

Configuração Básica de Ambiente de Computação em Nuvem

Objetivo: Os estudantes devem configurar uma máquina virtual em uma plataforma de nuvem (AWS, Google Cloud ou Azure), instalar o sistema operacional, manipular arquivos no terminal Linux (10 a 15 comandos diferentes) e documentar o processo com capturas de tela detalhadas. O relatório deve incluir a descrição do ambiente criado, dificuldades encontradas e soluções aplicadas.

Nomes:

- Ettore Grecco
- Luciano Reis Massaro

1. Implementação de um Servidor Virtual na Nuvem AWS

Este projeto visa solidificar os conhecimentos fundamentais em computação em nuvem, focando na implementação e configuração de um servidor virtual desde sua concepção. A iniciativa representa um mergulho prático no universo das plataformas de nuvem, desenvolvendo habilidades cruciais como o gerenciamento de sistemas operacionais Linux em um ambiente remoto e a utilização eficiente do terminal para manipulação de arquivos e pacotes.

Para atingir esses objetivos, será estabelecido um ambiente na Amazon Web Services (AWS) com a criação de uma Máquina Virtual (VM) por meio do serviço EC2. O sistema operacional escolhido é o Ubuntu Linux, devido à sua popularidade em ambientes de servidor e à familiaridade inicial do grupo com a plataforma. Este servidor virtual funcionará como um ambiente de testes seguro, onde poderemos exercitar comandos, administrar serviços e aprofundar nosso entendimento sobre como as infraestruturas de nuvem operam, preparando-nos para desafios mais avançados na área.

2. Descrição do Ambiente

Para a realização desta atividade, foi configurado o seguinte ambiente:

- **Provedor de Nuvem:** Amazon Web Services (AWS)
- **Serviço:** Elastic Compute Cloud (EC2)
- **Sistema Operacional:** Ubuntu 24.04.3 LTS
- **Método de Acesso:** Conexão remota via terminal SSH, utilizando um par de chaves de segurança.

Toda a interação com o sistema, incluindo a configuração e manipulação de arquivos, foi realizada exclusivamente por meio da linha de comando, simulando um ambiente de servidor real.

3. Etapas Executadas e Comandos Utilizados

O projeto foi dividido nas seguintes etapas:

3.1. Criação e Acesso à Instância

1. **Criação da Instância:** Uma nova instância EC2 foi provisionada no painel de controle da AWS, selecionando a imagem (AMI) do Ubuntu 24.04.3 LTS.
2. **Conexão Segura:** O acesso à instância foi estabelecido via SSH, utilizando a chave privada (.pem) fornecida pela AWS durante a criação da máquina.

3.2. Configuração Inicial do Sistema

Após o primeiro acesso, foram executados os seguintes comandos para preparar o ambiente:

1. **Elevação de Privilégios:** Para obter permissões de administrador (root), foi utilizado o comando:

```
Bash
```

```
sudo su
```

2. **Atualização do Sistema:** Para garantir que todos os pacotes estivessem em suas versões mais recentes, o sistema foi atualizado:

```
Bash
```

```
apt update && apt upgrade -y
```

3.3. Manipulação de Arquivos e Diretórios

Foram executados 12 comandos distintos para demonstrar a manipulação básica do sistema de arquivos e processos, conforme detalhado na tabela abaixo:

Nº Comando	Descrição da Ação
1 pwd	Exibiu o caminho completo do diretório de trabalho atual.
2 ls	Listou os arquivos e diretórios presentes no local atual.
3 ls -la	Realizou uma listagem detalhada, incluindo arquivos ocultos e permissões.
4 mkdir novo_diretorio	Criou um novo diretório chamado novo_diretorio.
5 cd novo_diretorio	Navegou para dentro do diretório recém-criado.
6 touch meu_arquivo.txt	Criou um arquivo vazio chamado meu_arquivo.txt.
7 cp meu_arquivo.txt copia.txt	Copiou o arquivo, criando um novo chamado copia.txt.
8 echo "Teste de conteúdo" > meu_arquivo.txt	Inseriu um texto dentro do arquivo, sobrescrevendo seu conteúdo.
9 cat meu_arquivo.txt	Exibiu o conteúdo do arquivo diretamente no terminal.
10 ps aux	Listou todos os processos em execução no sistema.

Nº Comando	Descrição da Ação
11 apt install htop -y	Instalou a ferramenta de monitoramento de processos htop.
12 htop	Executou o monitor de processos interativo em tempo real.
Exportar para as Planilhas	

3.4. Instalação de Serviço Adicional (Servidor Web)

Para simular a instalação de uma aplicação, o servidor web Apache2 foi adicionado ao ambiente com o seguinte comando:

Bash

```
apt install apache2 -y
```

Após a instalação, o serviço foi iniciado automaticamente, tornando a instância capaz de responder a requisições HTTP..

4. Conclusão

A atividade foi concluída com sucesso, atingindo todos os objetivos propostos. Foi possível configurar uma instância na nuvem, administrar o sistema operacional Linux via linha de comando, manipular arquivos e diretórios, e instalar um serviço adicional. A prática reforçou a importância dos comandos básicos do Linux e proporcionou uma valiosa experiência no gerenciamento de uma infraestrutura em nuvem, fundamental para a atuação profissional na área de tecnologia.

5. Capturas de tela

```

0.0% Tasks: 33, 40 thr, 76 kthr; 1 running
0.7% Load average: 0.00 0.00 0.00
Mem[|||||] 266M/914M Uptime: 00:54:10
Swp[|||||] 0K/0K

Main I/O
PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command
7282 root 20 0 8752 4388 3492 R 0.7 0.5 0:00.06 htop
1 root 20 0 22632 13552 9456 S 0.0 1.4 0:02.74 /sbin/init
129 root 19 -1 50432 13960 12936 S 0.0 1.5 0:00.30 /usr/lib/systemd/systemd-journald
190 root RT 0 282M 26912 8608 S 0.0 2.9 0:00.11 /sbin/multipathd -d -s
197 root 20 0 26492 8132 5060 S 0.0 0.9 0:00.30 /usr/lib/systemd/systemd-udevd
202 root 20 0 282M 26912 8608 S 0.0 2.9 0:00.00 /sbin/multipathd -d -s
203 root RT 0 282M 26912 8608 S 0.0 2.9 0:00.00 /sbin/multipathd -d -s
204 root RT 0 282M 26912 8608 S 0.0 2.9 0:00.00 /sbin/multipathd -d -s
205 root RT 0 282M 26912 8608 S 0.0 2.9 0:00.00 /sbin/multipathd -d -s
206 root RT 0 282M 26912 8608 S 0.0 2.9 0:00.19 /sbin/multipathd -d -s
207 root RT 0 282M 26912 8608 S 0.0 2.9 0:00.00 /sbin/multipathd -d -s
325 systemd-re 20 0 21592 12864 10560 S 0.0 1.4 0:00.15 /usr/lib/systemd/systemd-resolved
537 systemd-re 20 0 22408 9728 8576 S 0.0 1.1 0:00.04 /usr/lib/systemd/systemd-networkd
599 root 20 0 2720 1932 1804 S 0.0 0.2 0:00.00 /usr/sbin/ecpid
603 root 20 0 7224 2744 2468 S 0.0 0.3 0:00.01 /usr/sbin/crond -f -p

```

```
NAME
    cp - copy files and directories

SYNOPSIS
    cp [OPTION]... [-t] SOURCE DEST
    cp [OPTION]... SOURCE... DIRECTORY
    cp [OPTION]... -t DIRECTORY SOURCE...

DESCRIPTION
    Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.

    Mandatory arguments to long options are mandatory for short options too.

    -a, --archive
        same as --dt --preserve=all

    --attributes-only
        don't copy the file data, just the attributes

    --backup[=CONTROL]
        make a backup of each existing destination file

    -b
        like --backup but does not accept an argument

    --copy-contents
        copy contents of special files when recursive

    -d
        same as --no-dereference --preserve=links

    --debug
        explain how a file is copied. Implies -v

    -f, --force
        if an existing destination file cannot be opened, remove it and try again (this option is ignored when the -n option is also used)

    -i, --interactive
        prompt before overwrite (overrides a previous -n option)

    -H
        follow command-line symbolic links in SOURCE
```