Projeto - Servidor Básico Linux

Servidor Linux Multiusuário

- Autor: André Luis
- Ambiente: Ubuntu Server 24.04
- Virtualização: VirtualBox
- Objetivo: Simular ambiente corporativo multiusuário com controle de acesso

---

1. Objetivo

Criar um servidor multiusuário, com separação de privilégios e permissões, grupos ,acesso remoto e segurança básica, simulando ambientes corporativos reais.

---

2. Escopo

2.1 Incluso

- **Servidor Linux multiusuário**: Ubuntu Server 24.04 em ambiente virtualizado (VirtualBox)
- **Gestão de usuários e grupos**: Criação de diferentes perfis (administradores e usuários comuns) com grupos específicos
- **Controle de acesso e privilégios**:
    - Separação de privilégios entre usuários
    - Configuração de sudo para grupo de administradores
- **Segurança básica**:
    - Configuração de firewall (UFW) com políticas restritivas
    - Restrições de acesso SSH (apenas grupo de administradores)
    - Bloqueio de acesso root via SSH
- **Serviços essenciais**:
    - Servidor SSH para acesso remoto (porta 22)
    - Servidor web Nginx com página HTML básica (porta 80)

2.2 Não incluso

- Alta disponibilidade
- Monitoramento avançado
- Ambiente em produção

---

3. Arquitetura Implementada

> *Como forma de controle de segurança foi criado uma `snapshot` com o objetivo de manter o estado neutro da máquina antes das alterações.*

> *Para a execução das etapas a máquina virtual foi reconfigurada para rede modo **Bridge(Placa em modo Brigde)** para melhor desenvolvimento e execução de serviços como ssh e firewall.*

3.1 Usuários e Grupos

Foram criados diferentes tipos de usuários para a simulação de diferentes perfis de acesso:

- `andre` –> administrador principal
- `aux_adm` –> administrador auxiliar
- `dev_1` / `dev_2` –> usuários comuns

Para o facilitamento o gerenciamento de permissões e de recursos foram criados tipos de grupo de acordo com perfil de usuário.

- `admins`: usuários responsáveis por tarefas administrativas.
- `devs`: usuários com acesso limitado

3.2 Controle de Acesso

- **SSH**: Restrito ao grupo `admins` via `configuração de ssh`
- **Firewall**: Política "**deny all incoming**" + permissão apenas para SSH e HTTP
- **Privilégios**: Grupo `admins` tem acesso **sudo**.

---

4. Configuração Técnica

4.1 Políticas de Firewall

```
# Políticas padrões
sudo ufw default deny incoming   # Bloqueia tudo que entra
sudo ufw default allow outgoing  # Permite tudo que sai

#Serviços
sudo ufw allow ssh              # Porta 22
sudo ufw allow 80/tcp           # HTTP (Para Nginx)
```

Com isso apenas as portas de SSH (22) e HTTP (80) estão acessíveis externamente com o restante das portas bloqueadas, fornecendo assim segurança e os serviços que serão utilizados no ambiente.

4.2 Segurança de Acesso Remoto

> **SSH configurado:**

Regra	Escolha	
`PermitRootLogin`	`no`	Bloqueio do Root
`PasswordAuthentication`	`yes`	Autenticação por senha
`AllowGroups`	`admins`	Acesso somente a administradores

4.3 Gestão de privilégios

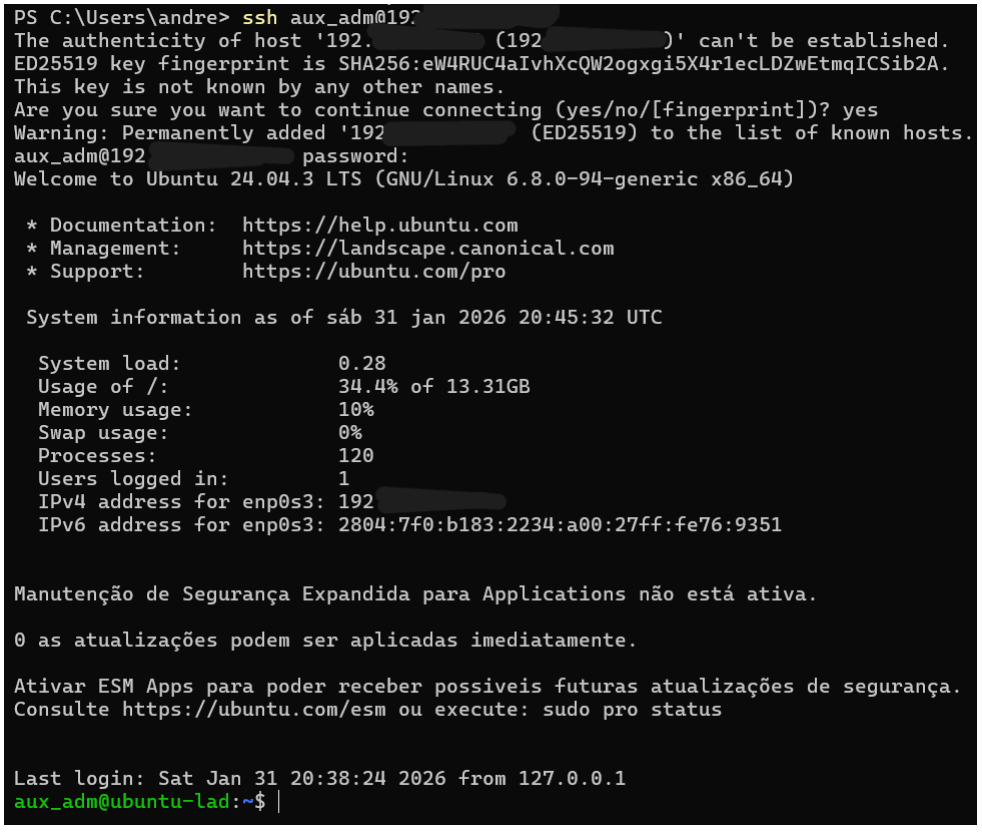
```
sudo visudo

#Arquivo sudoers
%admins ALL=(ALL) ALL
```

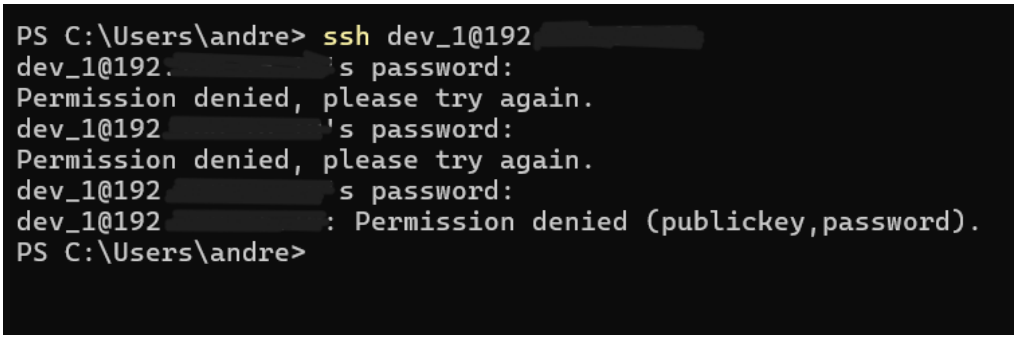
> **Acesso completo para administradores**

5. Testes

5.1 Validação SSH



> **O usuário `aux_adm` obteve sucesso no acesso via SSH, com o login realizado corretamente. Esse resultado confirma que o servidor está operando conforme as regras estabelecidas, apresentando um nível básico de segurança efetivo.**



> **Conforme configurado, o grupo `devs` não possui permissão de acesso via SSH, garantindo que apenas usuários devidamente autorizados possam utilizar esse serviço para entrada no sistema.**

5.2 Nginx

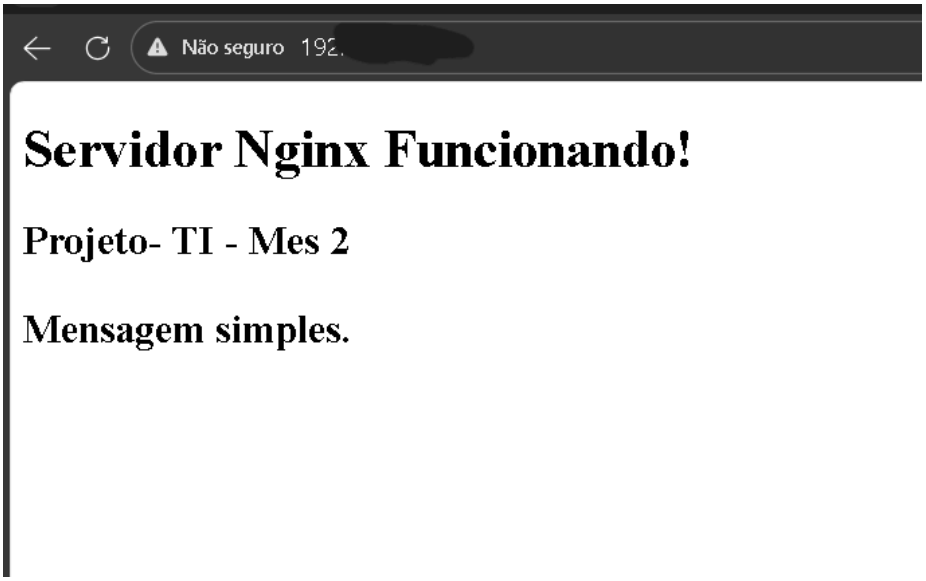
Para a simulação de um servidor corporativo visível, foi instalado e configurado o software **Nginx**, utilizado como interface de apresentação do servidor e validação do serviço web no ambiente. Para isso foi utilizado a linguagem **HTML** em estado básico.

```
<!-- sudo nano /var/www/html/index.html-->

<html>
<head>
    <title>Servidor Linux Basico - Projeto TI</title>
</head>
<body>
     <h1>Servidor Nginx Funcionando!</h1>
     <h2>Projeto - TI - Mes 2</h2>
     <p>Mensagem simples.</p>
</body>
</html>
```

Por fim, foram configuradas as permissões essenciais para garantir a segurança e o correto funcionamento do servidor web **Nginx**.

```
sudo chown -R www-data:www-data /var/www/html/ # Possibilita o dono modificar arquivos
sudo chmod -R 755 /var/www/html/ # Impede que outros usuarios ou grupos modifiquem o site
```



6.Diagrama do Ambiente