**Curso Linux**

**MÓDULO 1**

* **Sistema Operacional:**

É um conjunto de programas interrelacionados para interface entre usuário e hardware. Ex: Windows, Linux e MacOS

O Linux é usado, principalmente, para servidores, e ainda há espaço para ser usado para computadores pessoais, no desenvolvimento de software, pois tem uma grande quantidade de ferramentas disponíveis para devs.

A “tela visual” do OS é o ambiente gráfico, já a parte que liga o hardware e o software, a gente chama de Kernel (é o núcleo do OS).

O Linux, ou o Kernel dele, é open source e pode-se fazer várias adaptações ou otimizações para determinado tipo de uso ou grupo de usuários. Essas “versões” são chamadas de distribuições. Ex: Debian, Ubuntu, Red Hat, Fedora, Suse – Ubuntu, Debian e Fedora são as distribuições principais (inclusive para o mercado de trabalho), enquanto a maioria das outras são variações.

Elas, inclusive, podem ser usadas em sistemas embarcados, que são a combinação de hardware e software para uma função específica, como por exemplo os celulares Android – baseado em Linux com adaptações e ambientes gráficos (tanto que mesmo todos os aparelhos Android tenham uma interface parecida, elas levam pequenas diferenças, assim como ocorre com as distribuições citadas).

Para instalação, sempre que por procurar uma versão, opte pela LTS (long term support), ou seja, são super testadas e contam suporte de longo prazo.

* **Ambientes gráficos:**

Você pode trabalhar com diferentes ambientes gráficos, sua função é facilitar a utilização do sistema operacional, ou o kernel dele, pelo usuário. No Windows são somente customizações simples, já no Linux, apesar de vir com um padrão você simplesmente pode mudar para outro completamente diferente.

O ambiente gráfico mais famoso do Linux é o GNOME, que, além da interface, possui apps específicos (tipo explorador de arquivo, reprodutor de vídeo e imagem, etc).

Outro famoso é o KDE, também tem o Cinnamon – derivado do GNOME para uma distribuição específica do Linux. A diferença do KDE para o GNOME, é que o GNOME tem um ambiente mais simplista, enquanto o KDE é mais rico em recursos e altamente personalizado.

Algumas nomenclaturas diferentes: o “menu iniciar” é chamado de Dash. A tecla “windows” chama “super”. Já em comandos, win + pg up/dwn permite que você ande entre diferentes workspaces, enquanto a tecla win sozinha mostra todos os workspaces.

* **Instalações via arquivo**

Arquivos .deb são para Debian/Ubuntu, enquanto .rpm são para Fedora ou Open SUSE (que é baseado em Red Heat).

* **Sistema de arquivos e dispositivos de armazenamento**

Sistema de arquivos são conjuntos de estruturas lógicas que permite que o OS controle o acesso a dispositivos de armazenamento. Existe o NTFS – mais usado em Windows, o FAT 32 –usado nas versões antigas do Windows, o EXT4 usado pelo Linux, e o Fedora usa BTRFS – que é projetado para tirar o máximo de proveito do dispositivo SSD e capacidades avançadas de criação e gerenciamento de snapshots (cópias instantâneas do estado de um sistema de arquivos em um determinado momento), o que o torna uma escolha atraente para tarefas de backup e recuperação.

Curiosidade: o EXT4 usa o sistema de jornalização (journaling). Que é, registrar todas as operações em um “jornal”, antes que elas sejam efetivamente armazenadas. Com a jornalização, é possível recuperar o sistema de arquivos para um estado consistente após panes ou desligamentos inesperados.

Ou seja, o Windows não lê EXT4 naturalmente, precisa de um software pra isso, mas o Linux lê os padrões de Windows, que inclusive é muito poderoso!

Você ainda pode particionar os dispositivos de armazenamento e colocar cada partição com um sistema diferente. O dispositivo será reconhecido como duas unidades diferentes (tipo drive D e drive E, por exemplo).r

Tecnologias usadas para particionar: MBR (Master Book Record ou MS-DOS), que permite até 4 partições, e o GPT (GUID Partition Table), que é o Modelo Padrão mais novo, e por isso não funciona em OS antigos e permite até 126 partições.

Já a partição do Boot, é sempre bom usar FAT32.

* **Simular a adição de um novo HD no computador**

Se faz isso através das máquinas virtuais (no Linux: configurações>armazenamento>add em “controlador”>criar. Usar a VDI (Virtual Destktop Infrastructure).

Depois de criado, é necessário formatar a partição (já dentro do OS, nos drives). Escolher o formato – em “outro” você ainda encontra a partição Swap, que é específica para memória (mas não é o caso desta máquina virtual que estamos criando).

Depois de formatado, tem que montar a partição. Você pode reformatar sempre que quiser mudar o formato. Para a partição se manter sempre montada, é necessário desabilitar a opção “padrão de sessão do usuário”.

Vale ressaltar que, desta forma, você não tem a opção de particionar o disco que você criou, pra isso você precisa do software Gparted e o disco precisa estar desmontado.

As lista de partições chamam “tabela de partição”, ela irá existir, obviamente só quando você começar a particionar.

* **Criando novo usuário**

Em “configurações”, você pode criar um novo usuário (admin tem total poder, o outro tem limitações, tipo não pode instalar nada).

**MÓDULO 2**

* **Terminal (Shell)**

O terminal é a “caixa de texto”, o Shell é quem interpreta os comandos. @ significa “em” ou “at”. Ou seja: fernando@fedora = o user “Fernando” está no host chamado “Fedora”.

Os códigos incluem os comandos (o código básico, tipo “ls”, “mkdir”, “touch”, etc), na sequencia vem o argumento ou opções, que permitem que você personalize o comportamento dos comandos de acordo com suas necessidades (tipo: “ls -l”, etc).

* **Comandos básicos de movimentação**

Os comandos são sensíveis, então tem que respeitar se for letra maiúscula ou minúscula

$ indica que o user não é o admin (se você estivesse logado em um user boot, ao invés do $ seria #)

~ significa que estou logado na pasta de usuário

date: data, hora, fuso

clear: limpar tela (o atalho é Ctrl + L)

pwd: mostra o diretório em que você está (é sempre bom se localizar antes pra ter certeza)

cd + local: troca o diretório para o local indicado (raiz é somente a barra. Ex: cd /)

ls: lista arquivos no diretório

cd ..: volta um diretório acima

cd .. + nome de novo diretório (dentro da pasta que você vai voltar, já te coloca dentro do novo diretório que você criou).

A tecla tab é autopreencher

* **Filtrando a exibição de arquivos**

cd –: volta para onde estava

ls | more: (a barra reta chama “pipe”) no Linux server, não tem barra de rolagem no terminar, então para ver tudo você usa esse comando com estes argumentos para ir listando gradativamente, você vai dando enter para ver mais

Ctrl + c: cancela a rolagem acima, ou pesquisas, etc).

ls + nome de um arquivo dentro do diretório: mostra só o arquivo

No terminar é possível usar filtros ou caracteres coringa, por exemplo = ls + a\* (lista todos os arquivos começados em A, e se houver algum diretório no resultado, vai mostrar também o que tem dentro deles)

ls g?o\* (a 1ª letra tem que ser g e a 3ª o, o \* representa que depois disso não tem mais critério)

cd ~ vai para a pasta do usuário

touch + nome.formato: você cria um arquivo (pode criar vários de uma vez. Ex: texto1.txt texto2.txt texto3.txt)

ln você cria um link. Nele, o -s indica “simbólico”, um link simbólico é um atalho! Ex: ln -s /caminho/do/arquivo.txt /caminho/do/link\_simbolico

Depois, se você quiser listar estes, você digita “ls texto[2-4].txt – é bom para copiar ou excluir arquivos, por exemplo. Se você troca o – por , lista exatamente os itens indicados (no caso o texto2.txt e o texto4.txt). Se você colocar ^ antes, indica “todos menos este”.

Se eu usar o LS + diretório, eu consigo listar o que tem no diretório sem entrar nele e posso ir navegando em mais camadas também.

Find –name + nome do arquivo: procura o arquivo dentro do diretório, desde que tenha permissão. Lembrando que Ctrl + C interrompe o processo.

No find, se você completar com um -exec, você executa os arquivos encontrados. Olhe este exemplo: find /caminho/do/diretorio -name "\*.txt" -exec cp {} /caminho/do/diretorio\_de\_destino \;

Até o nome do arquivo ok, mas na sequência o argumento -exec indica que é para executar uma ação, o cp indica que esta ação é de cópia, o {} é um espaço reservado que representa os arquivos encontrados, e, por fim, o diretório destino.

* **Criar diretórios**

(tem que ter permissão)

mkdir + nome (você ainda pode fazer: mkdir + caminho/nome e criar no diretório que você quiser sem estar nele. Se você usar mkdir -p + caminho/nome, você cria todo o caminho caso ele não exista (-p pode ser substituído por --parents)

Para criar arquivos ou diretórios com nome composto, usar aspa simples, se esquecer disso vai criar um arquivo pra cada palavra. Para acessá-los, é a mesma coisa. Se aplica também se, de fato, você quer criar vários de uma vez, assim como o exemplo do texto1.txt

* **Excluir diretórios**

rmdir + nome (o diretório não pode ter arquivos dentro): exclui diretório

Neste caso também é possível usar filtros: rm\*.txt (excluirá todos os .txt), rm c\*(exclui todas as pastas começadas com c)

Rm -rf + nome do diretório ai exclui tudo, o diretório em questão e os arquivos dentro dele.

* **Obtendo ajuda**

A ajuda é um geral sobre o comando, ativado por --help, ele dá a lista de possíveis opções/argumentos para você incrementar o comando. Você ainda pode usar vários de uma vez, por exemplo ls -l ou ls -lh.

Também pode usar o man + comando, mas ai é em inglês

* **Super Usuário**

sudo – executa tarefa como admin de forma segura, já que o user admin é o root – que não é adequado ficar logado. Inclusive o sudo pede senha.

Desta forma, se eu quiser, por exemplo, criar uma pasta dentro do root (lembrando: cd / e em seguida mkdir [nome da pasta]) eu não conseguiria, mas se eu usar o sudo mkdir [nome da pasta], ai rola.

Se você ver os detalhes do arquivo criado na pasta (lembrando: ls -l), você vai ver que o dono é o root.

su – você loga como super usuário – o root – mas é bom lembrar que não deve deixar logado assim por questão de segurança. Logado no su, você pode fazer o que quiser, não precisando usar, portanto, o sudo.

Para deslogar, é só usar su [nome do user].

Lembrando que # significa logado no root, e $ significa logado no usuário convencional.

Vale lembrar que no Fedora, o usuário root não é pré-instalado. Já o Debian não dá permissão de su para convencionais (sudo).

* **Criando/Alternando para super usuário**

No Debian, o sudo não vem configurado. Para fazer isso você precisa entrar no diretório root, pasta raiz, e digitar nano /etc/sudoers, pois você precisa abrir um editor de texto para configurar o sudoers e nano é um editor de texto. Neste arquivo você poderá liberar privilégios para usuário ou grupo específico.

Vale destacar que o diretório /etc é onde estão localizados arquivos de configuração de sistema.

Para liberar usuário, é só editar no #User privilegie Specification, colocar o nome do usuário e copiar as liberações do root.

Se for grupo, é no campo de baixo “#Allow members of group sudo to execute any command”

No menu do editor, você verá o ^, isso significa a tecla Ctrl do lado esquerdo, só seguir a legenda para salvar e sair.

No Ubunto e Fedora, o sudo está configurado, mas não tem o root. Para configurar basta usar o sudo password root (que é um comando de mudança de senha), criar uma senha e está pronto.

* **Trabalhando com arquivos de texto**

Existem dois mais utilizados, o vi e o nano. O vi é mais rudimentar, já o nano, mais fácil de lidar, inclusive mais intuitivo. Vale ressaltar que existem outros softwares de edição de texto, mas tem que puxar.

Ambos é só digitar o comando com o nome deles e os argumentos que quiser “vi texto.txt” ou “nuno texto.txt”, você irá abrir pelo editor escolhido ou criar um arquivo de texto que irá abrir na sequência.

No vi, você deve apertar “i” para entrar no modo inserção. Esc sai do modo inserção e o menu abre com a tecla : - w grava, enquanto q sai, e você pode digitar os dois juntos “wq”. O nano é tranquilo e intuitivo.

cat – comando para visualizar o conteúdo do arquivo de texto sem abrir o editor (ex: cat texto.txt)

* **Histórico de comandos**

Seta para cima ou baixo você vai vendo os comandos listados (lembrando que é o histórico do usuário, se trocar de usuário, é outro histórico).

history - lista todos os últimos comandos, se você colocar um numeral na frente (ex: history 10), vai listar os últimos comandos de acordo com o número. É útil para segurança

numeral ! repete o comando que está no numeral na lista.

!! repete o ultimo comando

!? [termo ou palavra chave]? – procura comandos pela palavra chave e o repete.

* **Criando e excluindo usuários**

Existe o user comum ou doméstico, admin (pra configurar pcs) e servidores (usuários irão utilizar serviços deste servidor, como compartilhamento de arquivos, servidor FTB (web), etc.

Para fazer isso só logar no root e digitar “useradd [nome do novo usuário]”, se você acrescentar os argumentos -m cria o diretório pessoal, enquanto -c (que é de comentário), já dá o nome completo para o usuário (ai neste caso, em aspa simples).

Em seguida “passwd”, que coloca uma senha para o usuário.

userdel – deleta o user. Colocando -f (forçado), você força a exclusão caso haja erro, e já pode emendar com -r para deletar os arquivos pessoais junto.

* **Bloqueando e editando usuários**

Em useradd --help tem algumas opções que valem uma atenção extra:

-G atribui o user a um grupo

-d troca o diretório padrão do user (não recomendado)

-e data para a conta ser expirada (Ex: useradd [user] -e dd/mm/aaaa

usermod serve para fazer alterações no usuário. Ex: usermod [user] -e dd/mm/aaaa – c ‘João da Silva’

passwd [user] -e dd/mm/aaaa, coloca data para expirar a senha

passwd [user] -l bloqueia a conta, enquanto -u desbloqueia (bloqueado o user nem fica listado)

cat/etc/passd – lista todos os users criados e dados do user (senhas aparecem criptografadas). Ex: user:senha:id:id grupo:nome user:dir padrão:command shell

O command shell é o caminho de aplicação que irá identificar os comandos do user (se na criação do user u uso -m, vai ser bin/bash, pois o -m faz a criação completa no ambiente gráfico. Sem ele fica bin/sh – o que não impede de todos os users possam usar o ambiente gráfico.

Já os “nologin” não podem fazer logon no ambiente, isso porque são users apenas de uso do sistema.

hostname mostra o nome do Host do sistema, se você incrementar com -s, mostra a versão mais curta.

Eu ainda posso abrir essa lista no nano /etc/passd e editar absolutamente tudo na mão.

* **Arquivos de senhas (passwd)**

Vale destacar que um Linux server tem ambientes com centenas de usuários cadastrados.

Grep é o comando que vou usar para buscar os usuários que usam o ambiente, já que ele busca por padrões em arquivos de texto. Enriquecendo com -i, ele vai passar a menosprezar maiúsculas e minúsculas. Ex: em root – grep -i ‘fernando’ /etc/passwd

Isso vai abrir as infos do usuário, incluindo a senha criptografada.

As senhas criptografadas ficam em um arquivo chamado shadow (grep -i ‘Fernando’ /etc/shadow). Este comando fará aparecer as infos do usuário, mas com novas informações na senha (que é o segundo padrão das informações do usuário, lembrando que o primeiro é o nome do usuário).

Dentro destas informações de senha, cada padrão tem um significado:

1º é o tipo de criptografia (ex: $6$, $1$, $2ª$, $2y$, $5$). A sigla $6$, por exemplo, é a criptografia SH512.

Vale ressaltar que tem sites que geram criptografias no formato que você quiser.

2º padrão é quantos dias ela foi gerada (contando a partir da data de 1/1/1970), então para descobrir quanto tempo esta senha foi criada tem que fazer uma conta no excel: data de hoje – data padrão, e em seguida subtrair do resultado os dias da data.

3º é o mínimo de dias restantes sem poder alterar a senha (se não houver especificações, o valor é zero, isso significa que pode alterar quando quiser).

4º padrão é o numero de dias de validade da senha (sem especificação é 99999)

5º é o numero de dias que, antes da senha expirar, o usuário irá receber um alerta (sem especificação, o padrão é 7 dias)

6º é o número de dias em que, com a senha expirada, a conta será deletada (sem especificar fica vazio)

7º quantos dias desde 1970 a conta está parada

Vale lembrar que este arquivo é editável (nano /etc/shadow). É bom sempre verificar sua integridade, usando para isso o comando “stat” (stat /etc/shadow).

* **Grupos de usuários**

Estes grupos são utilizados para que seja possível agrupar usuários de acordo com suas permissões. Desta forma, qualquer alteração no grupo, altera todos os usuários que fazem parte dele.

Os grupos estão em um arquivo chamado “group” (/etc/group no root). Vale lembrar que este arquivo pode ser ativado com “cat”, “nano”, “stat”, etc.

Vale lembrar que, por padrão, cada usuário representa um grupo, e as vezes apps instalados geram grupos também.

Ao abrir o arquivo, as informações serão no seguinte formato: nome do grupo:senha do grupo (por padrão não existe):ID do grupo

groupadd [nome do grupo] cria-se um grupo

useradd -c ‘[usuário]’ -G [nome do grupo], já adiciona a pessoa no grupo escolhido

usermod [user] -G [grupo], você move o usuário para o grupo desejado

groupdel [grupo] exclui o grupo

* **Conhecendo o sistema de permissões**

Quando você lista os arquivos em um diretório (ls -l), o primeiro caractere indica o tipo de arquivo. “d” é um diretório e “-“ é um arquivo. Na sequencia você verá outros caracteres, sendo que R significa “read”, W “write” e X “execute”.

Cada três caracteres, representa um tipo de permissão: dono, grupo e outros usuários.

* **Alterando permissões de arquivos/diretórios**

chown é usado para mudar donos de arquivo ou diretório.

Ex: “chown usuário /diretório/” você muda o dono | “chown usuário:grupo /diretório/” você muda o grupo que é dono do diretório.

chmod altera as permissões, lembrando que são três caracteres que devem acompanhar o chmod, cada um representa as permissões que serão concedidas para dono, grupo e outros usuários.

Para ilustrar, 4 significa read, 2 write e 1 execute, e esses números podem ser somados. Por exemplo: chmod 770, significa que o dono tem permissão de abrir, alterar e executar, o grupo também, e terceiros nenhuma permissão.

Vale destacar que, quando você está editando as permissões de um diretório, caso você queira que elas se apliquem a arquivos e subpastas que estão dentro do diretório, você deve usar -R. Ex: chmod 761 -R.

Também é aplicável ao chown. Ex: chown root:grupo /diretório/ -R.

Chown root:grupo /diretório/, você assume que o dono é o root, mas ele pertence ao grupo indicado. Na sequência, você usa o chmod para editar todas as permissões, se for necessário.

* **Gerenciamento de pacote (Ubuntu/Debian)**

Gerenciamento de pacote serve para instalar, remover e atualizar apps através de linhas de comando. Lembrando que Ubuntu e Debian são do mesmo núcleo de distribuição, e usam pacotes tipo .deb

apt – get é uma das formas de fazer isso, mas é a mais antiga

apt é a forma atual

apt list abre tudo que tem disponível para instalação, você ainda pode colocar uma palavra chave na frente para ajudar a encontrar

apt search e a palavra chave, também procura a instalação desejada, inclusive dependências do próprio app que você busca. Também vem com informações detalhadas.

apt install + nome instala o que você busca, se você colocar -y na sequencia, ele dá “yes” em todas as perguntas na sequencia

apt remove + nome remove o app

apt update puxa a lista de updates

apt upgrade reinstala as atualizações necessárias após o update.

* **Gerenciamento de Pacotes (Fedora, Red Hat, CenTOS)**

Lembrando que estes pacotes são tipo .rpm

yum é o gerenciador antigo

dnf e o mais moderno, e tem todas as funcionalidades do .deb como lista, search, -y. A maior diferença é o update, pois ele não precisa fazer update e upgrade, somente o update atualiza tudo.

* **Instalando via SNAP**

SNAP é uma aplicação que permite instalar, atualizar e remover SNAPs – que são softwares auto contidos, ou seja, não precisa realizar instalação para que o software seja executado, ele já joga os arquivos em uma pasta, chamada SNAP e sempre que precisar, ele vai nela e executa.

A diferença é que ele permite executar o mesmo software em diferentes distribuições (embora haja excessos).

(Vale lembrar que o SNAP é da Canonical, mesma empresa do Ubuntu, por isso já é pré instalado no Ubuntu. Já no Fedora precisa instalar. Se você entrar no site oficial (snapcraft.io), ele dao passo a passo de como instalar no terminar cada app, e também o próprio SNAP.

Inclusive, durante a instalação do SNAP, é necessário criar um link simbólico, que é tipo o atalho mas usando o terminal. Ex: ao invés de usar /var/lib/snapd/snap, você só precisa usar /snap quando tiver um link simbólico.

* **Realizando instalação de arquivos RPM via DNF**

As vezes você não encontra determinado pacote (tipo para abrir arquivo .rar), você pode ir no rpmfind.net e procurar. Escolha sempre a versão de cima, a mais atual e copia o hyperlink.

wget + link para puxar, em seguida você ve o nome do arquivo e usa:

dnf install + nome do arquivo (você pode usar tab para autocompletar, caso o nome seja grande e complexo de escrever).

* **Particionando e formatando discos via terminal**

Vale ressaltar que há pequenas diferenças entre as distribuições para fazer esta ação.

shutdown 0 indica que o computador deve ser desligado imediatamente

reboot reinicia o computador

Vale lembrar que para criar uma configuração do VirtualBox, eu preciso ir nas configurações da maquina virtual > armazenamento > controlador SATA > add HD > criar > VDI

fdisk formata o disco, se colocar -l ele mostra as partições do HD)

fdisk + caminho do hd (/dev/sdb) e o “m” na sequencia, abre ajuda para você escolher o que fazer, neste caso vamos de “n” que adiciona uma nova partição, em seguida escolhe o tipo e numero de partições. Finalizado, o “w” grava

mkfs.ext4 (ou o padrão escolhido) /dev/sdb

* **Montando discos via terminal**

Para montar, preciso especificar o diretório, então vou criar com mkdir /disco2

E em seguida o comando: mont (monta) -t (tipo de arquivo) ext4 (escolhido) /dev/sdb /disco2

Desta forma, a partição não ficará montada caso reinicie a máquina, sendo assim, para que esse estado seja permanente tem de fazer uma alteração:

Nano /etc/fstab, e em seguida no arquivo:

/dev/sdb (que é o disco) /disco2 (onde vai ser montado) ext4 (tipo) defaults 0 0 (que são os parâmetros).

reboot – reinicia a máquina

df ou df -h você tem informações sobre os discos

* **Editando a variável de sistema $PATH**

Formatar, particionar ou montar no Ubuntu ou Fedora não há diferença, mas no Debian sim. Isso porque o fdisk não está adicionado à variável PATH. O fdisk fica na pasta /sbin (em qualquer sistema), desta forma, se não estiver adicionado ao PATH, você tem de ir até a pasta e ainda executar usando ./ antes (./fdisk), para evitar essa trabalheira toda que se adiciona ao PATH.

Para checar a variável PATH usa-se o echo (comando para printar/escrever) $PATH

Para adicionar o diretório, usa-se export PATH=$PATH:/sbin – aqui estou exportando para o path o que já está na variável (“export para o path o path e o sbin”), a utilização do $ é feita para que o export não subescreva o que já está no path.

* **Renomeando/movendo arquivos**

cp – copia o arquivo (cp [arquivo] [nome da cópia] e ainda pode colocar um destino)

mv – renomeia (mv [arquivo] [novo nome] ou move )mv [origem]/arquivo [destino])

Lembrando que os filtros sempre podem ser usados, se no nome do arquivo eu colocar \*, vai ser todos os arquivos, se for \*.txt, vai ser todos os arquivos de texto, e assim vai. É bom sempre incrementar com o -v (verboso), que exibe informações detalhadas sobre o processo de cópia ou de mover arquivos. -R (recursivo) inclui todos os arquivos e diretórios, já o -u é mais usado para cópia de backup já que, em caso de já haver um arquivo de mesmo nome, ele só subescreve se o arquivo de origem for mais atual que o arquivo no destino.

* **Excluindo arquivos e diretórios**

rm – remove, -i questiona antes de cada remoção, -f força a exclusão sem perguntar nada. Como todo comando, você pode estar no diretório, ou pode indicar o caminho.

Vale lembrar que pelo Virtual Box é possível criar snapshots (pontos de restauração) das máquinas virtuais.

* **Instalando outros ambientes gráficos**

No Ubutu – apt install kde-standard (ou full, mas não precisa). Demora mais na máquina virtual. Para trocar de ambiente é só voltar para a tela de login do usuário, que tem um botão para escolher qual OS você quer.

Já no Fedora – dnf group install “Deepin Desktop”. A troca de OS também é igual a do Ubuntu.

* **Wine – executando APPS do Windows no Linux**

Wine é um software que cria um ambiente similar ao do Windows para que você consiga executar apps do Windows (mas não todos). A palavra-chave mais precisa na busca de apps é “winehq”.

Depois você precisa puxar o play on Linux, que é o app que você vai abrir para executar o software do Windows.

**MÓDULO 3**

* **O que é um servidor?**

O servidor faz o papel de centralizar a operação. Eles possuem OSs específicos para gerenciar tarefas comuns em uma rede – este modelo é conhecido como arquitetura cliente-servidor (dos clientes vão para a rede, e a rede busca no server).

Os computadores de servidores precisam de mais potência, de RAM, processador, etc. Ainda pode se ter um datacenter com vários servidores – 1 para arquivo, 1 para web, 1 para aplicações web, etc.

* **Criando máquinas virtuais**

Vale lembrar que todo o processo é semelhante entre Debian e Ubuntu, já o Fedora é diferente. Criar a máquina no Virtual Box é a mesma coisa, mas é preciso mudar a rede, uma com rede interna e uma segunda em modo bridge, para que a máquina física converse com a virtual).

* **Particionamento LVM**

É uma versão mais atual, LVM significa Gerenciador de Volume Lógico, e é um método para alocar espaços de HD em volumes lógicos que podem ser facilmente redimensionados, ao contrário das partições.

Você pode ter vários HDs e colocar todos em um mesmo volume. De qualquer forma, o LVM não mexe no boot tradicional, que ainda sai de algum destes HDs. O volume lógico usa dois pontos de montagem, o /home e o /, e ainda posso deixar alocado um espaço livre. As novas versões todas usam LVM.

* **Aula extra – IP**

IP significa Internet Protocol, ou Protocolo de Internet.

Atualmente há duas versões em uso, o IPv4 e o IPv6, sendo o IPv4 o predominante (com um endereçamento composto por um número de 32 bits, ou seja, 4 bilhões de combinações) mas já está esgotando as combinações e por uso, seu sucessor é o IPv6 (composto por um número de 126 bits, ou seja, 340 undecilhões de endereços).

O IPv4 éc omposto por 4 números, divididos por pontos (ex: 192.168.1.101) e dividido por cinco classes nomeadas de A a E.

- Classe A: o primeiro octeto indica a numeração de rede (1 a 126 – o 127 é usado para teste, também chamado de loopback), é utilizado em grandes corporações, companhias telefônicas e de TV, etc.

- Classe B: os dois primeiros octetos indicam rede, os demais indicam computadores (de 128 a 191.255)

- Classe C: os três primeiros octetos indicam rede e o último, o computador (para redes locais de pequenas e médias empresas).

- Classe D: Reservada para hardwares que utilizam multicasting – transmissão de datagramas IP desde uma fonte para múltiplos destinatários de uma rede IP, como roteadores (de 224.0.0.0 até 239.255.255.0).

- Classe E: reservado para testes e pesquisas (de 240.0.0.0 até 254.255.255.0)

Máscara de rede: A máscara é para que você indique a classe: uma máscara de classe C, por exemplo é 255.255.255.0. Ou seja, indica que os três primeiros são rede e o último é o número de hosts. O dispositivo/host só se conecta com os demais que estão na mesma rede.

O host ainda pode receber outras informações, como o endereço do servidor DNS primário e secundário, e o endereço do gateway da rede.

Estático: A configuração estática do endereço de IP é realizada por hosts que precisam ser identificados pelos seus IPs e, por isso, não pode ter alteração, pois isso acarretaria problemas em maquinas de clientes de serviço e as funções destes PCs ou dispositivos (mais utilizado em servidores, roteadores, etc).

Gateway padrão: é o endereço IP do dispositivo de rede (roteador), que dá acesso a este host a outra rede (caso o host tenha acesso a internet, é necessário indicar o endereço IP do roteador que dará acesso a outra rede fora da rede local).

DNS primário: DNS da rede local, mesmo server que pode acumular serviço de domínio.

DNS secundário: responsável pela resolução de nomes em IPs da rede externa, no caso, internet. No secundário, o IP de uma saída DNS público pode ser utilizado com o endereço 8.8.8.8. Em caso de alteração, deve ser feito manualmente.

Dinâmico: Feito para facilitar a configuração IP em redes wifi e redes cabeadas de médias e grandes infraestruturas.

Em uma rede pequena é fácil inserir manualmente os IPs, mas em uma rede gigante é difícil, por isso a configuração correta do DHCP é de suma importância para que seja possível controlar e renovar IPs de uma maneira mais personalizada e que atenda as demandas da infraestrutura.

* **Definindo o hostname**

hostnamectl + set-hostname + novo nome do servidor.

* **Configuração de rede**

No Ubuntu o software de gerenciamento de rede é o netplan e no Fedora o Network Manager.

ip a – lista as placas de rede

No Ubuntu, para alterar os IPs você precisa ir até a pasta do software (/etc/netplan) e em seguida abrir o arquivo .yaml e editar (via Nano ou VI). No IP dos servidores, se houver mais de 1 eu separo por vírgula. Na placa enp0s8, o “true” indica que ela vai tentar buscar um IP na rede automaticamente. Depois tem que reiniciar a máquina ou usar o comando netplan apply.

Em nameservers, adresses é o DNS e search é o domínio.

Se eu quisesse que a primeira placa, a enp0s3 recebesse um IP automaticamente, é só apagar as infos e colocar o parâmetro igual a segunda placa.

No Fedora, usa-se o nmcli connection modify [placa de rede] IPv4 [novo IP]. Em seguida, você precisa desligar a placa de rede, usando nmcli connection down [placa], e em seguida reativar, trocando o “down” por “up”.

Se eu quiser modificar o gateway, é só trocar “addresses” por “gateway”, a mesma coisa se for DNS, etc. Já para procurar automaticamente usa ipv4.method auto.

Vale lembrar que se eu quiser checar a coneção, só usar ping + um site, se houver retorno, você está online.

* **Atualizando o servidor**

Da mesma forma que atualiza o Gnome ou qualquer ambiente gráfico: apt update/upgrade em Ubunto e Debian, e dnf update em Fedora.

* **Acesso remoto ao Ubuntu server**

A ferramenta mais popular é o open ssh. Para verificar se o serviço está em execução, usa-se o comando “systemctl status ssh”, se não estiver eu ativo ele como “systemctl start ssh”. Para o serviço ficar sempre ativo, usa-se o “systemctl enable ssh”.

Em seguida, é necessário editar o arquivo de configuração, pelo “Nano ou VI /etc/ssh/sshd\_config” para que você possa usar o acesso remoto via usuário root. Com o arquivo aberto, procure o #PermitRootLogin (#indica que a opção está comentada), e tirar o # e trocar a opção “prohibit-password” por “yes”, e em seguida, reiniciar o serviço, usando o comando “systemctl restart ssh”.

Para fazer o acesso remoto precisa ter o IP da máquina (ip a).

Para acessar em outra máquina Linux, logar com o root nesta máquina e usar o comando “ssh root@[endereço de IP]. Já no Windows faz a mesma coisa usando o arquivo “putty”.

* **Acesso remoto ao Fedora server**

(Por padrão, os novos updates já vem com o serviço instalado e em execução).

systemctl status sshd

systemctl start sshd

Para a execução permanente: nano /etc/ssh/sshd\_config e faz o mesmo procedimento do Ubuntu, seguido de systemctl restart sshd

Firewall-cmd --list-all checa se o acesso remoto está liberado (em serviços, tem que estar liberado, entre outras coisas, o ssh, se não estiver: firewall-cmd –add-service = ssh --permanent

Em seguida: firewall-cmd –reload

**MÓDULO 4**

* **Servidores arquivos/Adicionando um novo disco**

Arquivos compartilhados com todos, idealmente, devem ficar em um local centralizado, ou ainda, colocar permissões. Desta forma, preciso criar um server de arquivos.

Utilizando o serviço samba, consigo criar um server Linux para uso dos PCs Windows da rede para compartilhar serviços como arquivos, diretórios, impressão, etc através do protoco smb.

É importante que tenha um disco separa para compartilhar arquivos, pois é mais fácil para fazer backup e caso eu perca o disco, perde só o OS, já que os arquivos estão em outro espaço, e também deixa o sistema mais rápido.

Tem que criar o disco no VB, particionar (fdisk -l), formatar (mkfs.ext4 /dev/sdb), montar (mkdir/disco2 > mount -t ext4 /dev/sdb /disco2> e configurar para manter montado (nano /etc/fstab> /dev/sdb/disco2 ext4 defaults 0 0 > salvar, sair e checar com df -h.

* **Instalando o Samba**

Usa-se o Samba para que usuários possam acessar a pasta compartilhada.

Tem que criar uma pasta compartilhada no HD e dar total permissão (chmod 777 /disco2/nome da pasta).

apt -y install samba (instalar)

systemctl status smbd (checa se está ativo)

Em seguida, precisa compartilhar a pasta configurando o arquivo do Samba, usando: nano /etc/samba/smb.conf (“map to guest = bad user não pode estar comentado. Este comando impede que pessoas não autorizadas acessem).

Na última linha do documento adicionar:

[nome da pasta]

-

path = /disco2/nome da pasta (caminho)

writable = yes (usuários poderão gravar)

guest ok = yes (convidados poderão acessar?)

guest only = yes (só convidados? Desta forma não pedirá user e senha)

create mode = 0777 (permissão para criar)

directory mode = 0777 (permissão total para diretórios)

Grava, salva e sai

Para ativar: systemctl start smbd e em seguida system ctl status smbd, e para manter ativo: systemctl enable smbd

* **Máquina cliente (Windows)**

Para máquina virtual, o ideal é Windows 7 ou 8.1, o 10 fica muito pesado.

Tem que deixar o Adaptador de rede em “Rede Interna”.

Já no Windows, ir nas configurações de rede ipv4, colocar o IP, fazer o teste se há comunicação entre cliente e servidor, via CMD, usando ping + ip do servidor, se houver resposta, tem comunicação, então as pastas compartilhadas pelo servidor devem estar disponíveis.

Voce ainda pode ir em Meu Computador>Rede (botão direito)>Mapear unidade de Rede>IP, desta forma, você não precisa digitar o endereço de IP toda hora para acessar as pastas.

* **Compartilhando uma pasta restrita**

Lembrando como se cria um usuário: Useradd + nome, e pode ser incrementado com -c “nome da máquina” -G (nome do grupo), etc. O -M não cria pasta pessoal dentro do home, já que para um servidor, o usuário não precisa acessar a pasta compartilhada. Da mesma forma, também não precisa de senha, pois ele não irá realizar logon no servidor.

Entretanto, ele precisa de senha para acessar o samba: smbpasswd -a [user] (o -a) significa “add user” dentro das opções do smbpasswd (lembrando que se houver dúvida, é só usar o –help)

Chgrp muda o grupo. Ex: chgrp [grupo] [caminho], na sequência, você precisa alterar as permissões, neste caso, da pasta restrita

Chmod 770 /disco2/restrita (lembrando que 7 – dono, 7 – grupo, 0 – geral)

Em seguida, no samba, é necessário compartilhar esta pasta. Lembrando:   
nano /etc/samba/smb.config e faz o mesmo processo da pasta pública. Lembrando:

[restrita]  
path = /disco2/restrita (caminho)  
writable = yes (gravável?)  
guest ok = no (pode convidados?)  
create mode = 0770 (segue o chmod, importante lembrar que começa com 0)   
directory mode = 0770 (segue o chmod, importante lembrar que começa com 0)  
valid users = @GRP\_VENDAS (quem pode? Detalhe para o @ antes do nome do grupo)

Lembrando que sempre que mexer no samba, precisa restartar e depois checar se está ativo:

systemctl restart smbd  
systemctl status smbd

* **Acessando compartilhamento restrito**

Só ir no cliente e acessar pelo ip, você precisará colocar o usuário e senha.

* **Instalando o Samba no Fedora**

É a mesma coisa, mas ao invés de usar “smbd”, usa-se “smb”, além do Fedora contar com algumas camadas de proteção que você precisa lidar.

Firewall: checar se está ativo com firewall-cmd --list-all, se o samba não aparecer no resultado, é necessário adicionar o serviço: firewall-cmd --add-service=samba --permanent

Em seguida, reload no firewall: firewall-cmd --reload e depois checa novamente.

A outra camada é o SELinux, que trata de acesso às pastas. Você pode configurá-lo ou desabilitá-lo.

Para desabilitar: geten force (para ver se está ativo), em seguida seten force: desativa ou ativa, a opção 1 ativa, a 0 desativa. Ex: “seten force 0”.

Em seguida, precisa deixar isso permanente: nano /etc/selinux/config, onde está “SELINUX” você coloca “selinux=disabled”

Depois é necessário voltar às configs do samba (nano /etc/samba/smb.conf). Em [global], coloca o nome do workgroup igual ao workgroup do cliente, e acrescentar no final da lista “map to guest = Bad User”.

* **Configuração do SELinux**

(o SELinux também está no redhat e outras aplicações)

O SELinux define controle de acesso para aplicações, processos e arquivos em um sistema, incluindo o Samba.

Utiliza políticas de segurança (conjunt de regras que dizem se o SELinux pode ser acessado).

Nano /etc/selinux/config – configurações do selinux. Se eu deixar “enforcing” no “SELINUX=” ele fica ativo.

Para configurar o SELinux sem precisar desativá-lo, de forma somente que permita o samba a acessar pastas, faça: setsebool -P samba\_enable\_home\_dirs on

Em seguida, precisa tirar qualquer máscara do SELinux que tenha aplicado à pasta em questão: restorecon -R /[pasta] (o R é usado para ser recursivo).

Em seguida, tem de aplicar nova máscara, dizendo que essa pasta pode ser acessada pelo samba: chcon -R -t samba-share-t /[pasta]/

E por final, reiniciar o samba: systemctl restart smdb

**MÓDULO 5 – Servidor Web/DNS (Ubuntu)**

* **O que é um servidor web ou Web server**

Refere-se ao hardware ou software, ou ambos trabalhando juntos.

No hardware: É um computador que armazena os arquivos que compõem os sites (docs HTML, img, CSS, arquivos JS), e os entregam para o dispositivo final.

Está conectado à internet e pode ser acessado via nome do domínio (DNS), ex: google.com

No Software: O web server inclui diversos componentes que controlam como os usuários acessam os arquivos hospedados – no mínimo um servidor HTTP (software que compreende URLs e protocolos HTTP que o navegador utiliza para visualizar páginas web.

Você acessa o webserver via browser, ele faz uma requisição HTTP e o server retorna a requisição também em HTTP.

Os dois principais softwares para criação de webserver são Apache (Linux) e IIS (Microsoft).

* **Instalando o Apache**

apt install apache2 -y

teste o apache: systemctl status apache2 e depois, no cliente, digite no navegador o IP do servidor, se abrir a página do Apache, está tudo certo. Lá ela ainda indica onde ficará hospedados os arquivos que compõe o site (/var/www/html).

Lembrando:

wget + url é usado para fazer download

unzip + nome do arquivo descompacta o arquivo zipado.

O webserver também pode ser usado para instalar webapps dentro de uma rede local (intranet).

* **O que é um servidor DNS**

O server DNS é a “lista telefônica da intranet”. O DNS que encontra o IP do domínio que você busca no browser (no caso da intranet, busca o servidor da empresa). Ou seja, ele converte o nome de domínio em IP.

Para saber o seu server DNS é só digitar ipconfig /all (no Windows) ou cat /etc/resolv.config (Linux) e checar o DNS server.

Em empresas tem dois servidores, geralmente, um para consulta interna e outra para sites externos.

* **Criando um servidor DNS**

Tem que converter o server em server DNS, para isso, é preciso criar uma espécie de banco de dados com os nomes e IPs que o servidor irá resolver. Em seguida, você, grosso modo, “avisa a placa de rede” que ele é um servidor DNS.

Tem que ir até /etc/netplan, em seguida editar o arquivo que está na pasta: nameserver são os servidores DNS (no caso, vai colocar o mesmo IP do servidor interno e também o servidor para internet, que é o 8.8.8.8, separados por vírgula. Em seguida, netplan apply.

Depois, precisa instalar uma aplicação para server DNS, o bind9 e a ferramenta bind9utils (vale ressaltar que o app chama bind9, mas o nome do serviço é “named”).

Para criar o banco de dados interno: nano /etc/bind/named.conf (lá ele chama os arquivos de configuração dele, logo abaixo eu preciso indicar o meu, da rede interna (include “/etc/bind/named.conf.rede-interna”;

Também preciso configurar o arquivo named.conf.options. Os servidores DNS externo (estrangeiro) está comentado, preciso ativar ele e trocar o servidor DNS para 8.8.8.8, desabilitar os requisitos de IPv6 (trocar “any” por “none”).

E preciso dizer que o server vai atender requisições vindo da rede (entre o “dnssec” e o “listen-on-v6”:

allow-query { Localhost; 192.168.100.0/24; }

allow-transfer { localhost; };

recursion yes

Em named.conf foi citado um arquivo chamado “rede-interna”,lá será criado as zonas de DNS. Agora é a hora de cria-lo: nano /etc/bind/named.conf.rede-interna. O arquivo estará em branco, desta forma preciso configurá-lo:

zone “empresa.local” IN { (domínio que vai responder)

type master (tipo principal)

file “/etc/bind/empresa.local.db”; (será o arquivo de configuração onde estará os ponteiros que eu vou criar para a zona.

Agora é criar o arquivo com os ponteiros. Pontos de destaque:

$TTL 24 (time to leave – essa configuração total possui 24h)

Em seguida vem a configuração de tempo para cada atualização, que é um cabeçalho padrão.

Também é importante respeitar o serial (para quando houver um servidor DNS secundário).

Em seguida vem quem é o server DNS (o IP dele, no caso), e abaixo os ponteiros (o IP de cada emento, como server1, server2, www, email, intranet, cliente).

O próximo passo é a pré configuração no named para ele responder IPv4 (nano /etc/default/named) e adicionar OPTIONS=”-u bind -4”.

Antes de startar o bind, precisa checar um arquivo, o resolv.config, onde estão especificados os servidores DNS: nameserver ver os dois IPs, o interno e o externo. Em seguida, restar no bind (systemctl restart named).

Para finalizar, na máquina cliente é preciso configurar para que ela saiba quem é o server DNS, indo até as configurações de rede IPv4 e em seguida preencher os campos de server DNS (primário interno, alternativo externo).

* **Virtualhost (mais de um site por servidor)**

Lembrando: para limpar todo o conteúdo de um diretório: rm -r, para voltar uma pasta: cd .., para descompactar: unzip [nome do arquivo].zip.

Ao fazer alterações, o /etc/resolv.conf (responsável por configurar os servidores DNS) pode ser atualizado pelo sistema. É preciso arrumar (via nano) e corrigir os 2 nameservers, com o IP do servidor (interno), e o ip externo (8.8.8.8).

Isso acontece por eu estar utilizando 2 placas de rede que está recebendo IP via DHCP (ele pega o DNS do DHCP e altera automático).

Para evitar isso, é necessário trocar o atributo deste arquivo: chattr (change atribute) -f (forçar) +i (não permite que o arquivo seja mexido).

(se precisar alterar futuramente, troco o + por -)

Em seguiuda, precisa criar uma nova zona de requisição primária (em named.conf.rede-interna)

zone “minhaempresa.com.br” IN {

type master;

file “/etc/bind/minhaempresa.com.br.db”; #(aqui é o arquivo com os ponteiros, preciso criar e configurar).

};

Em seguida, resetar e checar o status de named.

Na sequencia, direcionar as requisições, ou seja, criar o virtualhost, que é, basicamente, um arquivo chamado virtual.host.conf dentro de /etc/apache2/sites-available e configurá-lo.

Pontos de atenção: ficam os dados sites no mesmo arquivo. Nestas configurações vão o documento raiz (local), onde os requisitos no servidor, designado em “ServerName”, vão. Em seguida dois parâmetros para logs de acesso.

Próximo passo é atualizar os virtualhosts: a2ensite virtual.host, e reload e restart no apache2.

**MÓDULO 6 – Servidor Web/DNS (Fedora)**

* **Servidor Web (httpd) – Fedora**

1º passo é instalar o Apache, que no Fedora chama HTTPD (dnf -y install httpd), em seguida o systemctl start/status httpd para deixar, ou ver se está, tudo ok. Em seguida, systemctl “enable” httpd para ser ativado já no boot.

No Fedora tem o firewall, então precisa configurar também (firewall-cmd --list-all para checar as permissões). Firewall-cmd –add-service=http -permanent e depois dar reload (firewall-cmd --reload)

No cliente: checar se há comunicação com o servidor (ping + IP) e depois fazer um double check no browser

* **Servidor DNS – Fedora**

Será criado 2 domínios. Nota: domínios .local são reservados no Fedora, portanto, não funcionam corretamente.

nmcli connection modify enp0s3 ipv4.dns "192.168.100.11 8.8.8.8"

em seguida, resolver o resolv.conf nano /etc/resolv.conf

resolvectl também configura os servidores DNS. Ex: resolvectl dns enp0s3 192.168.100.11

Em seguida, instalar o bind e bind-utils: dnf -y install bind bind-utils

Próximo passo é configurar o arquivo principal do bind (named) – nano /etc/named.conf

Na porta 53, responsável pelas requisições DNS, deve acrescentar any; para que ele ouça, além de ele mesmo, qualquer requisição de qualquer faixa de IP.

Abaixo, deixar IPv6 como comentado, e em allow-query, que é para permitir buscas de requisitos, colocar any; ou o IP da rede

Não precisa incluir um arquivo de configuração, igual fizemos no Ubuntu, você pode colocar no fim deste arquivo mesmo:

zone "fedora.home" IN {

type master;

file "/var/named/fedora.home.lan";

allow-update { none; };

};

zone "minhacasa.com.br" IN {

type master;

file "/var/named/minhacasa.com.br.lan";

allow-update { none; };

};

Em seguida, liberar as consultas pelo ipv4 no server DNS em nano /etc/sysconfig/named e incluir no final OPTIONS="-4".

Agora é criar os dois arquivos de configuração para os respectivos domínios:

nano /var/named/fedora.home.lan

$TTL 24h

@ IN SOA dns.fedora.home. root.fedora.home. (

1 ; Serial

12h ; Refresh

15m ; Retry

3w ; Expire

2h ; Mininum TTL

)

@ IN NS dns.fedora.home.

dns IN A 192.168.100.11

server2 IN A 192.168.100.11

www IN A 192.168.100.11

nano /var/named/minhacasa.com.br.lan

$TTL 24h

@ IN SOA dns.minhacasa.com.br. root.minhacasa.com.br. (

1 ; Serial

12h ; Refresh

15m ; Retry

3w ; Expire

2h ; Mininum TTL

)

@ IN NS dns.minhacasa.com.br.

dns IN A 192.168.100.11

server2 IN A 192.168.100.11

www IN A 192.168.100.11

restartar o named e checar (systemctl restart/status named)

Liberar o firewall: firewall-cmd --add-service=dns --permanent

Dar reload no firewall: firewall-cmd –reload

Testar (ping)

* **Virtualhost no Fedora**