



Born2beRoot

Resumo: Este documento é um exercício relacionado à Administração do Sistema.

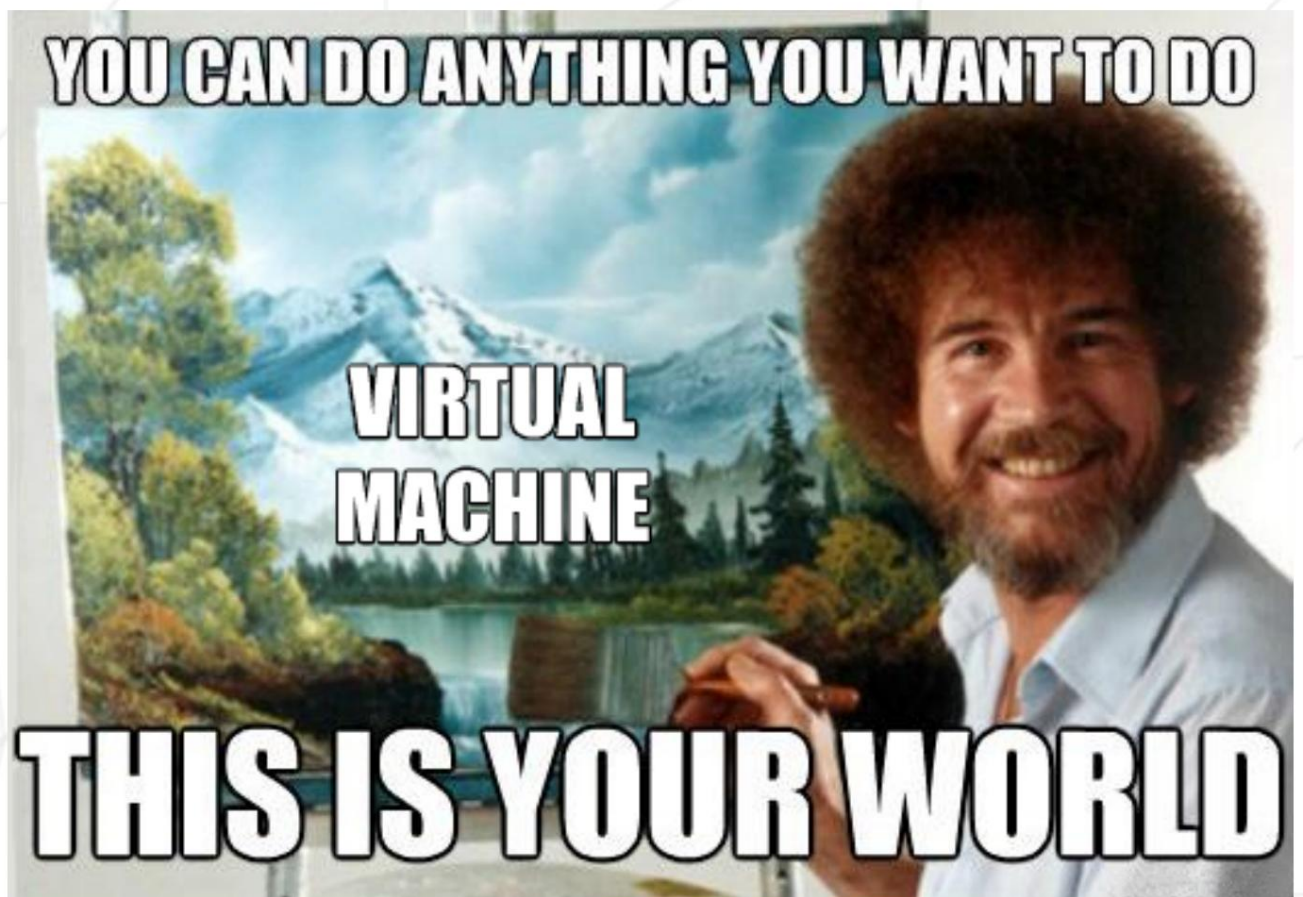
Versão: 3.1

Conteúdo

EU	Preâmbulo	2
II	Introdução	3
III	Diretrizes gerais	4
4	Parte obrigatória	5
	Parte bônus V	10
VI	Envio e avaliação por pares	12

Capítulo I

Preâmbulo



Capítulo II

Introdução

Este projeto tem como objetivo apresentar a você o maravilhoso mundo da virtualização.

Você criará sua primeira máquina no VirtualBox (ou UTM se não puder usar o VirtualBox) sob instruções específicas. Então, ao final deste projeto, você será capaz de configurar seu próprio sistema operacional enquanto implementa regras rígidas.

Capítulo III

Diretrizes gerais

- O uso do VirtualBox (ou UTM se você não puder usar o VirtualBox) é obrigatório.
- Você só precisa entregar um arquivo subscription.txt na raiz do seu repositório. Você deve colar nele a assinatura do disco virtual da sua máquina. Acesse Envio e avaliação por pares para obter mais informações.

Capítulo IV

Parte obrigatória

Este projeto consiste em configurar seu primeiro servidor seguindo regras específicas.



Por se tratar de configurar um servidor, você instalará o mínimo de serviços. Por esse motivo, uma interface gráfica é inútil aqui. É portanto proibido instalar o X.org ou qualquer outro servidor gráfico equivalente. Caso contrário, sua nota será 0.

Você deve escolher como sistema operacional a versão estável mais recente do Debian (sem teste/instável) ou a versão estável mais recente do Rocky. O Debian é altamente recomendado se você for novo na administração de sistemas.



Configurar Rocky é bastante complexo. Portanto, você não precisa configurar o KDUMP. Porém, o SELinux deve estar rodando na inicialização e sua configuração deve ser adaptada às necessidades do projeto. O AppArmor para Debian também deve estar em execução na inicialização.

Você deve criar pelo menos 2 partições criptografadas usando LVM. Abaixo está um exemplo do particionamento esperado:

```
wil@wil:~$ lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda                                  8:0    0   8G  0 disk
├─sda1                              8:1    0 487M  0 part  /boot
├─sda2                              8:2    0    1K  0 part
├─sda5                              8:5    0  7.5G  0 part
│ └─sda5_crypt                     254:0    0  7.5G  0 crypt
│   ├─wil--vg-root                 254:1    0  2.8G  0 lvm    /
│   ├─wil--vg-swap_1              254:2    0  976M  0 lvm    [SWAP]
│   └─wil--vg-home                 254:3    0  3.8G  0 lvm    /home
sr0                                  11:0    1 1024M  0 rom
```

wil@wil:~\$ _

Born2beRoot



Durante a defesa, serão feitas algumas perguntas sobre o sistema operacional escolhido. Por exemplo, você deve saber as diferenças entre aptitude e apt, ou o que é SELinux ou AppArmor. Resumindo, entenda o que você usa!

Um serviço SSH será executado apenas na porta 4242. Por razões de segurança, não deve ser possível conectar usando SSH como root.



O uso do SSH será testado durante a defesa através da criação de um novo conta. Você deve, portanto, entender como isso funciona.

Você tem que configurar seu sistema operacional com o UFW (ou firewalld para Rocky) firewall e, portanto, deixe apenas a porta 4242 aberta.



Seu firewall deve estar ativo quando você inicia sua máquina virtual. Para Rocky, você deve usar firewalld em vez de UFW.

- O nome do host da sua máquina virtual deve ser o seu login terminando em 42 (por exemplo, wil42). Você terá que modificar esse nome de host durante sua avaliação.
- Você precisa implementar uma política de senha forte.
- Você deve instalar e configurar o sudo seguindo regras estritas.
- Além do usuário root, um usuário com seu login como nome de usuário deve estar presente.
- Este usuário deve pertencer aos grupos user42 e sudo.



Durante a defesa, você terá que criar um novo usuário e atribuí-lo a um grupo.

Para configurar uma política de senha forte, você deve cumprir os seguintes requisitos:

comentários:

- Sua senha expira a cada 30 dias.
- O número mínimo de dias permitido antes da modificação de uma senha será
ser definido como 2.
- O usuário deverá receber uma mensagem de aviso 7 dias antes de sua senha expirar.
- Sua senha deve ter pelo menos 10 caracteres. Deve conter uma letra maiúscula, uma letra minúscula e um número. Além disso, não deve conter mais de 3 caracteres idênticos consecutivos.

Born2beRoot

- A senha não deve incluir o nome do usuário.
- A seguinte regra não se aplica à senha root: A senha deve ter pelo menos 7 caracteres que não façam parte da senha anterior.
- É claro que sua senha root deve estar em conformidade com esta política.



Após definir seus arquivos de configuração, você deverá alterar todas as senhas das contas presentes na máquina virtual, incluindo a conta root.

Para definir uma configuração forte para o seu grupo sudo, você deve cumprir o seguintes requisitos:

- A autenticação usando sudo deve ser limitada a 3 tentativas em caso de erro senha correta.
- Uma mensagem personalizada de sua escolha deverá ser exibida se ocorrer um erro devido a um erro a senha ocorre ao usar o sudo.
- Cada ação usando sudo deve ser arquivada, tanto as entradas quanto as saídas. O arquivo de registro deve ser salvo na pasta `/var/log/sudo/`.
- O modo TTY deve estar ativado por motivos de segurança.
- Também por razões de segurança, os caminhos que podem ser usados pelo sudo devem ser restritos.
Exemplo: /
`usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin`

Finalmente, você deve criar um script simples chamado `monitoramento.sh`. Deve ser desenvolvido em bash.

Na inicialização do servidor, o script exibirá algumas informações (listadas abaixo) em todos os terminais a cada 10 minutos (dê uma olhada na parede). A bandeira é opcional. Nenhum erro deve estar visível.

Seu script deve sempre ser capaz de exibir as seguintes informações:

- A arquitetura do seu sistema operacional e sua versão do kernel.
- O número de processadores físicos.
- O número de processadores virtuais.
- A RAM atualmente disponível em seu servidor e sua taxa de utilização como porcentagem.
- A memória atualmente disponível em seu servidor e sua taxa de utilização como porcentagem.
- A taxa de utilização atual dos seus processadores como uma porcentagem.
- A data e hora da última reinicialização.
- Se o LVM está ativo ou não.
- O número de conexões ativas.
- O número de usuários que utilizam o servidor.
- O endereço IPv4 do seu servidor e seu endereço MAC (Media Access Control).
- O número de comandos executados com o programa `sudo`.



Durante a defesa, você será solicitado a explicar como funciona esse script. Você também terá que interrompê-lo sem modificá-lo.
Dê uma olhada no cron.

Este é um exemplo de como o script deve funcionar:

```
Mensagem de transmissão de root@wil (tty1) (domingo, 25 de abril, 15:45:00 de 2021):
```

```
#Arquitetura: Linux wil 4.19.0-16-amd64 #1 SMP Debian 4.19.181-1 (2021-03-19) x86_64 GNU/Linux #CPU física: 1 #vCPU: 1

#Uso de memória: 74/987 MB (7,50%)
#Uso de disco: 1009/2 Gb (49%)
#Carga da CPU: 6,7%
#Última inicialização: 25/04/2021 14:45
#LVM uso: sim
#Conexões TCP: 1 ESTABELECIDO
#Log do usuário:
1 #Rede: IP 10.0.2.15 (08:00:27:51:9b:a5)
#Sudo: 42 cmd
```

Born2beRoot

Abaixo estão dois comandos que você pode usar para verificar alguns dos requisitos do assunto:

Para Rochoso:

```
[root@wil wil]# head -n 2 /etc/os-release
NAME="Rocky Linux"
VERSION="8.7 (Green Obsidian)"
[root@wil wil]# sestatus
SELinux status:                enabled
SELinuxfs mount:              /sys/fs/selinux
SELinux root directory:      /etc/selinux
Loaded policy name:           targeted
Current mode:                 enforcing
Mode from config file:       enforcing
Policy MLS status:           enabled
Policy deny_unknown status:   allowed
Memory protection checking:   actual (secure)
Max kernel policy version:    33
[root@wil wil]# ss -tunlp
Netid State  Recv-Q Send-Q Local Address:Port Peer Address:Port Process
tcp    LISTEN  0      128      0.0.0.0:4242  0.0.0.0:*      users:((("sshd",pid=28429,fd=6))
tcp    LISTEN  0      128      :::4242    :::*           users:((("sshd",pid=28429,fd=4))
[root@wil wil]# firewall-cmd --list-service
ssh
[root@wil wil]# firewall-cmd --list-port
4242/tcp
[root@wil wil]# firewall-cmd --state
running
[root@wil wil]# _
```

Para Debian:

```
root@wil:~# head -n 2 /etc/os-release
PRETTY_NAME="Debian GNU/Linux 10 (buster)"
NAME="Debian GNU/Linux"
root@wil:/home/wil# /usr/sbin/aa-status
apparmor module is loaded.
root@wil:/home/wil# ss -tunlp
Netid State  Recv-Q Send-Q Local Address:Port Peer Address:Port
tcp    LISTEN  0      128      0.0.0.0:4242  0.0.0.0:*      users:((("sshd",pid=523,fd=3))
tcp    LISTEN  0      128      :::4242    :::*           users:((("sshd",pid=523,fd=4))
root@wil:/home/wil# /usr/sbin/ufw status
Status: active

To Action From
--
4242 ALLOW Anywhere
4242 (v6) ALLOW Anywhere (v6)
```

Capítulo V

Parte bônus

Lista de bônus:

- Configure as partições corretamente para obter uma estrutura semelhante à abaixo:

```
# lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda                                  8:0    0 30.8G  0 disk
├─sda1                              8:1    0   500M  0 part  /boot
├─sda2                              8:2    0     1K  0 part
├─sda5                              8:5    0 30.3G  0 part
│   └─sda5_crypt                    254:0    0 30.3G  0 crypt
│       ├─LVMGroup-root              254:1    0   10G  0 lvm    /
│       ├─LVMGroup-swap              254:2    0   2.3G  0 lvm    [SWAP]
│       ├─LVMGroup-home              254:3    0     5G  0 lvm    /home
│       ├─LVMGroup-var               254:4    0     3G  0 lvm    /var
│       ├─LVMGroup-srv               254:5    0     3G  0 lvm    /srv
│       ├─LVMGroup-tmp               254:6    0     3G  0 lvm    /tmp
│       └─LVMGroup-var--log          254:7    0     4G  0 lvm    /var/log
sr0                                  11:0    1 1024M  0 rom
```

- Configure um site WordPress funcional com os seguintes serviços: lighttpd, MariaDB e PHP.
- Configure um serviço de sua escolha que você considere útil (ex-NGINX / Apache2 incluído!). Durante a defesa você terá que justificar sua escolha.



Para completar a parte bônus, você tem a possibilidade de configurar serviços extras. Neste caso, você pode abrir mais portas para atender às suas necessidades.

É claro que as regras do UFW/Firewalld devem ser adaptadas de acordo.

Born2beRoot



A parte bônus só será avaliada se a parte obrigatória for PERFEITA. Perfeito significa que a parte obrigatória foi feita integralmente e funciona sem mau funcionamento. Se você não passou em TODOS os requisitos obrigatórios, sua parte do bônus não será avaliada de forma alguma.

Capítulo VI

Envio e avaliação por pares

Você só precisa entregar um arquivo `subscription.txt` na raiz do seu repositório Git. Você deve colar nele a assinatura do disco virtual da sua máquina. Para obter esta assinatura, primeiro você precisa abrir a pasta de instalação padrão (é a pasta onde suas VMs são salvas):

- Windows: `%HOMEDRIVE%%HOMEPATH%\VirtualBox VMs\`
- Linux: `~/VMs do VirtualBox/`
- MacM1: `~/Library/Containers/com.utmapp.UTM/Data/Documents/`
- MacOS: `~/VMs do VirtualBox/`

Em seguida, recupere a assinatura do arquivo `".vdi"` (ou `".qcow2"` para usuários UTM) de sua máquina virtual no formato sha1. Abaixo estão 4 exemplos de comandos para um arquivo `rocky_serv.vdi`:

- Windows: `certUtil -hashfile rocky_serv.vdi sha1`
- Linux: `sha1sum rocky_serv.vdi`
- Para Mac M1: `shasum rocky.utm/Images/disk-0.qcow2`
- MacOS: `shasum rocky_serv.vdi`

Este é um exemplo de que tipo de saída você obterá:

- `6e657c4619944be17df3c31faa030c25e43e40af`



Observe que a assinatura da sua máquina virtual pode ser alterada após a sua primeira avaliação. Para resolver esse problema, você pode duplicar sua máquina virtual ou usar o estado de salvamento.



É claro que é PROIBIDO entregar sua máquina virtual em seu repositório Git. Durante a defesa, a assinatura do arquivo `subscription.txt` será comparada com a da sua máquina virtual. Se os dois não forem idênticos, sua nota será 0.

Born2beRoot



```
0010 01 11 111 001 000 11 01 10 1 0000 01 1 1010 111 11 0 000  
011 00 1 0000 1 0000 0 01 0100 1 0 010 10 01 1 0 0001 0 010 000  
00 111 10 111 0010 001100 001100 001100
```