários

Grep (procura por textos e retorna qual arquivo ou diretório foi encontrado, inclusive mostrando a linha em que isso ocorreu)

- DTAB COMPLETA COMANDO
- Comandos básicos

## 4. Cloud Computing

- História

A computação em nuvem começou nos anos 1960, mas se popularizou no final dos anos 2000, com o advento de empresas como Amazon, Google e Microsoft oferecendo serviços de armazenamento e processamento online.

- Definição: Cloud Computing

Cloud Computing é a entrega de serviços de computação (como armazenamento, processamento e software) pela internet, permitindo acessar dados e aplicativos sem precisar de hardware próprio.

- Localização da Cloud Computing – DataCenter

Os serviços de nuvem são hospedados em grandes **DataCenters**, que são instalações físicas com servidores, armazenamento e redes para fornecer esses serviços a partir de qualquer lugar do mundo.

- Princípios da Cloud Computing

- **Acesso remoto:** Conexão pela internet.
- **Escalabilidade:** Recursos podem ser aumentados ou reduzidos conforme necessário.
- **Pay-as-you-go:** Paga apenas pelo que usa.
- **Elasticidade:** Capacidade de expandir ou reduzir recursos dinamicamente.

- Tipos de nuvens

- **Nuvem Pública:** Recursos compartilhados entre vários usuários (ex: AWS, Google Cloud).
- **Nuvem Privada:** Recursos dedicados a uma única organização.
- **Nuvem Híbrida:** Combinação das duas, com alguns recursos privados e outros públicos.

- Modelo em camadas

A arquitetura da nuvem geralmente é dividida em três camadas:

- **IaaS (Infraestrutura como Serviço)**: Fornece recursos básicos como servidores e armazenamento.
- **PaaS (Plataforma como Serviço)**: Oferece uma plataforma para desenvolver e executar aplicativos.
- **SaaS (Software como Serviço)**: Fornece aplicativos prontos para uso (como Google Docs, Microsoft 365).

- Por que usar SaaS?
  - **Facilidade de uso**: Não precisa instalar ou gerenciar software.
  - **Acessibilidade**: Acessível de qualquer dispositivo com internet.
  - **Custos reduzidos**: Sem necessidade de infraestrutura própria.
- Por que usar IaaS?
  - **Escalabilidade**: Aumente ou diminua os recursos conforme a demanda.
  - **Flexibilidade**: Controle total sobre os recursos e a infraestrutura.
  - **Economia**: Paga apenas pelo que usar.
- Por que usar Paas?
  - **Desenvolvimento rápido**: Ambientes prontos para desenvolvimento de aplicativos.
  - **Gerenciamento simplificado**: Não precisa se preocupar com a infraestrutura.
  - **Escalabilidade**: Fácil de adaptar para cargas de trabalho variáveis
- Saas, Paas e IaaS – cenário de comparação
  - **SaaS**: Aplicativos prontos, como Gmail.
  - **PaaS**: Plataformas para desenvolvedores, como Heroku.
  - **IaaS**: Infraestrutura básica para rodar seus próprios sistemas, como AWS EC2
- Principais razões para o uso de Cloud Computing
  - **Escalabilidade**: Expansão rápida conforme a necessidade.
  - **Redução de custos**: Sem grandes investimentos em hardware.
  - **Mobilidade**: Acesso remoto aos dados e aplicativos.
- CAPEX e OPEX
 

**CAPEX** (Capital Expenditure): Investimento em ativos fixos, como servidores e hardware.

**OPEX** (Operational Expenditure): Gastos operacionais recorrentes, como custos mensais de serviços de nuvem.
- CAPEX e OPEX – como calcular as diferenças
  - **CAPEX** envolve grandes investimentos iniciais (ex: compra de servidores).

**OPEX** envolve custos recorrentes e mensais (ex: assinaturas de serviços de nuvem).

- OPEX em Cloud Computing

Com a nuvem, os custos se tornam mais previsíveis e recorrentes, eliminando grandes investimentos iniciais em infraestrutura.

- Por que usar Cloud Computing

- **Redução de custos:** Sem necessidade de grandes investimentos em hardware.
- **Escalabilidade:** Expansão rápida de recursos conforme necessário.
- **Acessibilidade:** Acesso remoto e flexível.

- Quando usar Cloud Computing

- Quando precisar de escalabilidade e flexibilidade.
- Quando quiser reduzir custos com infraestrutura.
- Quando precisar de acesso remoto a dados e aplicativos.

- Servidores hoje em dia

Os servidores modernos são virtualizados e podem ser acessados via nuvem, permitindo maior flexibilidade e escalabilidade.

- Elasticidade com Cloud Computing

A elasticidade permite ajustar a quantidade de recursos disponíveis (como CPU e memória) em tempo real, conforme a demanda do sistema. É ideal para lidar com picos de tráfego ou necessidades temporárias.

## 5. Conhecendo a Azure

- Grupos de Recursos

Um **grupo de recursos** no Azure é um container lógico que agrupa vários recursos relacionados, como máquinas virtuais, bancos de dados, redes, etc. Esses recursos podem ser gerenciados e monitorados juntos de forma organizada.

**Principais características:**

- Todos os recursos dentro de um grupo de recursos compartilham a mesma região de datacenter.
- Pode conter recursos de diferentes tipos (máquinas virtuais, redes, etc.).
- Facilita o gerenciamento de segurança, monitoramento e implantação.

- Recursos

**Recursos** são componentes individuais do Azure, como:

- **Máquinas Virtuais (VMs).**
- **Redes Virtuais.**
- **Armazenamento** (Blobs, discos, etc.).
- **Bancos de Dados** (SQL, Cosmos DB, etc.).
- **Funções** (Azure Functions, por exemplo).

Esses recursos são os blocos que formam a infraestrutura e os serviços que você usa no Azure.

- Azure Resource Manager (ARM)

O **Azure Resource Manager (ARM)** é o serviço de gerenciamento de recursos do Azure. Ele permite criar, atualizar e excluir recursos dentro de um grupo de recursos, além de fornecer controle de acesso, políticas e monitoramento.

**Funções principais do ARM:**

- **Gerenciamento centralizado:** Controla e organiza todos os recursos do Azure.
- **Implantação de recursos:** Permite implantar recursos de forma eficiente e automatizada (com templates, por exemplo).
- **Controle de acesso:** Usa o Azure Role-Based Access Control (RBAC) para controlar quem pode fazer o quê com os recursos.
- **Monitoramento:** Fornece insights sobre o desempenho e uso dos recursos.

## 6. Criando uma VM Linux Azure