Roteiro de Teste - Born2beroot

Testes Preliminares

1. Verificação do Repositório Git

- [ ] Verificar se o arquivo `signature.txt` está presente na raiz do repositório clonado
- [ ] Comparar a assinatura no `signature.txt` com a assinatura do arquivo `.vdi` da máquina virtual usando `diff`
- [ ] Garantir que as assinaturas são idênticas

Comandos para teste:

```bash
cat signature.txt
# Comparar com a assinatura do .vdi (localizar o arquivo .vdi primeiro)
```

Parte Obrigatória

2. Visão Geral do Projeto

Você deve ser capaz de explicar:

- [ ] Como uma máquina virtual funciona
- [ ] Sua escolha de sistema operacional (Debian ou CentOS)
- [ ] Diferenças básicas entre CentOS e Debian
- [ ] O propósito das máquinas virtuais
- [ ] **Se escolheu CentOS:** o que são SELinux e DNF
- [ ] **Se escolheu Debian:** diferença entre aptitude e apt, e o que é AppArmor
- [ ] Verificar se o script de monitoramento exibe informações a cada 10 minutos

3. Configuração Simples

Testes a realizar:

- [ ] Verificar que a máquina não possui ambiente gráfico no boot
- [ ] Fazer login com um usuário que NÃO seja root
- [ ] Verificar se a senha segue as regras impostas no subject
- [ ] Verificar se o serviço UFW está iniciado
- [ ] Verificar se o serviço SSH está iniciado
- [ ] Confirmar que o SO é Debian ou CentOS

Comandos para teste:

```bash
# Verificar UFW
sudo ufw status

# Verificar SSH
sudo systemctl status ssh
# ou
sudo service ssh status

# Verificar sistema operacional
cat /etc/os-release
```

4. Usuário

Testes a realizar:

- [ ] Verificar se existe um usuário com o login do estudante avaliado
- [ ] Confirmar que este usuário pertence aos grupos "sudo" e "user42"
- [ ] Criar um novo usuário e atribuir uma senha seguindo as regras do subject
- [ ] Criar um grupo chamado "evaluating" e adicionar o novo usuário a ele
- [ ] Verificar se o usuário pertence ao grupo "evaluating"

Comandos para teste:

```bash
```

```
# Verificar grupos do usuário
groups nome_usuario

# Criar novo usuário
sudo adduser novo_usuario

# Criar grupo
sudo groupadd evaluating

# Adicionar usuário ao grupo
sudo usermod -aG evaluating novo_usuario

# Verificar se foi adicionado
groups novo_usuario
```

Explicações necessárias:

- [ ] Como foram implementadas as regras de política de senha
- [ ] Vantagens da política de senha
- [ ] Vantagens e desvantagens da implementação

5. Hostname e Partições

Testes a realizar:

- [ ] Verificar se o hostname está formatado como: `login42`
- [ ] Modificar o hostname substituindo o login pelo seu, reiniciar e verificar se foi atualizado
- [ ] Restaurar o hostname original
- [ ] Visualizar as partições da máquina virtual
- [ ] Comparar a saída com o exemplo dado no subject

Comandos para teste:

```bash
```

```
# Verificar hostname atual
hostname

# Modificar hostname
sudo hostnamectl set-hostname novo_hostname

# Verificar partições
lsblk
# ou
fdisk -l

# Para LVM
sudo pvdisplay
sudo vgdisplay
sudo lvdisplay
```

Explicações necessárias:

- [ ] Como o LVM funciona e para que serve

6. SUDO

Testes a realizar:

- [ ] Verificar se o programa "sudo" está instalado
- [ ] Adicionar o novo usuário ao grupo "sudo"
- [ ] Verificar se a pasta `/var/log/sudo/` existe e contém pelo menos um arquivo
- [ ] Verificar o conteúdo dos arquivos (deve conter histórico de comandos sudo)
- [ ] Executar um comando via sudo e verificar se os logs foram atualizados

Comandos para teste:

```bash
```

```
# Verificar se sudo está instalado
which sudo

# Adicionar usuário ao grupo sudo
sudo usermod -aG sudo nome_usuario

# Verificar pasta de logs
ls -la /var/log/sudo/

# Ver conteúdo dos logs
sudo cat /var/log/sudo/*

# Testar comando sudo e verificar logs novamente
sudo ls
sudo cat /var/log/sudo/*
```

Explicações necessárias:

- [ ] Valor e funcionamento do sudo com exemplos
- [ ] Implementação das regras impostas pelo subject

7. UFW (Firewall)

Testes a realizar:

- [ ] Verificar se o programa "UFW" está instalado
- [ ] Verificar se está funcionando corretamente
- [ ] Listar regras ativas (deve existir regra para porta 4242)
- [ ] Adicionar nova regra para abrir porta 8080
- [ ] Verificar se a regra foi adicionada
- [ ] Deletar a nova regra

Comandos para teste:

```bash
```

```
# Verificar status do UFW
sudo ufw status

# Listar regras detalhadas
sudo ufw status verbose

# Adicionar regra para porta 8080
sudo ufw allow 8080

# Verificar se foi adicionada
sudo ufw status

# Remover regra
sudo ufw delete allow 8080
```

Explicações necessárias:

- [ ] O que é UFW e o valor de usá-lo

8. SSH

Testes a realizar:

- [ ] Verificar se o serviço SSH está instalado
- [ ] Verificar se está funcionando corretamente
- [ ] Verificar se o SSH usa apenas a porta 4242
- [ ] Fazer login SSH com o usuário recém-criado
- [ ] Verificar que NÃO é possível usar SSH com o usuário "root"

Comandos para teste:

```bash
```

```
# Verificar serviço SSH
sudo systemctl status ssh

# Verificar configuração da porta
sudo cat /etc/ssh/sshd_config | grep Port

# Testar conexão SSH (de outra máquina ou terminal)
ssh novo_usuario@localhost -p 4242

# Tentar login como root (deve falhar)
ssh root@localhost -p 4242
```

Explicações necessárias:

- [ ] O que é SSH e o valor de usá-lo

9. Script de Monitoramento

Testes a realizar:

- [ ] Verificar se o script roda a cada 10 minutos
- [ ] Modificar para rodar a cada 30 segundos
- [ ] Verificar se o script funciona com valores dinâmicos
- [ ] Parar o script sem modificá-lo
- [ ] Reiniciar o servidor e verificar se o script ainda existe no mesmo local
- [ ] Verificar se as permissões permaneceram inalteradas
- [ ] Verificar se o script não foi modificado

Comandos para teste:

```
bash
```

```
# Verificar cron jobs
sudo crontab -l

# Verificar se o script existe
ls -la /caminho/para/o/script

# Modificar intervalo do cron (exemplo para 30s - pode usar sleep)
# Editar crontab
sudo crontab -e

# Para parar o script
sudo systemctl stop cron
# ou comentar a linha no crontab

# Verificar permissões do arquivo
ls -la /caminho/para/o/script
```

Explicações necessárias:

- [ ] Como o script funciona (mostrar o código)
- [ ] O que é "cron"
- [ ] Como foi configurado para rodar a cada 10 minutos desde o início do servidor

Parte Bônus (apenas se a parte obrigatória estiver 100% correta)

10. Pontos Bônus

Verificar com o subject:

- [ ] **Configuração de partições (2 pontos)** - Implementação correta de partições extras
- [ ] **WordPress (2 pontos)** - Apenas com serviços requeridos pelo subject
- [ ] **Serviço de livre escolha (1 ponto)** - Explicar como funciona e por que é útil

Nota: NGINX e Apache2 são proibidos.

Checklist Final

- [ ] Todos os testes obrigatórios foram aprovados
- [ ] Todas as explicações foram satisfatórias
- [ ] O projeto funciona conforme especificado
- [ ] (Opcional) Bônus implementados corretamente

Comandos Úteis Adicionais

```bash
# Verificar todos os serviços em execução
sudo systemctl list-units --type=service --state=active

# Verificar informações do sistema
uname -a
whoami
id

# Verificar espaço em disco
df -h

# Verificar uso de memória
free -h

# Verificar processos
ps aux

# Verificar informações de rede
ip a
ss -tuln
```

Notas Importantes

1. Durante toda a defesa, você deve ser capaz de ajudar o avaliador a verificar qualquer ponto

2. Se algo não funcionar como esperado ou não for claramente explicado, a avaliação para
3. Mantenha-se educado, cortês e construtivo durante todo o processo
4. O script de monitoramento deve exibir informações a cada 10 minutos por padrão