

Guia de preparação LPIC-1 101

Tudo o que a prova cobra, organizado por domínio: conceitos-chave, comparações que confundem, pegadinhas reais, gatilhos do enunciado, questões comentadas e um plano para chegar pronto ao exame.

4
capítulos por domínio

28
questões comentadas

65%
nota de corte

6
provas completas no curso

ANTES DE COMEÇAR

Como usar este guia

Este guia foi feito para ser estudado junto com as 6 provas completas do curso LPIC-1 101: o guia explica o que cai e onde estão as armadilhas; as provas treinam o ritmo e mostram o que você ainda não domina.

Quando	O que fazer
Semana 1	Faça o teste diagnóstico (página seguinte) e leia os capítulos 1 e 2. No fim da semana, faça a Prova 1 do curso no tempo do exame.
Semana 2	Leia os demais capítulos, começando pelos domínios em que você errou mais no diagnóstico. Faça as provas 2 e 3.
Semana 3	Releia os alertas de pegadinha e as tabelas de comparação. Faça as provas 4 e 5 e revise cada erro pela explicação.
Semana 4	Estude a folha de cola, faça a Prova 6 e refaça as provas abaixo de 80%. Agende o exame quando estiver acima de 80% em todas (a nota de corte é 65%).

Mapa do exame

Cada prova do curso tem 60 questões e 90 minutos, no formato do exame. Referência de aprovação: 65%.

Domínio (capítulo)	Peso	Neste guia
1. 101 Arquitetura do Sistema	13,3%	17 conceitos · 14 pegadinhas
2. 102 Instalação do Linux e Gerenciamento de Pacotes	20,0%	18 conceitos · 14 pegadinhas
3. 103 Comandos GNU e Unix	43,3%	19 conceitos · 14 pegadinhas
4. 104 Dispositivos, Sistemas de Arquivos e FHS	23,3%	18 conceitos · 14 pegadinhas

Como ler os capítulos: os **tópicos-chave** são o que mais cai; a **tabela de memorização** separa o que a prova costuma confundir; os **alertas de prova** mostram as armadilhas das alternativas erradas; os **gatilhos** ligam palavras do enunciado à resposta; e o **checklist** diz se você está pronto para o domínio.

Estude um pouco todo dia, de graça

Instagram @strecht.simulados

Um post de prova por dia: questão comentada, pegadinhas e o "não confunda" de cada certificação.

Seguir no Instagram →

YouTube Strecht Simulados

Shorts diários com pegadinhas de prova e, aos sábados, um simulado comentado.

Inscrever-se no canal →

DIAGNÓSTICO

Você já sabe isso?

Responda às 12 questões sem consultar nada. As respostas, com a explicação e o capítulo para revisar, estão no fim do guia. Comece o estudo pelos capítulos em que errou.

D1 · 101 Arquitetura do Sistema

Um administrador esqueceu a senha de root e, no GRUB, adiciona um parâmetro para iniciar direto um shell em vez do init normal. Qual parâmetro é esse?

A. root=/bin/bash
B. quiet=/bin/bash
C. ro=/bin/bash
D. init=/bin/bash

D2 · 101 Arquitetura do Sistema

Um administrador isolou o rescue.target para manutenção. Qual comando volta ao alvo padrão do sistema?

A. systemctl reset
B. systemctl restart
C. systemctl default
D. systemctl reenable

D3 · 101 Arquitetura do Sistema

Em um sistema com systemd, qual comando define que o sistema passe a iniciar por padrão em modo texto multiusuário?

A. systemctl enable multi-user.target
B. systemctl isolate multi-user.target
C. systemctl set-default multi-user.target
D. systemctl get-default multi-user.target

D4 · 102 Instalação do Linux e Gerenciamento de Pacotes

Uma biblioteca aparece como libexemplo.so.2 no sistema. O que o número 2 costuma indicar?

A. O número de CPUs suportadas
B. A versão principal da biblioteca
C. A quantidade de programas que a usam
D. A prioridade de carregamento

D5 · 102 Instalação do Linux e Gerenciamento de Pacotes

Qual comando lista todos os arquivos instalados pelo pacote Debian openssh-server?

A. dpkg -s openssh-server
B. dpkg -P openssh-server
C. dpkg -L openssh-server
D. dpkg -c openssh-server

D6 · 102 Instalação do Linux e Gerenciamento de Pacotes

Qual comando informa se o pacote Debian curl está instalado e mostra sua versão?

A. dpkg -c curl
B. dpkg -s curl
C. dpkg -r curl
D. dpkg -i curl

D7 · 103 Comandos GNU e Unix

Um comando gera saída normal e mensagens de erro. Como gravar as duas no mesmo arquivo saida.log?

A. comando > saida.log 2>&1
B. comando 2> saida.log 1>&3
C. comando 2>&1 | saida.log
D. comando >> saida.log 2>/dev/null

D8 · 103 Comandos GNU e Unix

No vi, quais comandos copiam a linha atual e a colam abaixo do cursor?

A. cw e depois ZZ
B. x e depois P
C. yy e depois p
D. dd e depois u

D9 · 103 Comandos GNU e Unix

Qual comando mostra apenas as linhas que aparecem repetidas em lista.txt já ordenada?

A. uniq -d lista.txt
B. uniq -c lista.txt
C. uniq -i lista.txt
D. uniq -u lista.txt

D10 · 104 Dispositivos, Sistemas de Arquivos e FHS

Qual campo do /etc/fstab é recomendado para identificar a partição de forma estável, mesmo que a ordem dos discos mude?

A. Número do cilindro
B. Tamanho da partição
C. UUID=identificador
D. /dev/sdb1 direto

D11 · 104 Dispositivos, Sistemas de Arquivos e FHS

Por que não é possível criar um link físico de /home/ana/doc.txt em uma partição diferente?

A. Eles exigem permissão de root
B. Eles só funcionam para diretórios comuns
C. Eles apontam para inodes do mesmo sistema
D. Eles só existem em partições de swap

D12 · 104 Dispositivos, Sistemas de Arquivos e FHS

Na saída de ls -l, o que indica o primeiro caractere l, como em lrwxrwxrwx?

A. Um dispositivo de bloco
B. Um link simbólico
C. Um arquivo comum
D. Um diretório

01

CAPÍTULO 1 · 13,3% DA PROVA

101 Arquitetura do Sistema

Este é o menor tópico: 13,3%, cerca de 8 questões, mas é o mais decorável. Cobra a sequência de boot (firmware, carregador, kernel, init), SysV versus systemd, alvos e níveis de execução, journal, desligamento, módulos do kernel e detecção de hardware com udev, /proc e /sys. O que mais confunde é set-default versus isolate, rescue versus emergency, disable versus mask e modprobe versus insmod. Também caem caminhos exatos: /etc/inittab, /etc/init.d, /etc/modprobe.d, /etc/modules-load.d, /lib/modules/$(uname -r), /var/log/journal. Como são poucas questões, cada uma conta muito: não posso deixar pontos fáceis aqui.

Neste capítulo
17 tópicos-chave · 14 pegadinhas · 6 macetes · 14 gatilhos · 4 questões comentadas

Tópicos-chave

Sequência de inicialização

Firmware (BIOS/UEFI) faz o POST, localiza o dispositivo de boot e chama o carregador, que carrega o kernel e o initramfs; o kernel inicia o init (systemd ou SysV), que recebe o PID 1. A leitura do fstab e as unidades do systemd vêm depois, no espaço de usuário.

UEFI e Secure Boot

No UEFI, entradas de boot apontam para aplicativos .efi na partição de sistema EFI; ler o primeiro setor do disco é comportamento do BIOS legado. Secure Boot só executa carregadores e kernels assinados com chaves confiáveis. Legacy CSM emula o BIOS antigo.

initramfs

Sistema de arquivos temporário carregado com o kernel para fornecer módulos e preparar a montagem da raiz. Não substitui o carregador nem guarda logs ou senhas.

Parâmetros do kernel

/proc/cmdline mostra os parâmetros da inicialização atual (root=, quiet, systemd.unit=). init=/bin/bash substitui o programa inicial por um shell (recuperar senha de root); systemd.unit= escolhe um alvo.

Init SysV

O nível padrão fica na linha id:N:initdefault: do /etc/inittab. Scripts de serviços ficam em /etc/init.d e são ligados aos níveis por links em /etc/rcN.d: Snn inicia e Knn encerra o serviço ao entrar no nível. runlevel mostra o nível anterior e o atual (N 3 significa sem nível anterior) e telinit (ou init) muda o nível, como telinit 1 para monousuário.

Alvos do systemd

graphical.target equivale ao runlevel 5, multi-user.target ao 3 e rescue.target ao monousuário. emergency.target é mais restrito: shell mínimo com raiz só leitura, usado quando falhas (como erro no fstab) impedem o boot normal. systemctl list-units --type=target lista os alvos carregados.

Alvo padrão x alvo atual

systemctl get-default mostra e set-default altera o link default.target (persistente). systemctl isolate muda o estado atual sem persistir, iniciando o alvo pedido e parando o que não faz parte dele. systemctl default volta ao alvo padrão.

Habilitar, mascarar e consultar unidades

enable/disable tratam do início no boot; mask impede qualquer início, até manual, até unmask. is-enabled diz se inicia no boot e is-active se está rodando agora. Personalizações vão em /etc/systemd/system ou em drop-ins de systemctl edit; /usr/lib/systemd/system é sobrescrito por atualizações do pacote.

journald e journalctl

Com Storage=auto, o journald grava de forma persistente em /var/log/journal se o diretório existir; /run/log/journal fica em memória e se perde no reinício. journalctl -b seleciona inicializações (-b -1 anterior, -b 0 atual), -p err filtra prioridade, -k mostra mensagens do kernel, -u filtra unidade, -f acompanha e -n limita linhas.

Mensagens do kernel

dmesg e journalctl -k exibem as mensagens do kernel geradas no boot, como o hardware detectado.

Desligamento e avisos

shutdown -r reinicia, -h desliga, -c cancela e -k apenas avisa; o tempo pode ser +minutos ou hh:mm. halt para o sistema e pode deixar a máquina ligada; poweroff também desliga a energia (os dois existem em SysV e systemd). wall envia mensagem a todos os terminais; write a um usuário; mesg controla o recebimento.

acpid

Serviço que reage a eventos ACPI de energia, como botão de desligar e tampa do notebook. No systemd, o logind também trata alguns desses eventos.

Módulos do kernel

lsmod lista os carregados (Used by mostra quem depende de cada um); modinfo mostra descrição, autor e parâmetros; modprobe carrega com dependências pelo modules.dep e modprobe -r descarrega com dependentes não usados; insmod e rmmod não tratam dependências; depmod gera o mapa de dependências. Os módulos ficam em /lib/modules/$(uname -r).

Configuração de módulos

Arquivos .conf em /etc/modprobe.d aceitam blacklist (não carregar automaticamente) e options (parâmetros). A lista de módulos para carregar em todo boot fica em /etc/modules-load.d, lida pelo systemd-modules-load; o Debian também usa /etc/modules.

udev e hotplug

O udev recebe eventos do kernel e cria dinamicamente os arquivos em /dev conforme regras de /etc/udev/rules.d. udevadm monitor mostra os uevents em tempo real, udevadm info mostra atributos, trigger dispara eventos de novo e settle espera a fila esvaziar. Hotplug é o dispositivo conectado com o sistema ligado; coldplug é o que já estava presente no boot.

/proc e /sys

procfs e sysfs são sistemas de arquivos virtuais gerados pelo kernel em memória. /proc/cpuinfo mostra o processador, /proc/modules os módulos, /proc/cmdline os parâmetros de boot. O sysfs organiza dispositivos por classe, barramento e driver; /sys/class/net lista as interfaces de rede.

lspci e lsusb

lspci consulta o barramento PCI (placas de rede e vídeo); lsusb o barramento USB (confirmar pendrive reconhecido); lsblk mostra o disco criado para o pendrive e lsmod os módulos.

Macetes para memorizar

💡 Ordem do boot

F-C-K-I: Firmware, Carregador, Kernel, Init (PID 1).

💡 Runlevels x targets

1 = rescue, 3 = multi-user (texto), 5 = graphical (gráfico).

disable x mask

Disable tira do boot; mask coloca uma máscara que impede até você de iniciar.

Links SysV

S de Start, K de Kill.

modprobe x insmod

modprobe é o 'pro': resolve dependências; insmod é o básico que só insere.

Journal persistente

/run corre e some (memória); /var fica (disco).

Tabela de memorização: não confunda

Quando é...	...e quando é
systemctl set-default Mudar o alvo das próximas inicializações (persistente).	**systemctl isolate** Mudar para o alvo agora, sem reiniciar e sem persistir.
rescue.target Modo monousuário para manutenção.	**emergency.target** Shell mínimo com raiz só leitura quando falhas, como erro no fstab, impedem o boot.
systemctl disable Tirar do boot, mas ainda permitir iniciar manualmente.	**systemctl mask** Impedir qualquer início, até manual, até fazer unmask.
modprobe / modprobe -r Carregar ou remover módulo resolvendo dependências.	**insmod / rmmod** Carregar arquivo de módulo ou remover um módulo sem tratar dependências.
systemctl is-enabled Saber se o serviço inicia automaticamente no boot.	**systemctl is-active** Saber se o serviço está em execução agora.
/var/log/journal Journal persistente entre reinícios.	**/run/log/journal** Journal em memória, perdido no reinício.
halt Para o sistema e pode deixar a máquina ligada.	**poweroff** Para o sistema e desliga a energia.

Números que caem

Item	Valor
Peso do tópico 101 na prova	**13,3% (cerca de 8 questões)**
PID do processo init (systemd ou SysV)	**1**

Runlevel equivalente ao multi-user.target	**3**
Runlevel equivalente ao graphical.target	**5**
journalctl -b	**-1 inicialização anterior, 0 atual**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Usar systemctl enable multi-user.target ou isolate para definir o modo padrão.
Como não cair: Padrão persistente é set-default; enable não define o alvo padrão e isolate vale só para a sessão atual.

⚠ **Alerta de prova:** Usar set-default rescue.target para entrar já em recuperação.
Como não cair: set-default só afeta a próxima inicialização; imediato é systemctl isolate rescue.target.

⚠ **Alerta de prova:** Achar que disable impede até o início manual.
Como não cair: Isso é mask; disable só tira do boot.

⚠ **Alerta de prova:** Usar insmod para carregar módulo com dependências.
Como não cair: insmod não resolve dependências; modprobe usa o modules.dep.

⚠ **Alerta de prova:** Usar rmmod para descarregar também os dependentes não usados.
Como não cair: rmmod remove um módulo só; modprobe -r trata as dependências.

⚠ **Alerta de prova:** Passar o nome do módulo ao lsmod para ver parâmetros.
Como não cair: lsmod não recebe nome de módulo; parâmetros, autor e descrição vêm do modinfo.

⚠ **Alerta de prova:** Achar que /run/log/journal mantém os logs após reiniciar.
Como não cair: /run fica em memória; persistência exige /var/log/journal.

⚠ **Alerta de prova:** Ver o boot anterior com journalctl -b 0 ou -f.
Como não cair: -b 0 é o boot atual e -f acompanha; o anterior é journalctl -b -1.

⚠ **Alerta de prova:** Filtrar prioridade com journalctl -u err ou -k err.
Como não cair: -u filtra unidade e -k não recebe prioridade; o filtro é -p err.

⚠ **Alerta de prova:** Usar root=/bin/bash para cair num shell no lugar do init.
Como não cair: root= define a partição raiz; substituir o programa inicial é init=/bin/bash.

⚠ **Alerta de prova:** Achar que o UEFI lê o primeiro setor do disco em MBR.
Como não cair: Isso é o BIOS legado; o UEFI executa um aplicativo .efi da partição EFI.

⚠ **Alerta de prova:** Colocar a blacklist de módulo em /etc/fstab ou /etc/passwd, ou a lista de módulos do boot em /etc/profile.d.
Como não cair: blacklist e options vão em /etc/modprobe.d; módulos de todo boot em /etc/modules-load.d.

⚠ **Alerta de prova:** Personalizar a unidade em /usr/lib/systemd/system.
Como não cair: Atualizações sobrescrevem esse diretório; use /etc/systemd/system ou systemctl edit.

⚠ **Alerta de prova:** Usar runlevel 1 para mudar o nível ou shutdown -k para desligar.
Como não cair: runlevel só mostra; muda-se com telinit 1. shutdown -k só avisa; cancelar é -c.

Como cai na prova

📝 **Como cai:** Cenário de manutenção com systemd: entrar em recuperação, mudar padrão ou voltar.
Como resolver: Pergunto se é agora (isolate, default) ou para os próximos boots (set-default, get-default para consultar) e qual alvo corresponde ao modo pedido.

📝 **Como cai:** Ordenar ou identificar etapas do boot.
Como resolver: Sigo firmware (POST) → carregador → kernel + initramfs → init (PID 1) → unidades e fstab; descarto alternativas que colocam etapas de usuário antes do kernel.

📝 **Como cai:** Ferramenta de módulo com alternativas da mesma família.
Como resolver: Listar: lsmod; detalhes: modinfo; carregar ou remover com dependências: modprobe/modprobe -r; sem dependências: insmod/rmmod; mapa: depmod; configuração: /etc/modprobe.d e /etc/modules-load.d.

📝 **Como cai:** Consulta ao journal com filtro.
Como resolver: Monto a combinação: -b para boot (-1 anterior), -p para prioridade, -k para kernel, -u para unidade, -f para acompanhar.

📝 **Como cai:** Comando de desligamento, aviso ou cancelamento.
Como resolver: shutdown -r reinicia, -h desliga, -c cancela, -k só avisa; tempo +N ou hh:mm; wall para mensagem a todos.

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
nível anterior e atual no SysV	**runlevel**
nível padrão no SysV	**id:N:initdefault: em /etc/inittab**
links S e K em /etc/rc3.d	**S inicia, K encerra o serviço naquele nível**
esqueci a senha de root, parâmetro no GRUB	**init=/bin/bash**
erro no fstab, shell mínimo, raiz só leitura	**emergency.target**
voltar ao alvo padrão depois do rescue	**systemctl default**
mensagens da inicialização anterior	**journalctl -b -1**
journal perdido a cada reinício	**Criar /var/log/journal**
avisar todos os usuários conectados	**wall (shutdown também avisa)**
reiniciar daqui a 10 minutos avisando	**shutdown -r +10 'mensagem'**
módulo não deve carregar automaticamente	**blacklist em /etc/modprobe.d/**
ver eventos do kernel ao conectar dispositivo	**udevadm monitor**
parâmetros passados ao kernel no boot atual	**/proc/cmdline**
só executar carregadores assinados	**Secure Boot**

🎯 Se você lembrar de só uma coisa deste capítulo...

- set-default persiste; isolate muda agora; systemctl default volta ao padrão.
- emergency.target é mais restrito que rescue.target e aparece quando o fstab quebra o boot.
- modprobe resolve dependências; insmod e rmmod não.
- Journal persistente só com /var/log/journal; /run/log/journal some no reinício.
- Módulos ficam em /lib/modules/$(uname -r); blacklist em /etc/modprobe.d; carga no boot em /etc/modules-load.d.

Essenciais para a prova (checklist)

- ☐ Sei ordenar firmware, carregador, kernel, initramfs e init e explicar o POST.
- ☐ Sei o papel do UEFI, da partição EFI, do Secure Boot e do initramfs.
- ☐ Sei usar /proc/cmdline e o parâmetro init=/bin/bash.

- ☐ Sei onde o SysV define o nível padrão e o que fazem runlevel, telinit, /etc/init.d e os links S e K.
- ☐ Sei a equivalência entre runlevels e targets e a diferença entre rescue e emergency.
- ☐ Sei usar set-default, get-default, isolate, default, enable, disable, mask, is-enabled e is-active.
- ☐ Sei filtrar o journal por boot, prioridade, kernel e unidade e torná-lo persistente.
- ☐ Sei agendar, cancelar e avisar desligamentos e diferenciar halt de poweroff.
- ☐ Sei usar lsmod, modinfo, modprobe, modprobe -r, insmod, rmmod e depmod e onde configurar módulos.
- ☐ Sei o papel do udev, do udevadm, de /proc, /sys, lspci e lsusb.

Questões de revisão do capítulo

R1.1 · 101 Arquitetura do Sistema

Em um sistema com init SysV, qual comando muda o nível de execução atual para o modo monousuário?

A. telinit 5
B. telinit 1
C. uptime 1
D. runlevel 1

R1.2 · 101 Arquitetura do Sistema

Em uma máquina com UEFI, o que o firmware carrega para iniciar o sistema?

A. O arquivo /etc/fstab da raiz
B. O primeiro setor do disco em MBR
C. O kernel direto de um pendrive
D. Um aplicativo .efi da partição EFI

R1.3 · 101 Arquitetura do Sistema

No init SysV, o que indicam os prefixos S e K nos links de /etc/rc3.d?

A. S é seguro e K é crítico para o sistema
B. S inicia e K encerra o serviço naquele nível
C. S é sistema e K é chave de criptografia
D. S é script e K é kernel do sistema

R1.4 · 101 Arquitetura do Sistema

Na saída do lsmod, o que indica a coluna Used by de um módulo?

A. Os módulos que dependem dele
B. Quanta memória RAM está livre
C. Quantos usuários estão conectados
D. Quantos discos estão montados

Respostas comentadas

R1.1. B) telinit 1

Por quê: telinit (ou init) altera o nível de execução; runlevel mostra o anterior e o atual.

✗ A) Incorreta. Solicita o nível gráfico.
✗ C) Incorreta. Mostra o tempo ligado.
✗ D) Incorreta. Apenas mostra os níveis.

> **Dica de prova:** Mudar runlevel no SysV: telinit.

R1.2. D) Um aplicativo .efi da partição EFI

Por quê: No UEFI, entradas de boot apontam para arquivos .efi na partição de sistema EFI.

✗ A) Incorreta. O fstab é lido depois pelo sistema.
✗ B) Incorreta. Esse é o comportamento do BIOS legado.
✗ C) Incorreta. Não é o fluxo padrão.

> **Dica de prova:** UEFI: executa .efi da ESP.

R1.3. B) S inicia e K encerra o serviço naquele nível

Por quê: Links Snn iniciam e Knn param serviços ao entrar no nível.

✗ A) Incorreta. Não é o significado.
✗ C) Incorreta. Não é o significado.
✗ D) Incorreta. Não é o significado.

> **Dica de prova:** S inicia, K encerra.

R1.4. A) Os módulos que dependem dele

Por quê: Módulos em uso não podem ser removidos até que os dependentes sejam descarregados.

✗ B) Incorreta. Não é essa informação.
✗ C) Incorreta. Não é essa informação.
✗ D) Incorreta. Não é essa informação.

> **Dica de prova:** Used by: módulos que dependem dele.

02

CAPÍTULO 2 · 20,0% DA PROVA

102 Instalação do Linux e Gerenciamento de Pacotes

Este tópico vale 20% da prova (cerca de 12 questões) e mistura cinco frentes: layout de disco, GRUB, bibliotecas compartilhadas, pacotes Debian/RPM e Linux como convidado virtualizado. A prova quase sempre mostra quatro comandos da mesma família (dpkg -L, -s, -S, -c; rpm -qi, -ql, -qc, -qf) e só um faz o que o enunciado pede, então eu preciso saber cada opção de cor. No GRUB 2, o ponto que mais derruba é lembrar que o grub.cfg é gerado: eu edito /etc/default/grub e rodo grub-mkconfig ou update-grub. Em pacotes, meus erros clássicos são confundir update com upgrade, remove com purge e o arquivo .deb com o pacote instalado. Na parte de virtualização caem conceitos curtos: contêiner usa o kernel do host, clones precisam de machine-id e chaves SSH novos, cloud-init configura a primeira inicialização.

Neste capítulo

18 tópicos-chave · 14 pegadinhas · 7 macetes · 14 gatilhos · 4 questões comentadas

Tópicos-chave

MBR x GPT

O MBR aceita 4 partições primárias, ou 3 primárias e 1 estendida com lógicas, e usa endereços de 32 bits, o que limita partições a cerca de 2 TiB. O GPT não tem esse limite e o padrão comum é de 128 partições. Disco grande (4 TB, 6 TB) usado inteiro pede GPT.

Partições separadas (/boot, /var, /home)

/boot separada e não criptografada existe porque o carregador precisa ler o kernel quando a raiz está em LVM ou criptografada. /var separada impede que logs, filas e caches encham a raiz. /home separada preserva os dados dos usuários numa reinstalação e limita o impacto de usuários enchendo o disco.

Partição de sistema EFI (ESP)

Em instalações UEFI é obrigatória a ESP, formatada em FAT, que guarda os carregadores e costuma ser montada em /boot/efi. As entradas de boot do firmware são listadas e reordenadas com efibootmgr.

LVM

Volumes físicos (PV) são agrupados em grupos de volumes (VG), dos quais se criam volumes lógicos (LV). A vantagem cobrada é redimensionar volumes com flexibilidade. O LV ainda precisa ser formatado, backup continua necessário e partições ou discos inteiros servem de PV.

Swap em arquivo

Sem partição de swap, cria-se um arquivo, prepara-se com mkswap e ativa-se com swapon. Aumentar /boot ou montar /tmp com noexec não cria área de troca.

Configuração do GRUB 2

Eu edito /etc/default/grub (GRUB_CMDLINE_LINUX para parâmetros permanentes do kernel, GRUB_DEFAULT para a entrada padrão, GRUB_TIMEOUT para a espera) e regenero com grub-mkconfig -o /boot/grub/grub.cfg. Os scripts usados na geração ficam em /etc/grub.d (10_linux, 40_custom). O grub.cfg nunca é editado à mão.

Variações de distribuição do GRUB 2

No Debian/Ubuntu, update-grub é o atalho que chama grub-mkconfig -o /boot/grub/grub.cfg (e não recebe -o). Na família Red Hat os comandos e diretórios usam o prefixo grub2-: grub2-mkconfig -o /boot/grub2/grub.cfg.

grub-install

grub-install /dev/sda instala o carregador no registro de inicialização do disco (sistema com BIOS). Gerar configuração é outra tarefa: grub-mkconfig ou update-grub. grub-probe só consulta dispositivos e grub-reboot define a entrada da próxima inicialização.

GRUB Legacy

O menu do GRUB Legacy fica em /boot/grub/menu.lst (ou grub.conf). grub.cfg, /etc/default/grub e /etc/grub.d pertencem ao GRUB 2.

Interação com o GRUB no boot

A tecla e na entrada do menu altera parâmetros do kernel só para aquela inicialização. No prompt grub ou grub rescue, ls lista dispositivos e a sequência set root, linux, initrd e boot carrega o sistema manualmente.

Bibliotecas compartilhadas

ldd programa mostra as bibliotecas de que um executável depende; 'not found' indica biblioteca não localizada. Depois de acrescentar um diretório em /etc/ld.so.conf ou /etc/ld.so.conf.d, ldconfig atualiza o /etc/ld.so.cache; ldconfig -p lista o cache. No nome libx.so.N, o N é a versão principal (soname).

LD_LIBRARY_PATH e locais de bibliotecas

LD_LIBRARY_PATH acrescenta caminhos de busca de bibliotecas para a sessão, sem mudar o sistema. Bibliotecas ficam em /lib, /usr/lib e, em algumas distribuições de 64 bits (família Red Hat), /lib64 e /usr/lib64.

dpkg

-i instala um .deb, -r remove mantendo configuração, -P (purge) remove também a configuração, -L lista arquivos de um pacote instalado, -s mostra status e versão, -S descobre o pacote dono de um arquivo, -c lista o conteúdo de um arquivo .deb e -l lista os instalados. dpkg-reconfigure refaz as perguntas de configuração (ex.: tzdata). O banco do dpkg fica em /var/lib/dpkg.

APT

update baixa índices; upgrade atualiza sem instalar ou remover dependências novas; dist-upgrade (ou apt full-upgrade) trata mudanças de dependências. purge apaga configurações, remove as mantém, autoremove tira dependências órfãs, autoclean limpa pacotes obsoletos do cache e -f install corrige dependências quebradas. Versão específica: pacote=versão; pacote:arquitetura escolhe arquitetura.

apt-cache, apt-mark e sources.list

apt-cache search procura por nome ou descrição, show (ou apt show) mostra metadados, depends lista dependências e rdepends quem depende dele, policy mostra versões e prioridades. apt-mark hold trava a atualização (unhold libera). Linha do /etc/apt/sources.list: deb URL distribuição componentes.

Consultas e operações do rpm

-qa lista instalados (--last ordena pela data de instalação), -qi informações, -ql arquivos, -qc configurações, -qd documentação, -qR dependências, -qf dono de um arquivo e -qp consulta um arquivo de pacote. -i instala, -U atualiza ou instala, -F só atualiza o que já existe, -e remove, -V verifica alterações, --import confia em chave GPG. O banco fica em /var/lib/rpm.

dnf/yum e zypper

dnf install ./app.rpm instala RPM local resolvendo dependências; provides (yum whatprovides) acha quem fornece um arquivo; repolist mostra repositórios habilitados (--all inclui desativados); check-update só consulta; group install instala grupos; history undo desfaz transação. Repositórios ficam em arquivos .repo de /etc/yum.repos.d. No openSUSE, zypper install instala e zypper refresh (ref) atualiza metadados.

Linux como convidado virtualizado

Contêineres isolam processos com namespaces e cgroups e compartilham o kernel do host; cada VM tem kernel próprio. Clones precisam regenerar /etc/machine-id (usado pelo systemd e D-Bus) e as chaves SSH do host; cloud-init configura usuários, chaves e pacotes na primeira inicialização. Drivers virtio melhoram disco e rede no KVM, o agente do convidado (qemu-guest-agent, guest additions) integra com o hipervisor, e o disco persistente em IaaS é armazenamento em bloco.

Macetes para memorizar

💡 Opções do dpkg

L = Lista do pacote, S = Search (quem é o dono), s = status, c = conteúdo do .deb, P = Purge, r = remove.

💡 Consultas rpm -q

i = info, l = lista, c = config, d = docs, R = Requires, f = file (dono), a = all, p = package (arquivo).

💡 rpm -i, -U, -F

Install instala, Upgrade faz upgrade ou instala, Freshen só refresca o que já existe.

💡 Fluxo do GRUB 2

Edito o default, gero o cfg: /etc/default/grub → grub-mkconfig → grub.cfg. Install é só para o disco.

💡 Hierarquia do LVM

PV vira VG, VG vira LV, LV vira sistema de arquivos (ainda precisa formatar).

💡 Sintaxe de versão no APT

Igual é versão, dois-pontos é arquitetura: nginx=1.24.0-1, nginx:amd64.

💡 Boot manual no shell do GRUB

Raiz, Linux, initrd, boot: set root → linux → initrd → boot.

Tabela de memorização: não confunda

Quando é...	...e quando é
apt-get update Baixar os índices dos repositórios antes de instalar ou atualizar.	**apt-get upgrade / dist-upgrade** upgrade instala versões novas dos pacotes sem mexer em dependências; dist-upgrade pode instalar ou remover dependências para completar a atualização.
remove (apt-get remove / dpkg -r) Tirar o pacote mantendo os arquivos de configuração.	**purge (apt-get purge / dpkg -P)** Tirar o pacote e apagar também as configurações.
dpkg -L / rpm -ql Recebe o NOME do pacote e lista os arquivos que ele instalou.	**dpkg -S / rpm -qf** Recebe o CAMINHO de um arquivo e diz qual pacote o instalou.
dpkg -c / rpm -qp Consultar um ARQUIVO de pacote (.deb/.rpm) ainda não instalado.	**dpkg -L / rpm -ql** Consultar um pacote JÁ instalado no banco.
rpm -i / rpm -U -i instala; -U atualiza ou instala se não existir. Nenhum baixa dependências.	**rpm -F / dnf install ./app.rpm** -F só atualiza o que já está instalado; dnf install ./arquivo.rpm resolve dependências pelos repositórios.

grub-install Gravar o carregador no disco (ex.: /dev/sda).	**grub-mkconfig / update-grub** Regenerar o grub.cfg depois de editar /etc/default/grub ou /etc/grub.d.
ldd Ver de quais bibliotecas um programa depende e quais estão 'not found'.	**ldconfig** Atualizar o cache /etc/ld.so.cache após mudar /etc/ld.so.conf(.d); com -p, listar o cache.

Números que caem

Item	Valor
Peso do tópico 102 na prova	**20% (cerca de 12 questões)**
Partições primárias no MBR	**4 (ou 3 primárias + 1 estendida com lógicas)**
Limite de tamanho de partição no MBR	**cerca de 2 TiB (endereços de 32 bits)**
Partições no GPT (padrão comum)	**128**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Instalar versão específica com apt-get install nginx:1.24.0-1 ou nginx-1.24.0-1.
Como não cair: No APT a versão vem com sinal de igual (nginx=1.24.0-1); dois-pontos indicam arquitetura e o hífen formaria outro nome de pacote.

⚠ **Alerta de prova:** Colocar GRUB_CMDLINE_LINUX direto no /boot/grub/grub.cfg para tornar o parâmetro permanente.
Como não cair: O grub.cfg é gerado automaticamente; o lugar é /etc/default/grub, seguido de grub-mkconfig ou update-grub.

⚠ **Alerta de prova:** Usar update-grub -o /boot/grub2/grub.cfg na família Red Hat.
Como não cair: update-grub é atalho do Debian e não usa -o; no Red Hat é grub2-mkconfig -o /boot/grub2/grub.cfg.

⚠ **Alerta de prova:** Achar que dpkg -c curl mostra os arquivos do pacote instalado.
Como não cair: -c lê um arquivo .deb; para pacote instalado é dpkg -L.

⚠ **Alerta de prova:** Usar dpkg -L /bin/ls para descobrir quem instalou o arquivo.
Como não cair: -L espera nome de pacote; o dono de um arquivo é dpkg -S (no RPM, rpm -qf).

⚠ **Alerta de prova:** Instalar RPM local com rpm -i esperando que as dependências sejam baixadas.
Como não cair: rpm não baixa dependências; quem resolve é dnf (ou yum) install ./app.rpm.

⚠ **Alerta de prova:** Marcar 128 como limite de primárias do MBR ou dizer que o GPT não passa de 4 partições.
Como não cair: MBR: 4 primárias. 128 é o padrão comum do GPT.

⚠ **Alerta de prova:** Justificar o GPT dizendo que o MBR só funciona com UEFI.
Como não cair: O MBR funciona com BIOS; o motivo para discos grandes é o limite de cerca de 2 TiB por partição.

⚠ **Alerta de prova:** Pensar que rpm -i --nodeps desativa a verificação de assinaturas.
Como não cair: --nodeps ignora dependências e pode deixar o sistema inconsistente; desligar assinatura é --nosignature.

⚠ **Alerta de prova:** Usar ldconfig /usr/bin/ssh para ver as bibliotecas do programa, ou ldconfig -v para listar o cache.
Como não cair: Bibliotecas de um programa: ldd. Listar o cache: ldconfig -p (o -v recria o cache mostrando detalhes).

⚠ **Alerta de prova:** Escolher PKG_CONFIG_PATH para testar bibliotecas de um diretório próprio.
Como não cair: PKG_CONFIG_PATH é usado na compilação; em tempo de execução é LD_LIBRARY_PATH.

⚠ **Alerta de prova:** Trocar autoremove por autoclean ou clean.
Como não cair: autoremove remove dependências que ninguém usa; autoclean limpa pacotes obsoletos do cache; clean limpa o cache de downloads.

⚠ **Alerta de prova:** Extrair um RPM com tar -xvf pacote.rpm ou gzip -d.
Como não cair: tar não lê RPM e RPM não é gzip simples; o caminho é rpm2cpio pacote.rpm | cpio -idmv.

⚠ **Alerta de prova:** Dizer que contêineres exigem hipervisor tipo 1 ou que VMs compartilham o kernel do host.
Como não cair: É o contrário: contêiner compartilha o kernel do host via namespaces e cgroups; cada VM tem kernel próprio sobre hardware virtualizado.

Como cai na prova

📝 **Como cai:** Quatro comandos da mesma ferramenta com opções diferentes (dpkg -L/-s/-P/-c; rpm -qi/-qR/-ql/-qc).
Como resolver: Identifico o que o enunciado entrega (nome de pacote, caminho de arquivo ou arquivo .deb/.rpm) e o que pede (arquivos, dono, status, dependências); depois casa com a letra da opção.

📝 **Como cai:** Cenário de manutenção do GRUB: alterar parâmetro, entrada padrão ou regenerar configuração.
Como resolver: Pergunto se é temporário (tecla e) ou permanente (/etc/default/grub). Se permanente, a resposta inclui regenerar com grub-mkconfig ou update-grub; se for gravar no disco, grub-install.

📝 **Como cai:** Planejamento de disco: por que separar /boot, /var, /home ou usar GPT/LVM.
Como resolver: Associo cada partição ao seu motivo: /boot legível pelo carregador, /var protege a raiz, /home preserva dados; GPT por tamanho acima de 2 TiB; LVM por redimensionar.

📝 **Como cai:** Família de distribuição implícita: caminho de repositório ou ferramenta de outra distro nas alternativas.
Como resolver: Separo as famílias: Debian (dpkg, apt, /etc/apt/sources.list), Red Hat (rpm, dnf/yum, /etc/yum.repos.d), SUSE (zypper), Gentoo (emerge) e descarto as de família errada.

📝 **Como cai:** Erro de biblioteca ao executar programa.
Como resolver: Diagnóstico com ldd; correção permanente com /etc/ld.so.conf.d + ldconfig; teste temporário com LD_LIBRARY_PATH.

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
parâmetro do kernel só nesta inicialização	**Tecla e na entrada do menu do GRUB**
parâmetro permanente / todas as inicializações	**GRUB_CMDLINE_LINUX em /etc/default/grub + grub-mkconfig ou update-grub**
caiu no prompt grub rescue	**set root, linux, initrd e boot**
ordem das entradas de boot no firmware UEFI	**efibootmgr**
libfoo.so.1 => not found	**Biblioteca ausente: instalar ou ajustar /etc/ld.so.conf e rodar ldconfig**
testar bibliotecas sem mudar a configuração do sistema	**LD_LIBRARY_PATH**
faltaram dependências depois do dpkg -i	**apt-get -f install**

impedir que o pacote seja atualizado	**apt-mark hold**
refazer as perguntas de configuração de um pacote Debian	**dpkg-reconfigure**
qual pacote fornece o arquivo, mesmo sem estar instalado	**dnf provides (yum whatprovides)**
desfazer uma atualização feita com dnf	**dnf history undo N**
confiar na chave do repositório RPM	**rpm --import chave.asc**
VM clonada de modelo com IDs repetidos	**Regenerar /etc/machine-id e as chaves SSH do host (cloud-init ajuda)**
configurar usuários e chaves na primeira inicialização da instância em nuvem	**cloud-init**

🎯 Se você lembrar de só uma coisa deste capítulo...

- grub.cfg é gerado: edito /etc/default/grub e rodo grub-mkconfig (update-grub no Debian, grub2-mkconfig no Red Hat).
- dpkg -L/rpm -ql recebem pacote; dpkg -S/rpm -qf recebem arquivo; dpkg -c/rpm -qp leem arquivo de pacote.
- rpm não baixa dependências; dnf install ./arquivo.rpm e apt-get -f install resolvem.
- MBR: 4 primárias e cerca de 2 TiB por partição; acima disso, GPT.
- Contêiner compartilha o kernel do host; clone de VM precisa de machine-id e chaves SSH novos.

Essenciais para a prova (checklist)

- ☐ Sei explicar por que /boot, /var e /home ficam em partições separadas.
- ☐ Sei os limites do MBR e quando usar GPT e a partição EFI em FAT.
- ☐ Sei a diferença entre grub-install, grub-mkconfig e update-grub e onde ficam /etc/default/grub e /etc/grub.d.
- ☐ Sei carregar o kernel manualmente no shell do GRUB e alterar parâmetros temporários com a tecla e.
- ☐ Sei usar ldd, ldconfig, ldconfig -p e LD_LIBRARY_PATH.
- ☐ Sei as opções -i, -r, -P, -L, -s, -S, -c e -l do dpkg.
- ☐ Sei diferenciar update, upgrade, dist-upgrade, remove, purge, autoremove e autoclean no APT.
- ☐ Sei as consultas rpm -qa, -qi, -ql, -qc, -qd, -qR, -qf e as ações -i, -U, -F, -e, -V.
- ☐ Sei usar dnf provides, repolist, check-update, group install e history undo, e onde ficam os .repo.
- ☐ Sei o que regenerar em clones de VM e para que servem cloud-init, virtio e o agente do convidado.

Questões de revisão do capítulo

R2.1 · 102 Instalação do Linux e Gerenciamento de Pacotes

Qual comando mostra versão, fornecedor, licença e descrição de um pacote RPM instalado?

A. rpm -qR httpd
B. rpm -ql httpd
C. rpm -qi httpd
D. rpm -qc httpd

R2.2 · 102 Instalação do Linux e Gerenciamento de Pacotes

Qual comando lista os repositórios habilitados em um sistema com dnf?

A. dnf repolist
B. dnf list all
C. dnf makecache
D. dnf clean all

R2.3 · 102 Instalação do Linux e Gerenciamento de Pacotes

Em uma nuvem IaaS, qual recurso fornece um disco persistente que pode ser anexado a uma instância?

A. Grupo de segurança
B. Imagem de instância
C. Endereço IP elástico
D. Armazenamento em bloco

R2.4 · 102 Instalação do Linux e Gerenciamento de Pacotes

A raiz do sistema ficará em um volume criptografado. Por que é comum manter /boot em uma partição separada e não criptografada?

A. O /boot precisa de sistema de arquivos XFS
B. O /boot guarda os dados dos usuários
C. O /boot substitui a área de troca
D. O carregador precisa ler o kernel

Respostas comentadas

R2.1. C) rpm -qi httpd

Por quê: rpm -qi (query info) exibe os metadados do pacote.

✗ A) Incorreta. Lista dependências.

✗ B) Incorreta. Lista arquivos do pacote.

✗ D) Incorreta. Lista arquivos de configuração.

> **Dica de prova:** Informações do RPM: rpm -qi.

R2.2. A) dnf repolist

Por quê: dnf repolist (com --all mostra também os desativados).

✗ B) Incorreta. Lista pacotes.

✗ C) Incorreta. Baixa metadados.

✗ D) Incorreta. Limpa caches.

> **Dica de prova:** Repositórios ativos: dnf repolist.

R2.3. D) Armazenamento em bloco

Por quê: Instâncias IaaS combinam computação, armazenamento em bloco e rede.

✗ A) Incorreta. Controla o tráfego de rede.

✗ B) Incorreta. Serve de base para criar máquinas.

✗ C) Incorreta. Fornece IP público.

> **Dica de prova:** Disco persistente na nuvem: armazenamento em bloco.

R2.4. D) O carregador precisa ler o kernel

Por quê: Separar /boot facilita a inicialização quando a raiz está em LVM ou criptografada.

✗ A) Incorreta. Não há essa exigência.

✗ B) Incorreta. Não guarda dados de usuários.

✗ C) Incorreta. Não substitui a swap.

> **Dica de prova:** /boot separada: acessível ao carregador.

03

CAPÍTULO 3 · 43,3% DA PROVA

103 Comandos GNU e Unix

É o maior tópico da prova: 43,3%, cerca de 26 questões. Cobra linha de comando pura: redirecionamentos, pipes, filtros de texto (cut, sort, uniq, tr, sed, grep), find e xargs, compactação e arquivamento, processos, sinais, prioridade e o editor vi. Quase tudo é sintaxe exata, então a alternativa errada costuma ser o comando certo com a opção vizinha (-c em vez de -n, -r em vez de -R, + em vez de -). Meus pontos fracos são a ordem dos redirecionamentos (2>&1 antes ou depois do >), uniq sem sort, sinais por número e as teclas do vi. Vale treinar lendo cada alternativa como se fosse rodar no terminal.

Neste capítulo

19 tópicos-chave · 14 pegadinhas · 8 macetes · 14 gatilhos · 4 questões comentadas

Tópicos-chave

Redirecionamentos

> sobrescreve, >> acrescenta, 2> redireciona o stderr e 2>/dev/null descarta só os erros. cmd > arq 2>&1 manda saída e erros para o arquivo; cmd 2>&1 > arq manda erros para a tela, porque a avaliação é da esquerda para a direita. &> descarta ou grava os dois. Com set -o noclobber, > não sobrescreve e >| força.

Pipes, tee e encadeamento

| liga a saída de um comando à entrada do outro. tee mostra na tela e grava em arquivo; tee -a acrescenta. && executa o segundo só se o primeiro der certo, || só se falhar e ; sempre. $? guarda o código de saída (0 é sucesso).

Variáveis e quoting

export coloca a variável no ambiente herdado pelos filhos; export PATH=$PATH:/opt/app/bin acrescenta ao PATH (separador dois-pontos). unset remove; env e printenv mostram só o ambiente; set mostra também variáveis locais e funções. Aspas simples preservam tudo literalmente ('$HOME'); aspas duplas expandem variáveis. EDITOR e VISUAL definem o editor de crontab e visudo.

Parâmetros especiais e histórico

$? código de saída, $$ PID do shell, $! PID do último job em segundo plano, $# número de argumentos. !! repete o último comando, !n o comando n do histórico e !$ o último argumento. cd - volta ao diretório anterior (OLDPWD).

type, alias e execução

type informa se o nome é builtin, alias, função ou arquivo; which só procura executáveis no PATH. alias ll='ls -lh --color=auto' cria apelido na sessão (persistir no ~/.bashrc). Script no diretório atual fora do PATH roda com ./script.sh.

Globbing

* casa qualquer sequência, ? exatamente um caractere e [abc] um caractere do conjunto. O shell expande o padrão antes de executar o comando. a?.txt casa a1.txt e a2.txt, mas não a10.txt.

Expressões regulares e grep

. é um caractere qualquer, * repete o anterior, ^ início e $ fim de linha; dentro de colchetes, ^ nega ([^0-9]). \. casa ponto literal. grep -E (egrep) usa ER estendida com | e {n}; grep -F (fgrep) busca texto fixo. Opções: -i ignora caixa, -v inverte, -n numera, -c conta, -l lista arquivos, -r recursivo, -w palavra inteira, -x linha inteira, -A/-B/-C contexto, -e vários padrões.

sed

s/a/b/ troca a primeira ocorrência da linha e s/a/b/g todas. -n com p imprime só o selecionado ('5,10p', '/INICIO/,/FIM/p'); d apaga ('/^$/d' tira linhas vazias). s/^/# / insere no início. -i grava no arquivo; sem -i o resultado vai para a saída padrão.

cut, paste, tr

cut -d define delimitador e -f campos (padrão TAB); -c recorta caracteres; N- vai do campo N ao fim. paste une linhas correspondentes lado a lado (join exige campo comum ordenado). tr 'a-z' 'A-Z' converte, tr -d apaga (ex.: '\r' de arquivos do Windows) e tr -s comprime repetições.

sort, uniq, wc

sort -r inverte, -n numérico, -h entende K/M/G, -u remove duplicados, -t separador, -k chave (sort -t: -k3 -n /etc/passwd). uniq só compara linhas adjacentes, por isso vem depois do sort: -c conta, -d só repetidas, -u só únicas. Ranking: sort | uniq -c | sort -nr.

Visualizar arquivos

head e tail usam -n para linhas e -c para bytes; tail -f acompanha o arquivo; tail -n +5 começa na linha 5; head -n 20 | tail -n 5 pega as linhas 16 a 20. nl e cat -n numeram, od -c mostra caracteres invisíveis, less e more paginam, split -l divide por linhas e -b por bytes, file identifica o tipo pelo conteúdo.

find e xargs

find filtra por -name/-iname, -type (f, d, l, b, c, p, s), -size +100M (+ maior, - menor), -mtime -7, -newer arquivo, -user, -empty e -maxdepth 1. -exec rm {} \; executa por arquivo e -delete é alternativa. xargs transforma a entrada em argumentos; xargs -I {} coloca o item no meio do comando.

Gerenciamento de arquivos

cp -r copia diretórios e cp -a preserva permissões, datas, links e dono. mv move ou renomeia (último argumento é o destino). mkdir -p cria a árvore e -m define o modo; rmdir só remove diretório vazio. touch cria arquivo vazio ou atualiza datas (-d define data). ls -a mostra ocultos e ls -ltr deixa os mais recentes por último.

tar, cpio e dd

tar: c cria, x extrai, t lista, r acrescenta; z gzip, j bzip2, J (--xz) xz; -C muda o diretório antes de extrair e nomes após as opções escolhem membros. find . | cpio -o > copia.cpio cria e cpio -i extrai. dd if= origem e of= destino (inverter destrói dados); bs tamanho do bloco e count quantidade.

Compactação e hash

gzip substitui o arquivo original pelo .gz (-k ou -c mantêm). Descompactar: gunzip, bunzip2 (= bzip2 -d), unxz (= xz -d). Ler sem descompactar em disco: zcat, bzcat, xzcat, cada um para seu formato. sha256sum calcula o hash e sha256sum -c confere uma lista de somas.

Processos e monitoramento

ps aux é um retrato e top atualiza continuamente; ps -u filtra por usuário, -C por comando, -p por PID. Estados: R executando, S dormindo, D espera de E/S, T parado, Z zumbi. pgrep e pidof mostram PIDs pelo nome. uptime mostra carga média de 1, 5 e 15 minutos; free -h mostra RAM e swap, e a coluna available estima a memória utilizável sem swap. watch -n 5 repete um comando.

Sinais, jobs e sessões

kill sem opção envia SIGTERM (15); -9 é SIGKILL, que não pode ser tratado; -1 SIGHUP (recarregar configuração); -19 SIGSTOP; SIGCONT retoma. kill usa PIDs, killall e pkill usam nomes; kill -l lista sinais. jobs, fg %n, bg %n e kill %n trabalham com jobs. nohup ignora SIGHUP e grava em nohup.out; screen (-r reanexa) e tmux (new -s, attach -t) mantêm sessões.

nice e renice

nice vai de -20 (maior prioridade) a 19 (menor); sem -n, nice soma 10. Só o root reduz o valor (aumenta a prioridade). nice vale ao iniciar; renice altera processos em execução com -p (PID), -g (grupo de processos) ou -u (usuário). A coluna NI do ps -l e do top mostra o nice.

Editor vi

i insere antes do cursor, a depois, o abre linha abaixo e O acima. dd apaga linha, 3dd três linhas, dw até a próxima palavra, yy copia, p cola abaixo, u desfaz. /texto busca para a frente, ?texto para trás, n repete. 0 vai ao início da linha, $ ao fim, G à última linha. :wq ou ZZ gravam e saem, :q! e ZQ saem sem gravar, :w nome grava em outro arquivo, :%s/a/b/g troca em todo o arquivo, :set number numera.

Macetes para memorizar

Ordem do 2>&1

Primeiro escolha o destino, depois junte os erros: > arq 2>&1.

Letras do tar

c cria, x extrai, t tabela (lista), r acrescenta; z gzip, j bzip2, J maiúsculo xz (--xz).

Sinais

1 HUP recarrega, 9 KILL mata, 15 TERM pede para sair (é o padrão).

nice

Quanto mais 'legal' (nice alto, até 19), menos CPU o processo pede; ficar menos legal (negativo) só o root.

Inserção no vi

i antes, a after (depois), o abre linha abaixo, O maiúsculo abre acima.

Sair do vi

:wq e ZZ gravam; :q! e ZQ fogem sem gravar.

Sinais + e - no find

+ é 'mais que', - é 'menos que': -mtime -7 é recente, +7 é antigo.

Seções do man

1 comandos, 5 formatos de arquivos, 8 administração.

Tabela de memorização: não confunda

Quando é...	...e quando é
cmd > arq 2>&1 Saída e erros vão para o arquivo.	**cmd 2>&1 > arq** Erros vão para a tela e só a saída vai para o arquivo (ordem esquerda para direita).
grep -E Usar ER estendida: alternativa com | e repetição {n}.	**grep -F** Procurar texto literal, sem tratar + ou | como operadores.

kill Enviar sinal a um PID (ou a um job com %n).	**killall / pkill** Enviar sinal a processos pelo nome (pkill também por usuário com -u).
nice Iniciar um comando já com prioridade ajustada.	**renice** Alterar a prioridade de processo em execução (-p, -g, -u).
SIGTERM (15) Padrão do kill; pede término e pode ser tratado.	**SIGKILL (9)** Encerramento imediato que o processo não pode tratar.
env / printenv Só variáveis exportadas do ambiente.	**set** Tudo do shell, inclusive variáveis locais e funções.
uniq Só remove repetições vizinhas.	**sort | uniq** Remove ou conta todas as repetições, mesmo fora de ordem.

Números que caem

Item	Valor
Peso do tópico 103 na prova	**43,3% (cerca de 26 questões)**
Faixa do nice	**-20 (maior prioridade) a 19 (menor)**
nice sem -n	**acrescenta 10**
Sinais mais cobrados	**1 SIGHUP, 2 SIGINT, 9 SIGKILL, 15 SIGTERM, 19 SIGSTOP**
Carga média do uptime	**1, 5 e 15 minutos**
Seções do man	**1 comandos, 5 formatos de arquivos, 8 administração**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Contar linhas distintas com uniq antes do sort.
Como não cair: uniq só compara linhas adjacentes; primeiro sort, depois uniq (e uniq -c para contar).

⚠ **Alerta de prova:** Achar que cmd 2>&1 > saida.txt manda tudo para o arquivo.
Como não cair: Redirecionamentos são avaliados da esquerda para a direita: o 2>&1 copiou a tela antes; o certo é > arq 2>&1.

⚠ **Alerta de prova:** Usar tee sem -a para acrescentar ao log.
Como não cair: tee sozinho sobrescreve; tee -a acrescenta.

⚠ **Alerta de prova:** Usar sed -e ou sed -n para gravar a substituição no arquivo.
Como não cair: Só sed -i grava; -e só mostra o resultado e -n suprime a saída.

⚠ **Alerta de prova:** Usar sed 's/http/https/' achando que troca todas as ocorrências.
Como não cair: Sem g troca só a primeira de cada linha; no vi, :s sem % vale só para a linha atual.

⚠ **Alerta de prova:** Usar find -size -100M ou -mtime +7 com o sentido invertido.
Como não cair: +N é mais que N e -N é menos que N: maior que 100 MB é +100M; últimos 7 dias é -mtime -7.

⚠ **Alerta de prova:** Esquecer {} no find -exec rm \;.
Como não cair: {} representa o arquivo e \; encerra: find . -name '*.bak' -exec rm {} \;.

⚠ **Alerta de prova:** Usar nice -n -20 ou renice 19 ./processa.sh para baixar a prioridade.
Como não cair: Menor prioridade é 19 com nice ao iniciar; -20 é prioridade máxima e exige root; renice recebe PIDs.

⚠ **Alerta de prova:** Usar kill 1 para encerrar o job [1].
Como não cair: Número sem % é PID (o PID 1 é o init); job é kill %1.

⚠ **Alerta de prova:** Usar kill firefox para encerrar por nome.
Como não cair: kill espera PIDs; por nome é killall ou pkill; pgrep só lista.

⚠ **Alerta de prova:** Criar tar.xz com --gzip ou --bzip2.
Como não cair: xz é -J (--xz); -z é gzip e -j bzip2; --lzma usa o formato lzma antigo.

⚠ **Alerta de prova:** Usar dd if=/dev/zero ... bs=100 count=1M para 100 MB, ou /dev/null como origem.
Como não cair: bs=1M count=100 gera 100 MB; /dev/null não fornece dados; inverter if e of grava no disco.

⚠ **Alerta de prova:** Usar sort -n em saída de du -h.
Como não cair: sort -n ignora os sufixos K, M e G; sort -h entende.

⚠ **Alerta de prova:** Usar export -n para apagar uma variável.
Como não cair: export -n só tira do ambiente exportado; remover é unset.

Como cai na prova

📝 **Como cai:** Mesmo comando com quatro opções diferentes (grep -n/-c/-l/-h; uniq -c/-d/-u; sort -r/-n/-h).
Como resolver: Traduzo a necessidade do enunciado (numerar, contar, listar arquivo, inverter) para a letra da opção e descarto as que fazem algo próximo mas diferente.

📝 **Como cai:** Pipeline com a ordem dos comandos trocada nas alternativas.
Como resolver: Simulo o fluxo da esquerda para a direita: o que cada etapa recebe e entrega. uniq sem sort, head/tail invertidos ou origem e destino trocados denunciam a errada.

📝 **Como cai:** Redirecionamento com descritores (>, >>, 2>, 2>&1, &>, >|).
Como resolver: Leio da esquerda para a direita, anotando para onde aponta o 1 e o 2 a cada passo, e checo se o arquivo é sobrescrito ou acrescentado.

📝 **Como cai:** Gerenciar processo: prioridade, sinal ou job.
Como resolver: Verifico se o processo já está rodando (renice/kill) ou vai iniciar (nice/nohup), se a referência é PID, nome ou %job, e qual sinal o cenário pede.

📝 **Como cai:** Tecla ou comando do vi.
Como resolver: Separo modo de comando (dd, yy, p, u, /, ZZ) de comandos ex com dois-pontos (:wq, :q!, :w nome, :%s, :set number) e confiro a direção (o/O, p/P, //?).

📝 **Como cai:** Múltipla resposta (escolha DUAS) com dois comandos que atendem ao mesmo pedido.
Como resolver: Procuro os pares que resolvem a mesma tarefa: kill e kill -15 (SIGTERM), bunzip2 e bzip2 -d (descompactar .bz2), fg e bg (retomar job suspenso), pgrep e pidof (PID pelo nome), more e less (paginar), zcat e xzcat (ler compactado sem descompactar, cada um no seu formato).

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
gravar saída e erros no mesmo arquivo	**comando > arq 2>&1**
descartar apenas as mensagens de erro	**2>/dev/null**
mostrar na tela e gravar ao mesmo tempo	**tee (tee -a para acrescentar)**
rodar o segundo comando só se o primeiro der certo	**&&**
continuar rodando depois do logout do SSH	**nohup comando & (ou screen/tmux)**

retomar sessão depois de cair a conexão	**screen -r ou tmux attach**
processo não responde ao término padrão	**SIGKILL: kill -9**
daemon deve recarregar a configuração	**SIGHUP: pkill -HUP nome**
estado Z no ps	**Processo zumbi**
arquivo vindo do Windows com \r	**tr -d '\r' (ou dos2unix)**
página de manual do formato de arquivo, não do comando	**man 5 (seção 5 é formato de arquivo)**
não lembro o nome do comando, só uma palavra	**man -k (apropos)**
ranking dos itens mais frequentes	**sort | uniq -c | sort -nr**
imprimir o texto sem expandir a variável	**Aspas simples: echo '$HOME'**

🎯 **Se você lembrar de só uma coisa deste capítulo...**

- Redirecionamentos são lidos da esquerda para a direita: > arq 2>&1.
- uniq só enxerga linhas vizinhas: sempre sort antes.
- kill sem opção é SIGTERM (15); SIGKILL (9) e SIGSTOP não podem ser tratados.
- Só o root dá nice negativo; renice é para processo em execução.
- sed e :s sem g trocam só a primeira ocorrência da linha; sed só grava com -i.

Essenciais para a prova (checklist)

- ☐ Sei prever o efeito de >, >>, 2>, 2>&1, &>, >| e da ordem entre eles.
- ☐ Sei usar export, unset, env, set, alias, type e as aspas simples e duplas.
- ☐ Sei montar regex básica e estendida e escolher entre grep, grep -E e grep -F.
- ☐ Sei as opções -i, -v, -n, -c, -l, -r, -w, -x, -A, -B e -e do grep.
- ☐ Sei substituir, imprimir intervalos e apagar linhas com sed.
- ☐ Sei combinar cut, sort, uniq, tr, paste, head e tail em pipelines.
- ☐ Sei filtrar com find por nome, tipo, tamanho, datas e dono e agir com -exec ou xargs.
- ☐ Sei criar, listar e extrair com tar e gzip, bzip2, xz, e usar cpio e dd.
- ☐ Sei gerenciar processos com ps, top, kill, pkill, jobs, fg, bg, nohup, nice e renice.
- ☐ Sei editar, buscar, copiar, colar, desfazer, substituir e sair no vi.

Questões de revisão do capítulo

R3.1 · 103 Comandos GNU e Unix

Qual comando move todos os arquivos terminados em .log do diretório atual para o diretório logs?

A. mv logs/ *.log
B. mv -r .log logs/
C. mv ?.log logs/
D. mv *.log logs/

R3.2 · 103 Comandos GNU e Unix

Qual comando mostra apenas os 5 primeiros caracteres de cada linha de codigos.txt?

A. head -5 codigos.txt
B. tail -5 codigos.txt
C. cut -c1-5 codigos.txt
D. cut -f1-5 codigos.txt

R3.3 · 103 Comandos GNU e Unix

Qual comando apaga, a partir do diretório atual, os arquivos com extensão .bak usando a própria ação do find?

A. find . -name '*.bak' -exec rm \;
B. find . -name '*.bak' -print rm {}
C. find . -name '*.bak' -exec rm {} \;
D. find . -name '*.bak' -ls rm {} \;

R3.4 · 103 Comandos GNU e Unix

Qual comando substitui texto diretamente no arquivo config.ini, gravando a alteração?

A. sed -r 's/8080/9090/' config.ini
B. sed -e 's/8080/9090/' config.ini
C. sed -i 's/8080/9090/' config.ini
D. sed -n 's/8080/9090/' config.ini

Respostas comentadas

R3.1. D) mv *.log logs/

Por quê: O shell expande o glob antes de executar o mv; o último argumento é o destino.

✗ A) Incorreta. Inverte origem e destino.

✗ B) Incorreta. Não usa glob e -r não existe.

✗ C) Incorreta. Casa só nomes de um caractere.

> **Dica de prova:** Mover vários: mv *.log destino/.

R3.2. C) cut -c1-5 codigos.txt

Por quê: cut -c trabalha com posições de caracteres; -f com campos.

✗ A) Incorreta. Mostra as 5 primeiras linhas.

✗ B) Incorreta. Mostra as 5 últimas linhas.

✗ D) Incorreta. Recorta campos.

> **Dica de prova:** Primeiros caracteres: cut -c.

R3.3. C) find . -name '*.bak' -exec rm {} \;

Por quê: No -exec, {} representa o arquivo e \; encerra o comando; -delete é alternativa.

✗ A) Incorreta. Falta o marcador do arquivo.

✗ B) Incorreta. -print não executa comandos.

✗ D) Incorreta. -ls apenas lista.

> **Dica de prova:** Executar para cada resultado: find -exec {} \;.

R3.4. C) sed -i 's/8080/9090/' config.ini

Por quê: sed -i altera o arquivo; sem -i o resultado vai para a saída padrão.

✗ A) Incorreta. Só ativa regex estendida.

✗ B) Incorreta. Mostra o resultado sem gravar.

✗ D) Incorreta. Suprime a saída e não grava.

> **Dica de prova:** Editar no lugar: sed -i.

04

CAPÍTULO 4 · 23,3% DA PROVA

104 Dispositivos, Sistemas de Arquivos e FHS

Este tópico pesa 23,3% (cerca de 14 questões) e cobra a vida inteira de um disco: particionar, formatar, verificar, montar, controlar permissões e localizar arquivos segundo o FHS. As questões costumam ser de comando exato: qual opção do tune2fs, qual campo do fstab, qual octal de permissão, qual opção do mount. Onde mais erro é em permissões especiais (4000/2000/1000, s minúsculo e maiúsculo, t), em umask aplicada a arquivo versus diretório e em links físicos versus simbólicos. O FHS cai em pares que se parecem: /tmp e /var/tmp, /media e /mnt, /usr/local e /usr/bin. Também preciso separar as ferramentas de ext (e2fsck, tune2fs) das de XFS (xfs_repair, xfs_db, xfs_fsr).

Neste capítulo

18 tópicos-chave · 14 pegadinhas · 7 macetes · 14 gatilhos · 4 questões comentadas

Tópicos-chave

fdisk, gdisk e parted

No fdisk, n cria, d apaga, p mostra, w grava e q sai sem gravar; fdisk -l lista tabelas de todos os discos. gdisk é próprio para GPT. No parted, mklabel gpt (ou msdos) cria a tabela e mkpart cria partições.

Códigos de tipo no MBR

83 é Linux, 82 é swap Linux, 8e é Linux LVM e 07 é NTFS/exFAT.

Criação de sistemas de arquivos e swap

mkfs.ext4 /dev/sdb1 (ou mkfs -t ext4) cria ext4; mkfs.vfat serve para pendrive compatível com Windows e macOS. Swap: mkswap prepara e swapon ativa; no fstab o tipo é swap; swapon --show, swapon -s e /proc/swaps mostram a swap ativa.

Tipos de sistemas de arquivos

Btrfs oferece nativamente cópia na escrita, subvolumes e snapshots. ext2 não tem journal nem snapshots. VFAT e exFAT são comuns em mídias removíveis. XFS e ext4 não são lidos nativamente no Windows.

tune2fs

tune2fs -l mostra o superbloco (contagem de montagens, intervalo de verificação); -c e -i ajustam verificações periódicas; -L define rótulo (como e2label); -m define a reserva de blocos do root, que por padrão é 5%; -j adiciona journal.

e2fsck e fsck

fsck deve rodar em sistema desmontado ou montado só leitura. No e2fsck, -f força a verificação mesmo limpo, -n responde não a tudo sem corrigir, -y responde sim a tudo e -p repara automaticamente o que é seguro. O lost+found não deve ser apagado.

Ferramentas XFS

xfs_repair repara, xfs_db inspeciona e depura estruturas, xfs_fsr reorganiza (desfragmenta) arquivos com o sistema montado. e2fsck é para ext2/3/4, não para XFS.

df, du e inodes

df -h mostra uso de cada sistema de arquivos montado; df -i mostra inodes, que podem acabar antes do espaço quando há muitos arquivos pequenos. du soma uso de diretórios; du -sh dá o total legível. ls -i só mostra o inode de cada arquivo.

/etc/fstab

Seis campos: dispositivo, ponto de montagem, tipo, opções, dump e ordem do fsck (0, 1 ou 2). UUID= ou LABEL= identificam de forma estável; /dev/sdb1 pode mudar entre inicializações. blkid e lsblk -f mostram UUID, tipo e rótulo.

Opções de montagem

defaults = rw, suid, dev, exec, auto, nouser, async. noauto impede montagem no boot e no mount -a; user deixa um usuário comum montar e desmontar, users deixa qualquer usuário desmontar. noexec bloqueia binários, nosuid ignora SUID/SGID, nodev ignora arquivos de dispositivo e noatime não atualiza datas de acesso.

mount e montagens especiais

O ponto de montagem precisa ser um diretório existente. mount -o loop monta arquivo .iso; mount --bind expõe um diretório em outro lugar; --move move uma montagem existente; mount -o remount,ro / altera opções sem desmontar. findmnt mostra a árvore de montagens. No systemd, montagens também podem ser unidades .mount.

Desmontagem com alvo ocupado

lsof /ponto e fuser -m mostram os processos que mantêm arquivos abertos. umount -l desliga o ponto da árvore e limpa quando deixar de estar ocupado; -f força em NFS.

Permissões numéricas e simbólicas

r=4, w=2, x=1: 755 é rwxr-xr-x, 644 é rw-r--r--, 600 é rw-------. No modo simbólico, u, g, o e a indicam quem; + e - acrescentam ou removem e = define exatamente. Chave privada SSH usa 600, pois o cliente recusa permissões abertas demais.

Bits especiais

4000 é SUID (o processo roda com o UID do dono, como em /usr/bin/passwd), 2000 é SGID (em diretório, arquivos novos herdam o grupo; chmod g+s) e 1000 é sticky (em /tmp, 1777, impede apagar arquivos de outros). No ls -l, s minúsculo indica bit com execução e S maiúsculo sem execução; o sticky aparece como t no fim.

umask

Arquivos partem de 666 e diretórios de 777, e a umask remove bits. umask 022 dá 644 a arquivos e 755 a diretórios; 027 dá 640 e 750; 077 dá 600 e 700; 002 dá 775 a diretórios. umask sem argumento mostra em octal e umask -S em formato simbólico.

chown e chgrp

chown usuário:grupo muda os dois; chown :grupo muda só o grupo (igual a chgrp); chown usuário: muda o dono e o grupo para o grupo de login. O recursivo é -R maiúsculo. Só o root muda o dono; o dono pode mudar o grupo para um grupo do qual participe.

Links físicos e simbólicos

Link físico aponta para o mesmo inode, só existe no mesmo sistema de arquivos, não vale para diretórios e mantém os dados vivos até o contador de links (número após as permissões no ls -l) chegar a zero. Link simbólico guarda um caminho, cruza sistemas de arquivos e fica quebrado se o destino sumir; se for relativo, é resolvido a partir do diretório onde o link está. ln -s ORIGEM DESTINO cria; ln -sf substitui; readlink -f resolve até o final.

FHS e busca de arquivos

/etc configuração, /var/log logs, /srv dados servidos, /usr/local software compilado localmente, /usr/share dados independentes de arquitetura, /boot kernel/initramfs/GRUB, /dev nós de dispositivos gerenciados pelo udev, /proc e /sys virtuais em memória. locate consulta o banco do updatedb (exclusões em PRUNEPATHS de /etc/updatedb.conf), find percorre a árvore em tempo real, which mostra o executável do PATH e whereis acha binário, fontes e manuais.

Macetes para memorizar

💡 Bits especiais

4-2-1 de novo: 4 SUID (dono), 2 SGID (grupo), 1 sticky (outros, o t no fim).

💡 umask

Arquivo nasce de 666, diretório de 777; subtraia a umask. 022 → 644/755, 027 → 640/750, 077 → 600/700.

💡 Comandos do fdisk

n novo, d delete, p print, w write, q quit sem gravar.

💡 Campos do fstab

Do 'onde' ao 'quando verificar': Dispositivo, Ponto de montagem, Tipo, Opções, Dump, Pass (ordem do fsck: 0, 1 ou 2). Todos os seis campos fazem parte da linha.

💡 Opções do e2fsck

f força, n nega tudo, y yes para tudo, p preen (automático e seguro).

💡 Ferramentas XFS

repair repara, db debuga, fsr reorganiza (file system reorganizer).

💡 Códigos de partição MBR

83 é Linux, 82 é o vizinho de baixo (swap), 8e é LVM.

Tabela de memorização: não confunda

Quando é...	...e quando é
Link físico (ln) Mesmo inode, mesmo sistema de arquivos, sem diretórios; apagar o original não afeta o conteúdo.	**Link simbólico (ln -s)** Guarda caminho, cruza partições, aceita diretórios; quebra se o destino for apagado.
/tmp Temporário que pode ser limpo na reinicialização.	**/var/tmp** Temporário que deve persistir entre reinicializações.
/media Mídias removíveis montadas pelo ambiente gráfico (pendrives).	**/mnt** Montagens temporárias manuais do administrador.
df Uso e espaço livre de cada sistema de arquivos montado (-i para inodes).	**du** Quanto um diretório ou arquivo ocupa (-sh para total legível).
locate Busca rápida por nome em banco indexado; arquivo recente só aparece depois do updatedb.	**find** Percorre a árvore em tempo real e filtra por tipo, tamanho, dono, permissão e datas.
which Qual executável do PATH roda ao digitar o nome.	**whereis** Binário, fontes e páginas de manual em locais padrão.

e2fsck / tune2fs Verificar e ajustar ext2/3/4.	**xfs_repair / xfs_db / xfs_fsr** Reparar, inspecionar e reorganizar XFS.

Números que caem

Item	Valor
Peso do tópico 104 na prova	**23,3% (cerca de 14 questões)**
Reserva padrão de blocos para o root no ext4	**5%**
Bits especiais	**4000 SUID, 2000 SGID, 1000 sticky**
Permissão base antes da umask	**666 para arquivos, 777 para diretórios**
Tipos de partição MBR no fdisk	**83 Linux, 82 swap, 8e LVM, 07 NTFS/exFAT**
Campos do /etc/fstab	**6 (o sexto é a ordem do fsck: 0, 1 ou 2)**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Procurar arquivos SUID com find / -perm -2000.
Como não cair: 4000 é SUID, 2000 é SGID e 1000 é sticky: SUID é find / -perm -4000.

⚠ **Alerta de prova:** Responder 755 para arquivo novo com umask 022.
Como não cair: Arquivo parte de 666: com 022 vira 644. 755 é o diretório.

⚠ **Alerta de prova:** Achar que s na posição do grupo (drwxrwsr-x) é SUID ou sticky.
Como não cair: s no grupo é SGID; SUID apareceria na posição do dono e sticky como t no fim.

⚠ **Alerta de prova:** Usar ln -s para trocar o destino de um link que já existe, ou ln -f sem -s.
Como não cair: ln -s falha se o link existe e ln -f sozinho tentaria criar link físico; o certo é ln -sf.

⚠ **Alerta de prova:** Usar chown suporte: arq.txt para mudar só o grupo.
Como não cair: Com dois-pontos no fim muda o dono e o grupo de login; só o grupo é chown :suporte ou chgrp suporte.

⚠ **Alerta de prova:** Usar chown -r para recursão.
Como não cair: A opção recursiva é -R maiúsculo.

⚠ **Alerta de prova:** Usar df -sh /var/log ou ls -sh para o total de um diretório.
Como não cair: df não aceita -s e ls lista arquivo por arquivo; o total é du -sh.

⚠ **Alerta de prova:** Procurar falta de inodes com du -i ou ls -i.
Como não cair: du -i é inválido e ls -i só mostra o inode de cada arquivo; inodes livres por sistema de arquivos é df -i.

⚠ **Alerta de prova:** Usar e2fsck -n ou -p quando o sistema aparece limpo e se quer verificar mesmo assim.
Como não cair: -f força; -n só responde não sem corrigir e -p repara o que é seguro.

⚠ **Alerta de prova:** Montar ISO com mount -o bind ou remontar a raiz com mount -o noauto,ro.
Como não cair: ISO usa -o loop; bind é para diretório; mudar opções de algo montado é -o remount,ro; noauto só vale no fstab.

⚠ **Alerta de prova:** Confundir os códigos 83, 82 e 8e no fdisk.
Como não cair: 83 Linux, 82 swap, 8e LVM.

⚠ **Alerta de prova:** Usar sticky bit ou SUID no diretório para fazer arquivos herdarem o grupo.
Como não cair: Herança de grupo é SGID no diretório; sticky protege exclusões e SUID não tem efeito útil em diretórios.

⚠ **Alerta de prova:** Colocar /dev/sdb1 direto no fstab como identificação estável.
Como não cair: O nome pode mudar entre inicializações; use UUID= ou LABEL=, vistos com blkid ou lsblk -f.

⚠ **Alerta de prova:** Instalar software compilado localmente em /usr/bin.
Como não cair: /usr/bin é gerenciado pelos pacotes da distribuição; o lugar do software local é /usr/local.

Como cai na prova

📝 **Como cai:** Cálculo de permissão: octal, simbólico ou resultado de umask.
Como resolver: Converto cada trio com r=4, w=2, x=1; para umask, parto de 666 (arquivo) ou 777 (diretório) e removo os bits; confiro se a alternativa é de arquivo ou diretório.

📝 **Como cai:** Cenário de disco: criar tabela, formatar, verificar ou ajustar parâmetro.
Como resolver: Identifico a etapa (particionar: fdisk/gdisk/parted; formatar: mkfs; verificar: e2fsck ou xfs_repair; ajustar: tune2fs) e o tipo de sistema de arquivos antes de escolher a ferramenta.

📝 **Como cai:** Linha ou opção do fstab e do mount.
Como resolver: Lembro a ordem dos seis campos e o que cada opção faz; se o enunciado fala em sistema já montado, procuro remount; se fala em arquivo .iso, loop; se fala em diretório, bind.

📝 **Como cai:** Comportamento de link após apagar, mover ou cruzar partição.
Como resolver: Pergunto se é físico (inode, mesmo sistema, conteúdo preservado) ou simbólico (caminho, quebra se o destino some, relativo depende do diretório do link).

📝 **Como cai:** Onde fica X segundo o FHS.
Como resolver: Classifico o dado: configuração (/etc), variável (/var, /var/log), serviço (/srv), software local (/usr/local), compartilhado de leitura (/usr/share), removível (/media), temporário (/tmp ou /var/tmp).

📝 **Como cai:** Localizar arquivo ou comando.
Como resolver: Executável do PATH: which; binário mais manuais: whereis; por nome rápido: locate (atualizar com updatedb); por atributos: find.

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
espaço livre, mas não cria arquivos	**Inodes esgotados: df -i**
alvo está ocupado ao desmontar	**lsof ou fuser -m; umount -l desmonta quando liberar**
identificar a partição mesmo que a ordem dos discos mude	**UUID= (ou LABEL=) no fstab, valores via blkid**
sexto campo do fstab	**Ordem de verificação pelo fsck (0, 1, 2)**
arquivos novos devem pertencer ao grupo do diretório	**SGID: chmod g+s ou 2xxx**
impedir que um usuário apague arquivos de outros em diretório compartilhado	**Sticky bit: chmod +t ou 1777**
reduzir a reserva de 5% do root no ext4	**tune2fs -m 1**
subvolumes e snapshots nativos	**Btrfs**
locate não encontra arquivo criado há pouco	**Executar updatedb**

updatedb deve ignorar um diretório	**PRUNEPATHS em /etc/updatedb.conf**
conteúdo de um diretório aparecer em outro sem copiar	**mount --bind**
partição de dados não deve executar binários	**Opção noexec**
número logo após as permissões no ls -l	**Quantidade de links físicos**
caminho absoluto final de link encadeado	**readlink -f**

🎯 **Se você lembrar de só uma coisa deste capítulo...**

- Link físico não cruza sistema de arquivos e não aceita diretório; simbólico quebra se o destino sumir.
- e2fsck e tune2fs são de ext; XFS usa xfs_repair, xfs_db e xfs_fsr.
- fsck só em sistema desmontado ou montado somente leitura.
- No fstab, prefira UUID= ou LABEL=; defaults = rw, suid, dev, exec, auto, nouser, async.

Essenciais para a prova (checklist)

- ☐ Sei criar e gravar partições no fdisk e criar tabela GPT com gdisk ou parted mklabel gpt.
- ☐ Sei preparar e ativar swap com mkswap e swapon e conferir com swapon --show.
- ☐ Sei usar tune2fs -l, -L, -m, -c, -i e as opções -f, -n, -y, -p do e2fsck.
- ☐ Sei diagnosticar falta de inodes com df -i e somar diretórios com du -sh.
- ☐ Sei os seis campos do fstab e as opções defaults, noauto, user, noexec, nosuid e nodev.
- ☐ Sei montar ISO com loop, diretório com bind e remontar com remount,ro.
- ☐ Sei calcular permissões octais, aplicar chmod simbólico e resultados de umask.
- ☐ Sei o efeito de SUID, SGID e sticky e como aparecem no ls -l (s, S, t).
- ☐ Sei o comportamento de links físicos e simbólicos ao apagar, mover e cruzar partições.
- ☐ Sei onde fica cada tipo de dado no FHS e quando usar find, locate, which e whereis.

Questões de revisão do capítulo

R4.1 · 104 Dispositivos, Sistemas de Arquivos e FHS

Um disco novo de 4 TB deve usar tabela de partição GPT. Qual ferramenta é própria para criar essas partições?

A. e2fsck
B. tune2fs
C. mkswap
D. gdisk

R4.2 · 104 Dispositivos, Sistemas de Arquivos e FHS

Há espaço livre no disco, mas não é possível criar novos arquivos. Qual comando ajuda a verificar se acabaram os inodes?

A. df -i
B. du -i
C. free -i
D. ls -i

R4.3 · 104 Dispositivos, Sistemas de Arquivos e FHS

Qual comando exibe o superbloco de um sistema ext4, incluindo contagem de montagens e intervalo de verificação?

A. fdisk -l /dev/sda1
B. mount -l /dev/sda1
C. tune2fs -l /dev/sda1
D. mkfs.ext4 -l /dev/sda1

R4.4 · 104 Dispositivos, Sistemas de Arquivos e FHS

O e2fsck informa que o sistema de arquivos está limpo, mas o administrador quer verificar mesmo assim. Qual opção força a verificação?

A. e2fsck -p /dev/sdb1
B. e2fsck -n /dev/sdb1
C. e2fsck -f /dev/sdb1
D. e2fsck -y /dev/sdb1

Respostas comentadas

R4.1. D) gdisk

Por quê: gdisk trabalha com GPT; fdisk moderno e parted também suportam GPT.

✗ A) Incorreta. Verifica ext2/3/4.

✗ B) Incorreta. Ajusta parâmetros de ext.

✗ C) Incorreta. Prepara área de troca.

> **Dica de prova:** Partições GPT: gdisk.

R4.2. A) df -i

Por quê: Muitos arquivos pequenos podem esgotar os inodes antes do espaço.

✗ B) Incorreta. Opção inválida.

✗ C) Incorreta. Opção inválida.

✗ D) Incorreta. Mostra o inode de arquivos.

> **Dica de prova:** Inodes esgotados: df -i.

R4.3. C) tune2fs -l /dev/sda1

Por quê: tune2fs -l mostra e tune2fs -c/-i ajusta verificações periódicas.

✗ A) Incorreta. Lista partições.

✗ B) Incorreta. Lista montagens com rótulos.

✗ D) Incorreta. Criaria um sistema de arquivos.

> **Dica de prova:** Superbloco ext4: tune2fs -l.

R4.4. C) e2fsck -f /dev/sdb1

Por quê: -f força; -n só verifica; -y aceita correções; -p repara de forma automática e segura.

✗ A) Incorreta. Repara automaticamente o seguro.

✗ B) Incorreta. Só responde não, sem corrigir.

✗ D) Incorreta. Responde sim a tudo.

> **Dica de prova:** Forçar verificação: e2fsck -f.

DIA DA PROVA

Estratégia de prova

A prova 101-500 tem 60 questões em 90 minutos, com múltipla escolha, múltipla resposta e preenchimento, e aprovação com 500 numa escala de 200 a 800. O estilo é de administrador na frente do terminal: o enunciado descreve uma tarefa curta e as alternativas são comandos quase idênticos, mudando só uma opção, a ordem dos argumentos ou a ferramenta da família vizinha. O tópico 103 (Comandos GNU e Unix) sozinho pesa 43,3%, seguido de 104 (23,3%), 102 (20%) e 101 (13,3%). As alternativas erradas costumam ser comandos reais que fazem outra coisa, então reconhecer o que cada opção faz vale mais do que decorar só a certa. Nas questões de preenchimento não há alternativa para lembrar: a sintaxe tem de estar na ponta dos dedos.

Como responder

✓ Distribua o estudo pelos pesos: quase metade da nota vem de comandos GNU e Unix (redirecionamento, filtros de texto, find, processos, vi).

✓ Com 90 minutos para 60 questões, mantenha ritmo de cerca de 1,5 minuto por questão e marque para revisão as que travarem.

✓ Em cada alternativa, pergunte o que aquele comando faria se você o digitasse; a errada geralmente faz algo real, mas diferente do pedido.

✓ Identifique o que o enunciado entrega (nome de pacote, caminho de arquivo, PID, nome de processo, arquivo de pacote) antes de escolher a opção.

✓ Em múltipla resposta, procure os pares que atendem ao mesmo pedido (kill e kill -15, bunzip2 e bzip2 -d, dpkg -l e rpm -qa, dmesg e journalctl -k) e confira quantas devem ser marcadas.

✓ Para preenchimento, treine digitar os comandos completos com opções exatas, respeitando maiúsculas e minúsculas (-R, -J, -P, -S).

✓ Descarte alternativas da família de distribuição errada: apt/dpkg é Debian, rpm/dnf/yum é Red Hat, zypper é SUSE.

✓ Em cenários de tempo (agora x próximo boot, temporário x permanente), decida isso primeiro: isso separa isolate de set-default e a tecla e do GRUB de /etc/default/grub.

Erros que derrubam candidatos

✗ Trocar opções vizinhas do mesmo comando: dpkg -L e -S, rpm -ql e -qf, grep -c e -n, uniq -d e -u.

✗ Errar a ordem dos redirecionamentos (2>&1 antes do >) e usar uniq sem sort.

✗ Confundir maiúsculas e minúsculas em opções: chown -r em vez de -R, tar -j em vez de -J, dpkg -p em vez de -P.

✗ Aplicar a umask como se arquivo e diretório partissem do mesmo valor (666 x 777).

✗ Editar arquivo gerado (grub.cfg) ou de pacote (/usr/lib/systemd/system) em vez do arquivo de configuração correto.

✗ Confundir ferramentas de ext com as de XFS e rodar fsck em sistema montado.

✗ Escolher comando que só consulta quando o enunciado pede alterar, ou vice-versa (get-default x set-default, runlevel x telinit, check-update x upgrade).

Na semana da prova

- ☐ Refaça as provas simuladas cronometradas em 90 minutos e revise cada alternativa errada, não só a certa.
- ☐ Revise as tabelas de opções que mais caem: dpkg, rpm -q, apt-get, grep, sort/uniq, tar, find, kill/sinais, tune2fs/e2fsck, systemctl e journalctl.
- ☐ Pratique no terminal os pipelines, redirecionamentos e o vi; digitar fixa a sintaxe para as questões de preenchimento.
- ☐ Faça uma folha de umask, permissões especiais, campos do fstab e equivalência runlevel/target e revise na véspera.
- ☐ Na véspera, só revisão leve dos macetes e dos itens de lembrar de cada tópico; nada de conteúdo novo.

REVISÃO FINAL

Folha de cola

Leia na véspera e na manhã da prova: os gatilhos e as pegadinhas mais importantes de cada domínio.

1. 101 Arquitetura do Sistema

‣ nível anterior e atual no SysV → **runlevel**

‣ nível padrão no SysV → **id:N:initdefault: em /etc/inittab**

‣ links S e K em /etc/rc3.d → **S inicia, K encerra o serviço naquele nível**

‣ esqueci a senha de root, parâmetro no GRUB → **init=/bin/bash**

‣ erro no fstab, shell mínimo, raiz só leitura → **emergency.target**

⚠ Padrão persistente é set-default; enable não define o alvo padrão e isolate vale só para a sessão atual.

⚠ set-default só afeta a próxima inicialização; imediato é systemctl isolate rescue.target.

⚠ Isso é mask; disable só tira do boot.

2. 102 Instalação do Linux e Gerenciamento de Pacotes

‣ parâmetro do kernel só nesta inicialização → **Tecla e na entrada do menu do GRUB**

‣ parâmetro permanente / todas as inicializações → **GRUB_CMDLINE_LINUX em /etc/default/grub + grub-mkconfig ou update-grub**

‣ caiu no prompt grub rescue → **set root, linux, initrd e boot**

‣ ordem das entradas de boot no firmware UEFI → **efibootmgr**

‣ libfoo.so.1 => not found → **Biblioteca ausente: instalar ou ajustar /etc/ld.so.conf e rodar ldconfig**

⚠ No APT a versão vem com sinal de igual (nginx=1.24.0-1); dois-pontos indicam arquitetura e o hífen formaria outro nome de pacote.

⚠ O grub.cfg é gerado automaticamente; o lugar é /etc/default/grub, seguido de grub-mkconfig ou update-grub.

⚠ update-grub é atalho do Debian e não usa -o; no Red Hat é grub2-mkconfig -o /boot/grub2/grub.cfg.

3. 103 Comandos GNU e Unix

‣ gravar saída e erros no mesmo arquivo → **comando > arq 2>&1**

‣ descartar apenas as mensagens de erro → **2>/dev/null**

‣ mostrar na tela e gravar ao mesmo tempo → **tee (tee -a para acrescentar)**

‣ rodar o segundo comando só se o primeiro der certo → **&&**

‣ continuar rodando depois do logout do SSH → **nohup comando & (ou screen/tmux)**

⚠ uniq só compara linhas adjacentes; primeiro sort, depois uniq (e uniq -c para contar).

⚠ Redirecionamentos são avaliados da esquerda para a direita: o 2>&1 copiou a tela antes; o certo é > arq 2>&1.

⚠ tee sozinho sobrescreve; tee -a acrescenta.

4. 104 Dispositivos, Sistemas de Arquivos e FHS

‣ espaço livre, mas não cria arquivos → **Inodes esgotados: df -i**

‣ alvo está ocupado ao desmontar → **lsof ou fuser -m; umount -l desmonta quando liberar**

‣ identificar a partição mesmo que a ordem dos discos mude → **UUID= (ou LABEL=) no fstab, valores via blkid**

‣ sexto campo do fstab → **Ordem de verificação pelo fsck (0, 1, 2)**

‣ arquivos novos devem pertencer ao grupo do diretório → **SGID: chmod g+s ou 2xxx**

⚠ 4000 é SUID, 2000 é SGID e 1000 é sticky: SUID é find / -perm -4000.

⚠ Arquivo parte de 666: com 022 vira 644. 755 é o diretório.

⚠ s no grupo é SGID; SUID apareceria na posição do dono e sticky como t no fim.

GABARITO

Respostas do diagnóstico

D1. D) init=/bin/bash → revise o capítulo 1

Por quê: init= substitui o programa inicial; systemd.unit= escolhe um alvo.

✗ A) Incorreta. Define a partição raiz.
✗ B) Incorreta. quiet reduz mensagens.
✗ C) Incorreta. ro monta a raiz só leitura.

> **Dica de prova:** Shell no lugar do init: init=/bin/bash.

D2. C) systemctl default → revise o capítulo 1

Por quê: systemctl default equivale a isolar o default.target.

✗ A) Incorreta. Não é um subcomando válido.
✗ B) Incorreta. Exige o nome da unidade.
✗ D) Incorreta. Reabilita unidades.

> **Dica de prova:** Voltar ao alvo padrão: systemctl default.

D3. C) systemctl set-default multi-user.target → revise o capítulo 1

Por quê: set-default altera o link default.target; isolate muda o estado atual sem persistir.

✗ A) Incorreta. Não define o alvo padrão.
✗ B) Incorreta. Muda o alvo só na sessão atual.
✗ D) Incorreta. Apenas mostra o alvo padrão.

> **Dica de prova:** Alvo padrão persistente: systemctl set-default.

D4. B) A versão principal da biblioteca → revise o capítulo 2

Por quê: O soname inclui a versão principal; mudanças incompatíveis mudam esse número.

✗ A) Incorreta. Não é essa informação.
✗ C) Incorreta. Não é essa informação.
✗ D) Incorreta. Não é essa informação.

> **Dica de prova:** libx.so.N: versão principal (soname).

D5. C) dpkg -L openssh-server → revise o capítulo 2

Por quê: dpkg -L lista arquivos; -s mostra status; -c lê um arquivo .deb; -P expurga.

✗ A) Incorreta. Mostra o status do pacote.
✗ B) Incorreta. Remove o pacote e as configurações.
✗ D) Incorreta. Lista o conteúdo de um arquivo .deb, não de um pacote instalado.

> **Dica de prova:** Arquivos de um pacote Debian: dpkg -L.

D6. B) dpkg -s curl → revise o capítulo 2

Por quê: dpkg -s exibe o status (instalado ou não) e metadados do pacote.

✗ A) Incorreta. Lista conteúdo de um .deb.

✗ C) Incorreta. Remove o pacote.

✗ D) Incorreta. Instala um arquivo .deb.

> **Dica de prova:** Pacote instalado?: dpkg -s.

D7. A) comando > saida.log 2>&1 → revise o capítulo 3

Por quê: 2>&1 aponta o stderr para onde o stdout aponta; a ordem importa.

✗ B) Incorreta. Usa um descritor não configurado.

✗ C) Incorreta. Envia a saída para um comando chamado saida.log.

✗ D) Incorreta. Descarta os erros.

> **Dica de prova:** Saída e erros juntos: > arquivo 2>&1.

D8. C) yy e depois p → revise o capítulo 3

Por quê: yy copia (yank) a linha; p cola abaixo e P acima.

✗ A) Incorreta. Altera a palavra e sai.

✗ B) Incorreta. Apaga um caractere e cola acima.

✗ D) Incorreta. Apaga e desfaz.

> **Dica de prova:** Copiar e colar linha no vi: yy e p.

D9. A) uniq -d lista.txt → revise o capítulo 3

Por quê: uniq -d mostra duplicadas, -u únicas e -c conta.

✗ B) Incorreta. Conta todas as linhas.

✗ C) Incorreta. Ignora maiúsculas na comparação.

✗ D) Incorreta. Mostra só as linhas únicas.

> **Dica de prova:** Só repetidas: uniq -d.

D10. C) UUID=identificador → revise o capítulo 4

Por quê: UUID e LABEL identificam sistemas de arquivos de forma estável; blkid mostra esses valores.

✗ A) Incorreta. Não identifica o dispositivo.

✗ B) Incorreta. Não identifica o dispositivo.

✗ D) Incorreta. Pode mudar entre inicializações.

> **Dica de prova:** Identificação estável no fstab: UUID.

D11. C) Eles apontam para inodes do mesmo sistema → revise o capítulo 4

Por quê: Links simbólicos podem cruzar sistemas de arquivos; links físicos não.

✗ A) Incorreta. Usuários comuns podem criá-los.
✗ B) Incorreta. Não são permitidos para diretórios.
✗ D) Incorreta. Não é verdade.

> **Dica de prova:** Link físico entre partições: não é possível.

D12. B) Um link simbólico → revise o capítulo 4

Por quê: Tipos: - arquivo, d diretório, l link, b bloco, c caractere, p pipe e s socket.

✗ A) Incorreta. Seria b.
✗ C) Incorreta. Seria -.
✗ D) Incorreta. Seria d.

> **Dica de prova:** Primeiro caractere l: link simbólico.

Agora, treine no formato real

Faça as 6 provas completas do curso LPIC-1 101: 360 questões inéditas, tempo do exame, explicação de cada alternativa e link da documentação para revisar.

Ver as 6 provas completas e o cupom do mês →

Até o dia da prova, uma revisão por dia no seu feed

Instagram @strecht.simulados
Um post de prova por dia: questão comentada, pegadinhas e o "não confunda" de cada certificação.
Seguir no Instagram →

YouTube Strecht Simulados
Shorts diários com pegadinhas de prova e, aos sábados, um simulado comentado.
Inscrever-se no canal →

Material independente de preparação, sem afiliação ou endosso de Linux Professional Institute (LPI). As questões são inéditas e não reproduzem a prova real.