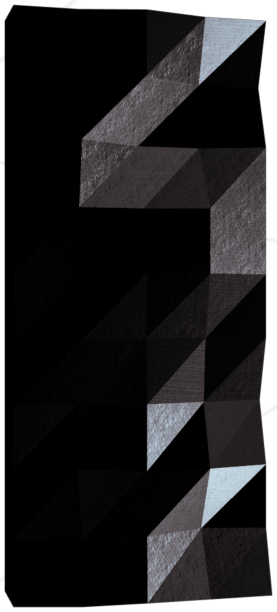


Born2beRoot

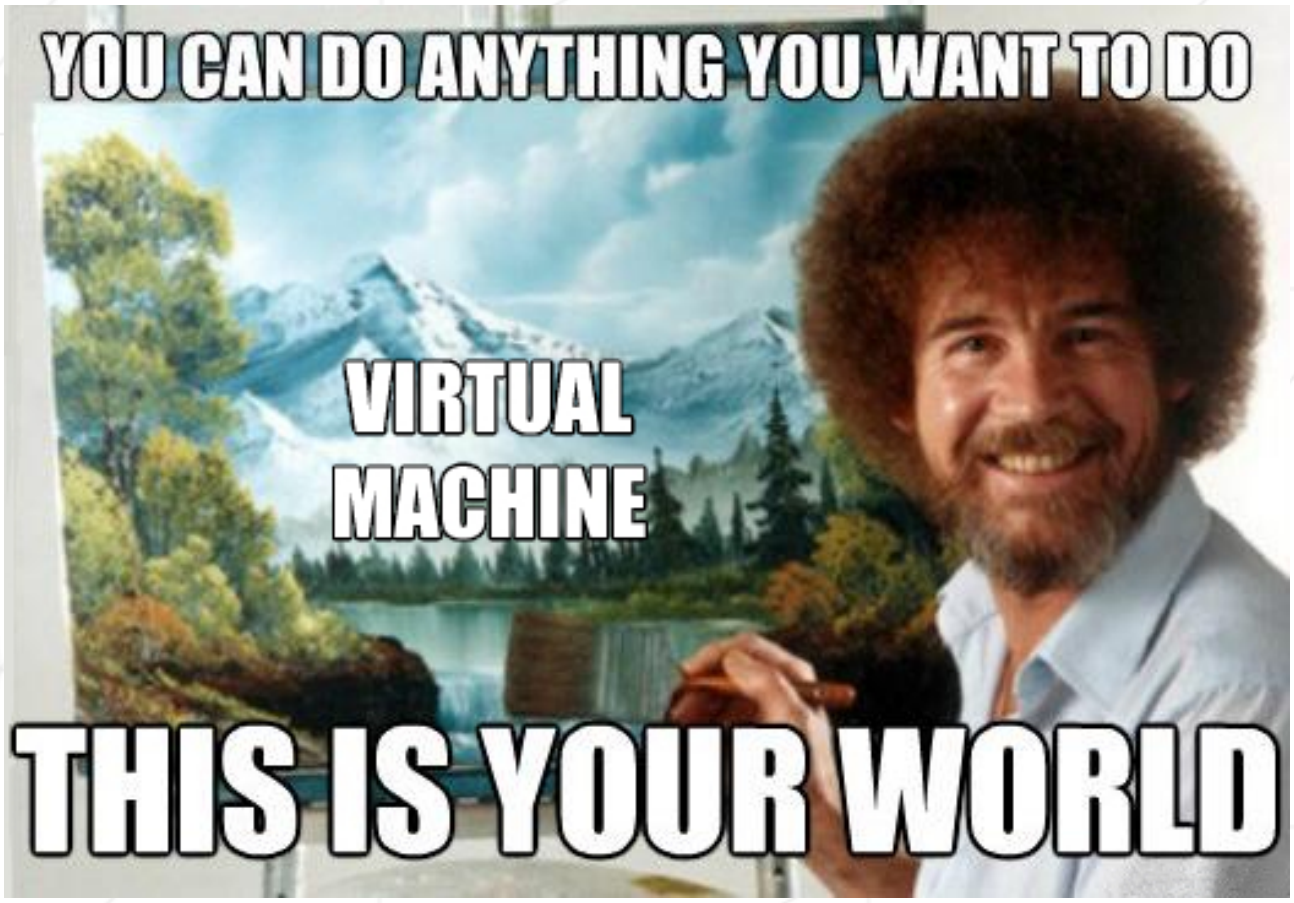
Preâmbulo: Este documento é um exercício relacionado à administração de sistemas.

Versão: 4

Sumário

Capítulo I

Preâmbulo



Capítulo II

Introdução

Este projeto visa introduzir você ao maravilhoso mundo da virtualização.

Você criará sua primeira máquina em `VirtualBox` (ou `UTM` se você não puder usar `VirtualBox`) usando instruções específicas. Então, ao final deste projeto, você será capaz de configurar seu próprio sistema operacional implementando regras estritas.

Capítulo III

Instruções gerais

- O uso do `VirtualBox` (ou `UTM` se você não puder usar `VirtualBox`) é obrigatório.
- Você só precisa enviar um arquivo `signature.txt` na raiz do seu repositório. Você deve colar nele a assinatura do disco virtual da sua máquina. Acesse a seção "Entrega e avaliação por pares"para mais informações.
- O uso de snapshots é proibido.

Capítulo IV

Instruções de IA

Contexto

Este projeto foi desenvolvido para ajudá-lo a descobrir os blocos de construção fundamentais do seu treinamento em TIC.

Para ancorar adequadamente os conhecimentos e habilidades-chave, é essencial adotar uma abordagem criteriosa ao uso de ferramentas e suporte de IA.

A verdadeira aprendizagem fundamental exige um esforço intelectual genuíno — através de desafios, repetição e trocas de aprendizagem entre pares.

Para uma visão geral mais completa de nossa posição sobre a IA — como ferramenta de aprendizagem, como parte do currículo de TIC e como expectativa no mercado de trabalho — consulte as perguntas frequentes dedicadas na intranet.

Mensagem principal

- Construa bases sólidas sem atalhos.
- Desenvolva verdadeiramente habilidades técnicas e de poder.
- Experimente a verdadeira aprendizagem entre pares, comece a aprender como aprender e resolver novos problemas.
- A jornada de aprendizagem é mais importante que o resultado.
- Aprenda sobre os riscos associados à IA e desenvolva práticas eficazes de controle e contramedidas para evitar armadilhas comuns.

Regras para o aluno:

- Você deve aplicar o raciocínio às suas tarefas atribuídas, especialmente antes de recorrer à IA.
- Você não deve pedir respostas diretas à IA.
- Você deve aprender sobre a abordagem global da 42 em relação à IA.

Resultados da fase:

Nesta fase fundamental, você obterá os seguintes resultados:

- Obter bases sólidas em tecnologia e codificação.
- Saber por que e como a IA pode ser perigosa durante esta fase.

Comentários e exemplo:

- Sim, sabemos que a IA existe — e sim, ela pode resolver seus projetos. Mas você está aqui para aprender, não para provar que a IA aprendeu. Não perca seu tempo (nem o nosso) apenas para demonstrar que a IA pode resolver o problema dado.
- Aprender na 42 não é sobre saber a resposta — é sobre desenvolver a capacidade de encontrar uma. A IA lhe dá a resposta diretamente, mas isso o impede de construir seu próprio raciocínio. E o raciocínio leva tempo, esforço e envolve falhas. O caminho para o sucesso não deve ser fácil.
- Lembre-se de que durante os exames, a IA não estará disponível — sem internet, sem smartphones, etc. Você perceberá rapidamente se confiou demais na IA em seu processo de aprendizagem.
- A aprendizagem entre pares o expõe a diferentes ideias e abordagens, melhorando suas habilidades interpessoais e sua capacidade de pensar de forma divergente. Isso é muito mais valioso do que apenas conversar com um bot. Então não seja tímido — converse, faça perguntas e aprenda juntos!
- Sim, a IA fará parte do currículo — tanto como ferramenta de aprendizagem quanto como um tópico em si. Você até terá a chance de construir seu próprio software de IA. Para saber mais sobre nossa abordagem crescente, consulte a documentação disponível na intranet.

✓ Boa prática:

> Estou travado em um novo conceito. Pergunto a alguém próximo como ele abordou isso. Conversamos por 10 minutos — e de repente, clica. Entendi.

✗ Má prática:

> Uso secretamente a IA, copio algum código que parece certo. Durante a avaliação entre pares, não consigo explicar nada. Eu falho. Durante o exame — sem IA — estou travado novamente. Eu falho.

Capítulo V

Parte obrigatória

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

Como se trata da configuração de um servidor, você instalará o mínimo de serviços. Por esse motivo, uma interface gráfica é inútil aqui. Portanto, é proibido instalar 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 testing/unstable), ou a versão estável mais recente do `Rocky`. `Debian` é altamente recomendado se você é iniciante em administração de sistemas.

Configurar o Rocky é bastante complexo. Portanto, você não precisa configurar o KDump. No entanto, o SELinux deve estar rodando na inicialização e sua configuração precisa ser adaptada para as necessidades do projeto. O AppArmor para Debian também deve estar rodando na inicialização.

Você deve criar pelo menos 2 partições criptografadas usando `LVM`. Abaixo está um exemplo de uma possível partição:

```
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:~$ _
```

Durante a defesa, você será questionado sobre o sistema operacional que escolheu. Por exemplo, você deve conhecer as diferenças entre aptitude e apt, ou o que é SELinux ou AppArmor. Em resumo, entenda o que você usa!

Um serviço SSH estará rodando obrigatoriamente na porta 4242 em sua máquina virtual. Por razões de segurança, não deve ser possível conectar usando SSH como root.

O uso do SSH será testado durante a defesa configurando uma nova conta. Você deve, portanto, entender como ele funciona.

Você precisa configurar seu sistema operacional com o firewall UFW (ou firewalld para Rocky) e, assim, deixar apenas a porta 4242 aberta em sua máquina virtual.

O exemplo mostra tamanhos de disco arbitrários. Você precisa determinar o tamanho apropriado para cada partição para garantir o funcionamento adequado, evitando o uso desnecessário de disco.

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

- O hostname de sua máquina virtual deve ser seu login terminando em 42 (por exemplo, wil42). Você terá que modificar este hostname durante sua avaliação.
- Você precisa implementar uma política de senhas forte.
- Você precisa 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:

- Sua senha deve expirar a cada 30 dias.

- O número mínimo de dias permitidos antes da modificação de uma senha será definido como 2.
- O usuário deve receber uma mensagem de aviso 7 dias antes da expiração de sua senha.
- 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.
- 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 fazem parte da senha anterior.
- Claro, sua senha root deve cumprir esta política.

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

Para configurar uma configuração forte para seu grupo `sudo`, você deve cumprir os seguintes requisitos:

- A autenticação usando `sudo` deve ser limitada a 3 tentativas em caso de senha incorreta.
- Uma mensagem personalizada de sua escolha deve ser exibida se ocorrer um erro devido a uma senha incorreta ao usar o `sudo`.
- Cada ação usando `sudo` deve ser arquivada, tanto inputs quanto outputs. O arquivo de log deve ser salvo na pasta `/var/log/sudo/`.
- O modo `TTY` deve ser habilitado por motivos de segurança.
- Também por motivos 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 precisa criar um script simples chamado `monitoring.sh`. Ele deve ser desenvolvido em `bash`.

Na inicialização do servidor, o script exibirá algumas informações (listadas abaixo) em todos os terminais a cada 10 minutos (veja `wall`). O banner é opcional. Nenhum erro deve ser visível.

Seu script deve sempre 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 disponível atualmente em seu servidor e sua taxa de utilização como porcentagem.
- O armazenamento disponível atualmente em seu servidor e sua taxa de utilização como porcentagem.
- A taxa de utilização atual de seus processadores como porcentagem.
- A data e hora da última reinicialização.
- Se o LVM está ativo ou não.
- O número de conexões ativas.
- O número de usuários usando 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 esse script funciona. Você também terá que interrompe-lo sem modificá-lo. Veja `cron`.

Este é um exemplo de como o script deve funcionar:

```
Mensagem de broadcast de root@wil (tty1) (dom 25 abr 15:45:00 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: 74/987MB (7,50%)
    #Uso de disco: 1009/2Gb (49%)
    #Carga da CPU: 6,7%
    #Última inicialização: 2021-04-25 14:45
    #Uso de LVM: sim
    #Conexões TCP: 1  ESTABLISHED
    #Log de usuário: 1
    #Rede: IP 10.0.2.15 (08:00:27:51:9b:a5)
    #Sudo: 42 cmd
```

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

Para `Rocky`:

```
[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 VI

Parte bônus

Lista de bônus:

- Configurar as partições corretamente para que você obtenha 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
```

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

O exemplo mostra tamanhos de disco arbitrários. Você precisa determinar o tamanho apropriado para cada partição para garantir o funcionamento adequado, evitando o uso desnecessário de disco.

Para completar a parte bônus, você tem a possibilidade de configurar serviços extras. Nesse caso, você pode abrir mais portas para atender às suas necessidades. Claro, as regras do UFW/Firewalld precisam ser adaptadas de acordo.

A parte bônus só será avaliada se a parte obrigatória estiver PERFEITA. Perfeita significa que a parte obrigatória foi integralmente feita e funciona sem mau funcionamento. Se você não tiver passado TODOS os requisitos obrigatórios, sua parte bônus não será avaliada.

Capítulo VII

Entrega e avaliação por pares

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

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

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 do tipo de saída que você obterá:

- `6e657c4619944be17df3c31faa030c25e43e40af`

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

É claro que é PROIBIDO enviar sua máquina virtual em seu repositório Git. Durante a defesa, a assinatura do arquivo signature.txt será comparada com a de sua máquina virtual. Se as duas não forem idênticas, sua nota será 0.

O uso de snapshots é PROIBIDO. Durante a defesa, se algum snapshot for detectado, sua nota será 0.

Durante a avaliação, uma breve **modificação do projeto** pode ser ocasionalmente solicitada. Isso pode envolver uma pequena mudança de comportamento, algumas linhas de código para escrever ou reescrever, ou um recurso fácil de adicionar.

Embora esta etapa possa **não ser aplicável a todos os projetos**, você deve estar preparado para ela se for mencionada nas diretrizes de avaliação.

Esta etapa visa verificar sua compreensão real de uma parte específica do projeto. A modificação pode ser realizada em qualquer ambiente de desenvolvimento que você escolher (por exemplo, sua configuração usual), e deve ser viável em poucos minutos — a menos que um prazo específico seja definido como parte da avaliação.
Você pode, por exemplo, ser solicitado a fazer uma pequena atualização em uma função ou script, modificar uma exibição ou ajustar uma estrutura de dados para armazenar novas informações, etc.

Os detalhes (escopo, alvo, etc.) serão especificados nas **diretrizes de avaliação** e podem variar de uma avaliação para outra para o mesmo projeto.

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