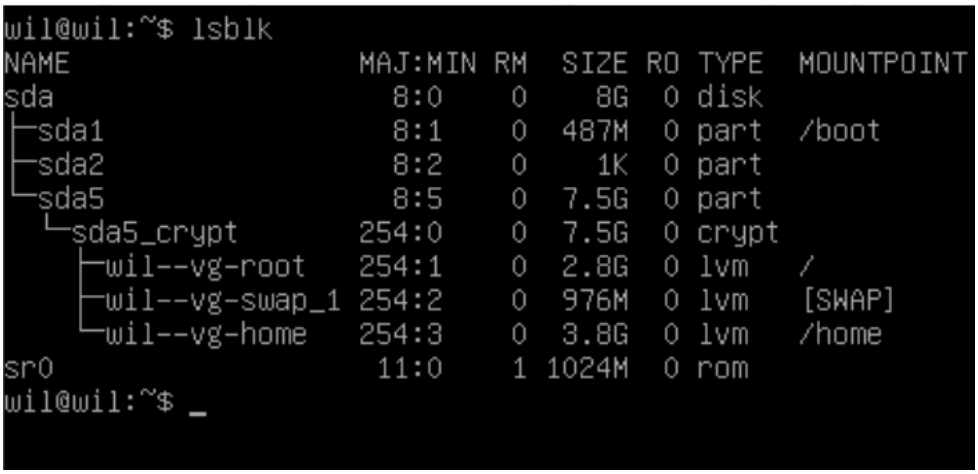
born2beroot

Requisitos

- Usar VirtualBox ou UTM para criar a VM
- Deve ser entregue apenas o arquivo signature.txt que precisa conter a assinatura do disco da máquina virtual
- Uma interface gráfica não deve ser instalada
- Escolher entre a última versão estável do Debian ou do CentOS
- SELinux deve rodar ao iniciar a VM e deve estar configurado para as necessidades da VM
- AppArmor deve rodar ao iniciar a VM, se o SO escolhido for Debian
- Pelo menos duas partições criptografadas devem ser criadas utilizando LVM
    - ▼ exemplo



- Serviço SSH deve estar rodando na porta 4242
- Não deve ser possível conectar como root usando SSH
- Configurar o sistema utilizando UFW Firewall para deixar apenas a porta 4242 aberta
- O hostname da VM deve ser login42
- Deve ser implementado uma política forte de senhas
  - A senha deve expirar a cada 30 dias
  - O minimo de dias necessários para modificar uma senha são 2
  - O usuário deve receber uma mensagem 7 dias antes da senha expirar
  - A senha deve ter pelo menos 10 caracteres, deve conter uma letra maiúscula, uma minúscula e um número e não deve conter mais de 3 caracteres iguais consecutivamente.
  - A senha não pode incluir o nome do usuário
  - A senha deve conter pelo menos 7 caracteres que não são parte da senha antiga
    - não aplicável para o usuário root
- Sudo deve ser instalado seguindo regras específicas
  - Autenticação utilizando o sudo deve ser limitada a 3 tentativas, no caso de senha errada
  - Uma mensagem customizada deve aparecer no caso de senha incorreta
  - Toda ação do sudo deve ser registrada (tanto inputs quanto outputs), na pasta /var/log/sudo/
  - o modo TTY deve estar ativado por motivos de segurança
  - O path utilizado pelo sudo deve ser restringido, por motivos de segurança
    - /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin
- Além do usuário root, um usuário com o login deve ser criado
  - Este usuário deve pertencer ao grupo root e user42
- Criar um script via bash chamado monitoring.sh
  - ao iniciar a máquina o script deve mostrar certas informações a cada 10 minutos em todos os terminal
    - Arquitetura do SO e a versão do Kernel
    - O número de processadores físicos
    - O número de processadores virtuais

        - A memória RAM disponível no servidor e sua utilização em porcentagem
        - A memória disponível no servidor e sua utilização em porcentagem
        - O grau de utilização dos processadores em porcentagem
        - O dia e hora do ultimo reboot
        - Se o LVM está ativo ou não
        - O número de conexões ativas
        - O número de usuários utilizando o servidor
        - O endereço IPv4 e MAC do servidor
        - O número de comandos executados com o sudo
    - Será necessário interromper o script sem alterá-lo → Cron

Máquina Virtual

O que é uma máquina virtual

> As máquinas virtuais são computadores de software com a mesma funcionalidade que os computadores físicos (programa que simula um ambiente computacional), executam aplicativos e um sistema operacional. Funciona como um sistema de computação com sua própria CPU, memória, interface de rede e armazenamento.

💡 **guest:** máquina virtual

💡 **host:** máquina física

Por que usar máquinas virtuais

> É útil para testar recursos em computadores. O software dentro da VM não pode adulterar o computador host. Vários ambientes de sistema operacional podem ser executados em um único computador físico

▼ fontes

- https://www.vmware.com/br/topics/glossary/content/virtual-machine.html
- https://tecnoblog.net/responde/o-que-e-uma-maquina-virtual/
- https://www.redhat.com/pt-br/topics/virtualization/what-is-a-virtual-machine

Sistema operacional Linux

Debian

- Debian tem uma base de usuários mais ampla, e é para uso tanto em situações comerciais e por usuários domésticos.

- Alta frequência de atualização, menos estável que o CentOS.
- Possui pacotes mais atualizados e uma maior quantidade de pacotes.
- Mais facilidade para encontrar fóruns de ajuda.
- Interface gráfica mais amigável.
- Gerenciador de pacote é .deb

CentOS

- CentOS é baseado no Red Hat Enterprise Linux e é voltado para o mercado comercial (fácil configuração).
- Possui pouca frequência de atualizações, mais estável.
- Menor quantidade de pacotes oferecidos.
- Interface gráfica menos intuitiva.
- Gerenciador de pacote é .rpm

💡 CentOS é mais popular para servidores

💡 Debian é mais seguro e estável

▼ fontes

- http://ptcomputador.com/Sistemas/linux/204305.html
- https://www.openlogic.com/blog/centos-vs-debian
- https://www.educba.com/centos-vs-debian/

LVM

> LVM é um gerenciador de discos do Kernel do Linux. Cria volumes lógicos no disco rígido, é um método de alocação de espaço. Permite redimensionar partições de disco em uso (o que só pode ser feito em partições livres ou fora de uso com outros métodos). O LVM tem como objetivo permitir uma grande flexibilidade para o administrador no gerenciamento de discos.

💡 **Volume físico** representa um único disco rígido em um sistema de computador

> Discos rígidos físicos podem ser subdivididos em unidades ou volumes lógicos múltiplos

Volumes lógicos são grupos de informações localizadas nos volumes físicos

> O processo pelo qual o sistema operacional delineia os limites lógicos de um disco rígido é chamado de particionamento

💡 **Kernel:** é o componente principal de um sistema operacional Linux e a interface central entre o hardware e os processos executados por um computador. Ele estabelece a comunicação entre ambos

💡 swap → memória virtual utilizada quando sobressai a memória RAM, porém muito lenta

- evita que o sistema trave
- para vazamento de memória

▼ fontes

- https://omegalaboratorios.com.br/qual-e-a-diferenca-entre-volumes-logicos-e-fisicos-em-discos-rigidos/
- https://www.redhat.com/pt-br/topics/linux/what-is-the-linux-kernel
- https://www.certificacaolinux.com.br/logical-volume-manager-lvm-no-linux/

APT and Aptitude

> São ferramentas para gerenciamento de pacotes no LInux. Ambas são capazes de manipular instalações, remoções, buscas etc. Foram inicialmente criadas para Debian mas hoje funcionam com rpm também.

💡 **Pacotes:** *são as peças que formam todas as distribuições Linux e podem conter programas, bibliotecas de sistema ou mesmo coisas como papéis de parede e ícones*

APT

> É inteiramente feito por linha de comando, não possui GUI. Encontra o pacote e suas dependências, instalando todos sem a necessidade do usuário se preocupar com esses requisitos extras de instalação. É altamente flexível, permite que o usuário configure ações de busca e instalação facilmente.

Aptitude

> É um gerenciador que abstrai detalhes de baixo nível e pode operar tanto por linha de comando quanto por interface interativa. Lida com mais funcionalidades que o apt, possui um gerenciador de pacotes melhor (ex: remoção automática de pacotes não utilizados).

Comparação

> O Apt-get só trabalha com linha de comando, o Aptitude possui tanto a linha de comando quanto uma interface gráfica no terminal. O Aptitude é mais intuitivo e de mais fácil utilização mas o Apt possui uma busca de pacotes mais completa (utiliza o apt-cache), além disso, por vir como padrão no Debian é mais usado pela comunidade, por isso tem mais fóruns de ajuda.

▼ fontes

- https://www.hardware.com.br/artigos/instalando-aplicativos-linux/pacotes.html#:~:text=Morimoto%2C “os pacotes são as,papéis de parede e ícones.
- https://www.tecmint.com/difference-between-apt-and-aptitude/#:~:text=While apt-get handles all,marking a package to be

TTY

> Em computação, TTY é um comando no Unix e sistemas operacionais tipo Unix que imprime o nome de arquivo do terminal conectado à entrada padrão. Pode se referir a um dispositivo de input (como uma porta serial) ou ao TTY virtual, que permite que os usuários interajam com o sistema. É um subsistema no LInux e Unix que gerencia processos e sessões. Sempre que se usa um emulador de terminal ou algum tipo de shell se esta interagindo com TTYs virtuais.

▼ fontes

- https://itsfoss.com/what-is-tty-in-linux/
- https://www.vivendobauru.com.br/o-que-e-o-tty-linux/
- https://linuxtect.com/what-is-tty-and-how-to-use-it-in-linux/

Comandos

Ligar e desligar a máquina

```
# reinicia a máquina (permissão root necessária)
reboot

# desliga a máquina (permissão root necessária)
poweroff
```

Sair de uma sessão

```
#permite sair de uma sessão para entrar em outra

logout or exit
```

Informações sobre o particionamento do disco

```
lsblk
```

Informações sobre o sistema operacional

- arquivo **/etc/osrelease**

Segurança

SELinux

> é uma arquitetura de segurança para sistemas Linux que permite aos administradores mais controle sobre quem pode acessar o sistema. Define controles de acesso para aplicações, processos e arquivos. Ele usa políticas de segurança, um conjunto de regras que dizem ao SELinux o que pode ou não ser acessado, para impor o acesso permitido por uma determinada política.
>
> > Quando uma aplicação ou processo, também conhecidos como entidade, solicita acesso a um objeto, como um arquivo, o SELinux executa uma verificação com um cache de vetor de acesso (AVC), local onde as permissões para entidades e objetos ficam armazenadas. É mais seguro na limitação de processos que o AppArmor, porque tem uma política mais complexa. É muito menos amigável para utilização justamente por ser mais seguro e ter uma configuração mais complexa.

▼ fontes

- https://www.redhat.com/pt-br/topics/linux/what-is-selinux#:~:text=O Security-Enhanced Linux (SELinux,quem%20pode%20acessar%20o%20sistema.

AppArmor

> AppArmor é um sistema de *Controle de Acesso Mandatório* (MAC - Mandatory Access Control). O kernel consulta o AppArmor antes de cada chamada do sistema para saber se o processo está autorizado a fazer a operação dada. As regras de segurança aplicadas dependem do caminho de instalação (path) do programa (hardlinks referenciando um objeto podem não sofrer as mesmas restrições que o objeto). Uso mais simples que o SELinux mas um pouco menos seguro.

▼ fontes

- https://debian-handbook.info/browse/pt-BR/stable/sect.apparmor.html
- https://archive.is/VbER4

```
# Escolhendo Debian, use AppArmor

# instala o apparmor (em Debian já vem instalado por padrão)
sudo apt-get install apparmor

# ativa ou desativa o apparmor -> deve estar ativo
sudo systemctl enable/disable apparmor

# lista todos os perfis e status do apparmor
sudo aa-status
```

UFW Firewall

> **Firewall** é um dispositivo de segurança da rede que monitora o tráfego de rede de entrada e saída e decide permitir ou bloquear tráfegos específicos de acordo com um conjunto definido de regras de segurança.
>
> > A conexão entre sistemas de TI ou deles com a internet é comum nas empresas, essa conexão expõe os usuários a perigos. O firewall atua como um filtro entre um dispositivo e sua conexão com a internet ou com a rede externa.

> **Uncomplicated Firewall (UFW)** é uma aplicação que tem como objetivo setar regras para monitorar o tráfego de informações e dados do computador para um network. O UFW é uma solução mais simples que não perde na eficiência de segurança

▼ fontes

- https://www.cisco.com/c/pt_br/products/security/firewalls/what-is-a-firewall.html
- https://flowti.com.br/blog/o-que-e-firewall-e-qual-e-a-sua-importancia
- https://www.swhosting.com/en/comunidad/manual/que-es-el-firewall-ufw-y-como-configurarlo-en-linux
- https://www.devmedia.com.br/ufw-firewall-do-ubuntu/18317

```
 # instala o UFW
sudo apt-get install ufw

# verifica o status do UFW e mostra as regras ativas -> mostra as portas liberadas
sudo ufw status verbose

# ativa/desativa o ufw -> deve estar ativado
sudo ufw enable/disable

# permite/bloqueia a entrada em todas as portas -> só a porta 4242 deve estar aberta
sudo ufw default allow/deny incoming

# permite/bloqueia a conexão em uma porta específica -> deve ser habilitado para a porta 4242
sudo ufw allow/deny <número da porta>

# permite/bloqueia a saída em todas as portas -> não mexer
# package manager and other essencial applications will stop working
sudo ufw default allow/deny outgoing

# habilitar/desabilitar portas cria regras para o UFW seguir
#mostra as regras com seu número correspondente
sudo ufw status numbered
# deleta a regra
sudo ufw delete <número da regra>
```

SSH

> É um protocolo que garante que cliente e servidor remoto troquem informações de maneira segura e dinâmica, o que permite a transferência de dados sem nenhuma perda de informação. Somente dois pontos acessam as informações: o servidor e o computador que enviou os dados para esse local remoto.

▼ fontes

- https://rockcontent.com/br/blog/ssh/

```
# instala o servidor e o cliente SSH
sudo apt-get install openssh-server openssh-client

# verifica o status do serviço SSH
sudo service ssh status

# ativa/desativa o serviço SSH -> deve estar ativado
sudo service ssh start/stop

# para alterar a porta padrão do SSH e negar o acesso do root é necessário editar o arquivo /etc/ssh/sshd_config
  # utilizando vim, necessário instalá-lo
  sudo apt-get install vim
  vim /etc/ssh/sshd_config
    # alterando a linha para alterar a porta padrão
      Port 22 -> Port 4242
    # alterando a linha para negar acesso ao root pelo SSH
      PermitRootLogin prohibit-password -> PermitRootLogin no

# necessário dar um reset no SSH para aplicar as mudanças
  sudo service ssh restart

# verifica se o ssh é a única conexão socket disponível no servidor
ss -tunlp

# é possível que a conexão com a internet seja classificada como socket
# para resolver isso é preciso definir o IP da máquina como static
  # instalar net-tools (permite a utilização do comando ifconfig) -> necessário para extrair informações da máquina
    sudo apt-get install net-tools
    # IP & netmask
      sudo ifconfig | grep "inet"
    # gateway
      sudo route -n
# edite o arquivo /etc/network/interfaces
  iface enp0s3 inet dhcp -> iface enp0s3 inet static
# insira as seguintes informações
  iface enp0s3 inet static
    address <ip da máquina> 192.168.15.82
    netmask <netmask da máquina> 255.255.255.0
    gateway <gateway da máquina> 192.168.15.1

# reboot pra validar as alterações
```

Testando SSH

```
# em um terminal fora da VM
ssh [usuário da VM]@[ip da VM] -p [porta da conexão]

# envia arquivos pro servidor por meio da SSH
scp -P [porta da conexão] [arquivo] [usuário da VM]@[ip da VM]:diretório que vai receber o arquivo]

# verifica o IP do servidor
ip addr show

# sair da conexão SSH
logout or exit
```

Sudo

Su, sudo e sudoers

> **Su:** chama o superusuário (entra no root).

> **Sudo:** permite que usuários sem acesso *root* executem comandos que normalmente precisariam de privilégios de super usuário. Atribuir poderes de administrador de maneira passageira para um determinado comando para que seja efetuada uma tarefa. Depois de terminado o processo, os poderes de administrador se encerram até que o sudo seja invocado novamente. Exemplo: `sudo apt-get upgrade`

> **Sudoers:** Dá instruções para o sistema sobre como lidar com o comando sudo. É um arquivo que indica quais tarefas os usuários podem realizar no sistema

▼ fontes

- https://sempreupdate.com.br/qual-a-diferenca-entre-su-e-sudo/
- https://sempreupdate.com.br/comando-su-o-que-e-para-que-serve-confira-alguns-exemplos/

```
# verifica se o sudo já existe no sistema
sudo --version

# instala o sudo
apt-get install sudo

# necessário editar o arquivo /etc/sudoers para adicionar as regras pedidas no subject
  # abre o arquivo
    sudo visudo

  # configura as regras
    # Limite de tentativas de senha = 3
      Defaults    passwd_tries=3

    # Mensagem personalizada de erro de senha
      Defaults    badpass_message="<mensagem de erro>"

    # Define onde salvar as mensagens de log (diretório /var/log/sudo)
      # é necessário criar esse diretório primeiro
        mkdir /var/log/sudo
      Defaults    logfile=/var/log/sudo/sudo.log

    # Habilita o TTY
      Defaults    requiretty

    # Define um path extra para o sudo, incluindo o /snap/bin no final
      Defaults    secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Exibe os usuários do sistema
  less /etc/passwd
  # cada linha contém 7 colunas
    # 1 : nome do usuário
    # 2 : senha criptografada
    # 3 : ID do usuário (UID)
    # 4 : ID do grupo do usuário (GID)
    # 5 : Nome completo do usuário
    # 6 : Home (diretório) do usuário
    # 7 : Login shell

# Exibe todos os grupos do sistema e seus usuários
  less /etc/group

# adiciona um usuário ao grupo sudo
  gpasswd -a <username> sudo

# remove um usuário do grupo sudo
  gpasswd -d <username> sudo

# verifica os grupos que um usuário específico participa
  groups <username>
```

Política de senha

```
# Regras aplicadas ao editar o arquivo /etc/login.defs
  # A senha deve expirar a cada 30 dias
    PASS_MAX_DAYS 30
  # O minimo de dias necessários para modificar a senha são 2
    PASS_MIN_DAYS 2
  # O usuário deve receber uma mensagem 7 dias antes da senha expirar
    PASS_WARN_AGE 7

# Para Ativar as novas regras aos usuários já criados
  # Número de dias para a senha expirar
    chage -M 30 <usuário>
  # Número mínimo de dias para trocar a senha
    chage -m 2 <usuário>
  # Número de dias para receber a mensagem de senha expirando
    chage -W 7 <usuário>
  # Mostra as regras aplicadas no usuário
    chage -l <usuário>

# Regras aplicadas a partir do pacte pam-pwquality
  # A senha deve conter uma letra maiúscula, uma minúscula e um número e não deve conter mais de 3 caracteres iguais consecutivamente.
  # A senha não pode incluir o nome do usuário
  # A senha deve conter pelo menos 7 caracteres que não são parte da senha antiga
      # Não aplicável ao root, por padrão
  sudo apt-get install libpam-pwquality
    # Editar o arquivo /ETC/PAM.D/COMMON-PASSWORD
      # Mínimo de 10 caracteres
        minlen = 10
      # Mínimo de 1 caracter maiúsculo
        ucredit = -1
      # Mínimo de 1 carácter numérico
        dcredit = -1
      # Máximo de 3 caractéres consecultivos idênticos
        maxrepeat = 3
      # Mínimo de 7 caracteres que não fazem parte da senha antiga
        difok = 7
      # Aplica as restrições de senha até quando o root está configurando a senha
        enforce_for_root

# Para Ativar as novas regras aos usuários já criados é necessário mudar a senha
  passwd <usuário>

root: TweetGummy1
mcerquei: TweetGummy2
```

Hostname, Users and Groups

Users group

> Um grupo é uma coleção de usuários. O principal objetivo dos grupos é definir o privilégio de exercer uma coleção de permissões, como ler, escrever ou executar um determinado arquivo para vários usuários.

▼ fontes

- https://linuxize.com/post/how-to-list-groups-in-linux/

Usuário root

> Usuário que tem acesso irrestrito aos arquivos e processos do sistema. Arquivos e processos ligados ao funcionamento do sistema, tem como proprietário natural o usuário root. Isso significa que só ele (e outros usuários que sejam definidos como "super usuários") pode alterá-los. O root também pode atuar sobre qualquer arquivo ou processo de outros usuários.

▼ fontes

- https://www.infowester.com/linroot.php

```
# Infos do hostname
hostnamectl status

# Muda o hostname
  # da pra mudar editando /etc/hostname
hostnamectl set-hostname <novo-hostname>

# Mostra todos os usuários do pc
  less /etc/passwd | cut -d ":" -f 1

# Mostra todos os usuários logados
  user

# Cria um novo usuário com home, nome completo -> + completo
  adduser <usuário>

# Cria um novo usuário -< parâmetro -m cria um home
  useradd -m <usuário>

# Deleta um usuário -> parâmetro -r deleta todos os arquivos vinculados ao usuário, inclusive home
  userdel -r <usuário>

# Altera o nome de usuário
  usermod -l <novo nome> <usuário>

# Alera o GID do grupo principal
  usermod -g <grupo (nome ou número)> <nome>

# Mostra o UID do usuário
  id -u <usuário>

# Mostra todos os grupos do pc
  less /etc/group | cut -d ":" -f 1

# Mostra os grupos do usuário
  groups <usuário>

# Exclui o usuário de um grupo
  sudo usermod -G <grupo pra ser mantido> <usuário>

# Cria um novo grupo
  groupadd <grupo>

# Deleta um grupo
  groupdel <grupo>

 # Adiciona usuário a um grupo
  gpasswd -a <usuário> <grupo>

# Remove usuário de um grupo
  gpasswd -d <usuário> <grupo>

# Mostra os usuários do grupo
  getent group <grupo>

# Mostra o grupo principal do usuário (GID)
  id -g <usuário>
```

Script

O que é Cron

> é um serviço que lança tarefas em tempos determinados. Está presente no Debian por padrão

Wall

> é um comando que permite que mensagens sejam escritas para todos os usuários em todos os terminais

```
# Instala CRON
  sudo apt-get install net-tools

# Criar arquivo monitoring.sh para adicionar o script
  sudo vim /usr/local/bin/monitoring.sh

# Torna o arquivo execultável
  sudo chmod 755 /usr/local/bin/monitoring.sh

# Testa o script
  sh /usr/local/bin/monitoring.sh

# COnfigura o cron
  sudo crontab -u root -e

#   m  h dom mon dow  command
  */10 * * * * sh /usr/local/bin/monitoring.sh | wall
```