Interface gráfica do usuário

Descrição gerada automaticamente com confiança média

1ºPasso: Escolher a linguagem do sistema que nesse caso e english

Texto

Descrição gerada automaticamente

2º Passo: Diz que tem uma nova atualização dessa imagem que baixei:

Selecionamos “Update to the new installer”

Texto

Descrição gerada automaticamente

3º Passo: Escolher o padrão do teclado

Texto

Descrição gerada automaticamente

4º Passo: Selecionar Ubuntu Server

Texto

Descrição gerada automaticamente

5º Passo: Nesse caso a interface “enp0s3” e a interface NAT, e a placa de rede que vem pegar os dados que vem da rede externa

A segunda interface de rede “enp0s8” e a placa que preciso configurar, que e a placa de rede interna

Texto

Descrição gerada automaticamente

Clico em “Edit IPv4” em seguida selecionamos manual

Texto

Descrição gerada automaticamente com confiança média

**Subnet:** Endereço da rede 192.168.1.0/24

**Address:** Endereço do servidor 192.168.1.10

**Gateway** e o endereço ip do roteador da sua rede, e uma forma de habilitar o trafego de informações de uma rede interna para uma rede externa como a internet

Gateway: 10.0.2.15

enp0s8 e a placa da rede interna

**Name servers** são o DNS

Texto

Descrição gerada automaticamente

Em seguida clico em “Done”

6ºPasso: Nesse caso não vou usar servidor proxy, vou selecionar “Done”

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

7 º Passo: Aqui e um espelho aonde tem os arquivos do ubuntu, somente seleciona “Done”

Texto

Descrição gerada automaticamente

8º Passo: Configuração sobre o particionamento do hd

Nesse caso não irei utilizar o LVM que e um gerenciador de volumes lógicos.

Em casos reais, o LVM permite o reparticionamento de partições, o volume pode ser redimensionado a qualquer momento por apenas uma linha de comando.

Utilização de discos paralelos, podemos ter uma unidade lógica centralizada dos discos físicos

Texto

Descrição gerada automaticamente

9º Passo: Essa tela abaixo só mostra como foi feito as configurações de armazenamento do hd

Texto

Descrição gerada automaticamente

10º Passo:

Clico em continue

Texto

Descrição gerada automaticamente

11º Passo: Aqui escolhemos o perfil do servidor, no caso: usuários e as senhas

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

12º Passo: Selecionar “Install OpenSSH Server”, pois permite acessar o servidor de forma remota

Texto

Descrição gerada automaticamente

13º Passo: Essa tela abaixo, mostra um conjunto de serviços que vc pode instalar no seu servidor, mas a gente n vai instalar nada

Texto

Descrição gerada automaticamente

14 º Passo: Clicar em Reboot Now para reiniciar

Texto

Descrição gerada automaticamente

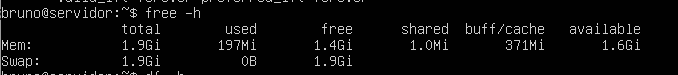
**Verificando as configurações de rede**

bruno@brunoserver:~#**ip a**

**Verificando a memória que está sendo utilizada**

bruno@brunoserver:~#**free -h**

**Nesse caso está sendo utilizado 197Mi**

****

**Verificando espaço no HD que está sendo utilizado**

bruno@brunoserver:~#**df -h**

**Nesse caso está sendo utilizado 4.9G**

**Texto

Descrição gerada automaticamente**

**Fazendo a configuração de rede**

Texto

Descrição gerada automaticamente

**Acessando esse arquivo, podemos alterar ele**

Texto

Descrição gerada automaticamente

**Acesso Remoto com os protocolos SSH e SFTP**

**SSH** = SECURE SHELL = Terminal Seguro

Diagrama

Descrição gerada automaticamente

No servidor e necessário está instalado o ssh server(servidor de protocolo ssh)

No cliente onde iremos acessar o servidor de forma remota e necessário ter o protocolo ssh cliente

No virtual box na minha máquina virtual ubuntu server, e necessário as configurações de rede do servidor ubuntu está em “Placa em modo Bridge”, para consegui se comunicar via ssh com a minha máquina física

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

* **Em seguida reinício a máquina**

bruno@servidor:~$**init 6**

* **Instalando o net-tools para permiter funcionar o comando ifconfig, netstart, rarp e etc**

root@servidor:/home/bruno# **sudo apt-get install net-tools**

* **Verificando o ip da minha máquina**

bruno@servidor:~$**ifconfig**

* **Comando para instalar o ssh servidor**

bruno@servidor:~$ **sudo su**

root@servidor:/home/bruno# **apt install openssh-server**

* **No prompt de comando do Windows digitar**

ssh bruno@ipdoservidor



* **Em seguida digito “yes” conforme imagem abaixo**

Placa azul com letras brancas

Descrição gerada automaticamente com confiança média

* **Em seguida digito a senha do meu Ubuntu servidor**

Texto

Descrição gerada automaticamente

Agora estou acessando remotamente meu servidor, ou seja, posso fazer a administração do meu servidor de forma remota

Observação: Na minha máquina cliente tem que ter o ssh cliente instalado

Caso a porta do servidor for diferente da porta 22, no comando abaixo eu coloco a porta correta no lugar de 22:

Utiliza o seguinte comando ssh -p 22 bruno@192.168.0.36

**Protocolo FTP = File Transfer Protocol** – Protocolo de Transferencia de arquivos. Atualmente o FTP está defasado, principalmente quando focamos em segurança

**Protocolo SFTP = SSH File Transfer Protocol ou Secure File Tranfer Protocol -** Protocolo de transferência de arquivo mais seguro

Quando você instala o SSH, você também instala o SFTP

* **1º Passo: Fazer a conexão SFTP utilizando o prompt de comando da minha máquina física.**

Esse comando **-oPort=22** significa que estou colocando uma porta personalizada que no caso e a porta 22

Comando **sftp -oPort=22 bruno@ipdoservidor**

Texto

Descrição gerada automaticamente

Vai pedir a senha, coloca a senha do servidor

Agora estou conectado no meu servidor via protocolo sftp

* **2º Passo: No sftp si eu colocar “l” na frente do comando, vai mostrar os arquivos do meu computador físico. Todo comando Linux que eu colocar “l” na frente, significa que vai ser executado na minha máquina local**

Exemplo:

Coloquei o comando:

sftp> **lls**

Esse “l” na frente do ls, vai permitir me mostrar os arquivos da pasta que estou atualmente no meu computador fisico

Texto

Descrição gerada automaticamente com confiança baixa

3º Passo: Si eu colocar esse comando abaixo, vai mostrar a pasta atual que estou no meu computador físico/máquina local

sftp> **lpwd**

Texto

Descrição gerada automaticamente

Nesse caso estou dentro da página usuário e dentro da pasta bruno

4º Passo: Si eu colocar o comando sem o “l”, esse comando vai ser executado no servidor ubuntu

sftp> **pwd**

Texto

Descrição gerada automaticamente

Exemplo:

Texto

Descrição gerada automaticamente

Entrei na pasta que se encontra na área de trabalho da minha máquina local chamada “deletar” utilizando o comando “lcd deletar” .

Em seguida utilizei o comando “lpwd” para verificar o caminho dessa pasta na minha máquina local.

Em seguida utilizei o comando “lls” para listar o conteúdo dessa página.

Tudo isso foi possível, pois coloquei o “l” na frente do comando Linux, e porque também estou utilizando o protocolo “sftp”

**5º Passo: Copiar um arquivo que encontra-se no servidor ubuntu via sftp e colar na minha máquina local,**

Nesse caso estou copiando da minha máquina servidor “ubuntu server” o arquivo “teste.txt” que se encontra no caminho /home/bruno/pasta-teste e na pasta “pasta-teste”.Como no prompt de comando do sftp na minha máquina local estou no caminho c:\users\bruno\desktop\deletar , ou seja, estou na pasta chamada “deletar”. Ao copiar esses arquivos do servidor ubuntu, automaticamente serão colados na pasta “deletar” da minha máquina local

sftp> get teste.txt



Essa imagem abaixo, mostra que o arquivo foi copiado



6º Passo: Quando eu quero copiar múltiplos arquivos do meu servidor para minha máquina local

Nesse caso vou copiar todos o arquivos da pasta que são “.txt”

sftp> **mget \*.txt**

**No caso foi copiado 3 arquivos como mostra a figura abaixo**

Texto

Descrição gerada automaticamente

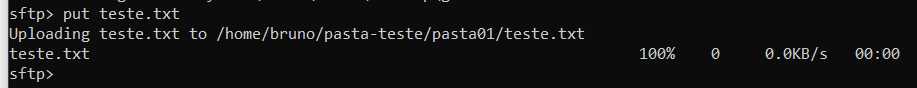
7º Passo: Agora irei, passar os arquivos da minha máquina local para o meu servidor ubuntu server

Nesse caso estou copiando da minha máquina local o arquivo teste.txt que esta localizado na pasta “gravação-teste” e colando e colando na pasta01 do servidor ubuntu

Esse e o caminho da minha pasta no ubuntu server /home/bruno/pasta-teste/pasta01

Esse e o caminho da minha pasta na máquina local c:\users\bruno\desktop\gravacao-teste

sftp> put teste.txt



Nesse momento copiei da minha máquina local e colei na minha máquina ubuntu server

8ºPasso: Copiando todos os arquivos que terminem com “.txt” da minha maquina local e colando na minha máquina Ubuntu Server

sftp >**mput \*.txt**

Texto

Descrição gerada automaticamente

O comando ip addr permite ver as interfaces de rede

Texto

Descrição gerada automaticamente

Essa imagem acima e da interface “1” que e chamada de “LOOPBACK”, que é um jeito da máquina falar com ela mesma

Texto

Descrição gerada automaticamente

A interface da imagem acima e a interface “2” de nome “enp0s3”

Temos que ver a linha “inet “

Que nesse caso e “inet 192.168.0.32”

Texto

Descrição gerada automaticamente

A interface da imagem acima e a interface “3” de nome “enp0s8”

Temos que ver a linha “inet “

Que nesse caso e “inet 192.168.1.10”

Criando uma configuração DHCP no UBUNTU SERVER

1º Passo: Deixar a primeira placa de rede no tipo “Nat”

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

2º Passo: Deixar a segunda placa de rede no tipo “Rede interna “ de nome “INFO4”

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

3º Passo: Verificar na tela as interfaces de rede

enp0s8 e a rede interna, 192.168.1.10 e o ip do meu servidor da rede interna

enp0s3 e o ip 10.0.2.15 do meu servidor em modo “Nat”

Texto

Descrição gerada automaticamente

4º Passo: Instalar no Ubuntu Server, o serviço DHCP

bruno@servidor:~$ sudo su

root@servidor:/home/bruno# apt install isc-dhcp-server

5ºPasso: Fazer a configuração do servidor, abrir o arquivo de configuração

root@servidor:/home/bruno# nano /etc/dhcp/dhcpd.conf

Texto

Descrição gerada automaticamente

No arquivo de configuração tem “**default-lease-time 600;**” que significa que o servidor DHCP vai controlar o tempo de renovação de endereços IPs, esse “600” significa que o servidor vai verificar a 600 segundos si as estações ainda estão ativas, ou seja, si as estações que estão usando esses servidores dhcps estão ativas. Fica verificando si aquela estação ainda está utilizando o ip que ele disponibilizou para a estação. No caso 600 segundos e 10 minutos

No arquivo tem “**max-lease-time 7200**;” significa o tempo máximo que a estação pode usar um determinado endereço IP.

Exemplo: Imagine uma rede que você tem mais endereços IPs do que máquinas, os endereços ips das estações raramente vão precisar mudar, mas no caso de uma rede congestionada esse “**max-lease-time 7200**;” se torna interessante.

Aonde existe escassez de endereços IPs como por exemplo em um provedor de acesso, aonde sempre existe mais clientes do que endereços IPs disponíveis, você trabalha contando que nem todos irão ficar conectados simultaneamente.

Então determinamos um tempo máximo que a máquina pode ficar com o IP, no caso 7200 segundos, podemos deixar como padrão o 7200

5ºPasso: Tiro o comentário da linha “authoritative;” para dizer que ele e o servidor DHCP oficial da minha rede

Texto

Descrição gerada automaticamente

6ºPasso: Fazer a minha subnet, nessa subnet eu vou fazer a configuração de como vai ser o meu servidor DHCP, como ele vai se comportar com as máquinas clientes

As máquinas na minha rede que usar esse servidor Ubuntu irão receber o ip entre o range “192.168.1.20” e “192.168.1.30”, as máquinas clientes receberão ip entre essa faixa de endereços ip

A linha “option domain-name-servers 8.8.8.8;” e o DNS que as máquinas clientes vão receber, nesse caso vou colocar 8.8.8.8

A linha “#option domain-name 8.8.8.8;” vou deixar comentada por que não estou trabalhando com o servidor de domínio

A linha “option subnet-mask 255.255.255.0” e a minha máscara opcional

A linha “option routers 192.168.1.10;“ e o IP do servidor, ou seja, todos os roteadores, os gateways das máquinas clientes vão receber esse IP 192.168.1.10

A linha “option broadcast-address 192.168.1.255;” e o endereço de broadcast

A linha “default-lease-time 600;” vou deixar 600 mesmo e a linha “max-lease-time 7200;” vou deixar em 7200

Basicamente e essa configuração do meu servidor DHCP

Texto

Descrição gerada automaticamente

Se por acaso você quiser definir um servidor de arquivo ou um servidor de impressoras que tem um ip fixo, você faz da seguinte maneira

Assim como uma máquina cliente que solicita informação do servidor DHCP, e essa máquina tiver o endereço físico “08:00:07:26:c0:a5” eu posso dizer que essa máquina vai receber um ip fixo “192.168.1.25”.

Toda vez que essa máquina que tiver esse mac “08:00:07:26:c0:a5”, que ele solicita um endereço IP ele vai receber esse IP 192.168.1.25.Geralmente isso e utilizado em servidores de arquivos e servidores de impressoras e etc

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Como nesse caso eu não tenho configuração de impressora, vou deixar comentado

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

e vou salvar as modificações

7ºPasso: Vou verificar se meu arquivo de configuração foi salvo de forma corretamente

bruno@servidor:~$ sudo dhcpd -t

Texto

Descrição gerada automaticamente

Nesse caso deu um erro, vou verificar novamente o arquivo

Tive que fazer uma alteração no arquivo dhcpd.conf

Rodando novamente o comando para verificar se está correto as modificações

Texto

Descrição gerada automaticamente

8º Passo: Fazer a configuração no arquivo isc-dhcp-server, vamos definir qual placa de rede o dhcp vai utilizar

Nesse caso na minha interface ipv4 vai utilizar a enp0s8, pois a outra placa enp0s3 e do meu modo NAT

bruno@servidor:~$ sudo nano /etc/default/isc-dhcp-server

Como o Servidor tem duas interfaces de rede e legal você estabelecer qual interface vai servir de servidor DHCP/serviço DHCP

Em seguida salvo o arquivo

Texto

Descrição gerada automaticamente

9º Passo: Estou reiniciando o serviço DHCP

bruno@servidor:~$ sudo systemctl restart isc-dhcp-server



10° Passo: Verificar se o serviço DHCP está funcionando

Nesse caso o serviço está funcionando

bruno@servidor:~$ sudo systemctl status isc-dhcp-server

Texto

Descrição gerada automaticamente

Agora meu servidor está configurado

**Configurando o Samba para compartilhamento de pastas/arquivos**

**Protocolo SMB – SERVER MESSAGE BLOCK**

Diagrama

Descrição gerada automaticamente

A ferramenta Samba tem um conjunto de programas, um desses programas implementa o SMB, ou seja, podemos fazer esse compartilhamento de arquivos e pastas utilizando a ferramenta samba.

No Samba podemos fazer o compartilhamento de arquivos e pasta entre sistemas operacionais diferentes (Windows, Linux, MacOs)

O Samba permite ter serviços de diretórios, autenticação de redes

No meu Ubuntu Server que tem o IP 192.168.1.10, vou instalar a ferramenta Samba no Servidor em seguida vou criar uma pasta e compartilhar ela

Vai ter um computador cliente1 que vai está na mesma rede, que vai ser a rede interna INFO4, sendo possível visualizar essa pasta compartilhada no servidor no computador do cliente1

1º Ir no Virtual Box e ver as configurações de rede do Servidor Ubuntu

O adaptador 1 de rede vou deixar como NAT

O adaptador 2 vou deixar como Rede Interna, e o nome da minha rede interna vai ser INFO4

Na minha máquina cliente, eu deixo ela como:

O adaptador 1 de rede da minha máquina cliente vai ser como Rede Interna, e o nome da minha rede interna vai ser INFO4.

Tanto meu servidor Ubuntu, como minha máquina cliente1 vão está na mesma rede chama INFO4

2ºPasso: Instalar a ferramenta Samba no Servidor Ubuntu

bruno@servidor:~$ sudo apt install samba

3º Passo: Fazer a configuração do servidor, essa configuração pode ser feita utilizando algumas ferramentas como por exemplo:

SWAT (Samba Web Administration Tool) que uma ferramenta que permite fazer essa administração de forma mais amigável para o usuário

Podemos fazer a configuração no próprio arquivo de configuração, que esse arquivo abaixo:

Por segurança iremos fazer um arquivo de backup do arquivo “smb.conf” que e o arquivo de configuração

bruno@servidor:~$ sudo cp /etc/samba/smb.conf /etc/samba/smb.conf.bkp



Agora iremos abrir o arquivo de configuração do samba.

bruno@servidor:~$ sudo nano /etc/samba/smb.conf

O arquivo de configuração e dividido em varias seções:

A primeira seção “Global” por exemplo:

Apresenta os parâmetros que serão aplicados no servidor como um todo. Por exemplo: Domínio, identificação da máquina, níveis de autenticação. Nessa seção global iremos alterar dois parâmetros

Primeiro parâmetro será:

O parâmetro está dessa maneira no arquivo de configuração do samba **workgroup = WORKGROUP** , esse parâmetro permite identificar o domínio que a máquina pertence, ou seja, o nome do grupo de trabalho. Nesse caso iremos colocar “lab”:

workgroup = lab

Texto

Descrição gerada automaticamente

Abaixo de workgroup = lab ,vou definir um parâmetro security do tipo “user” usuário. Na verdade o “security” vai definir a politica de acesso, de segurança e de autenticação dos compartilhamentos. O padrão “user” usuário, então o controle de acesso e baseado em usuário, nesse modo você deve cadastrar o login e a senha no servidor na hora que for compartilhar a pasta, serviço, autenticação e etc

security = user

Texto

Descrição gerada automaticamente

O segundo parâmetro será:

O parâmetro está dessa maneira no arquivo de configuração do samba **server string = %h server (Samba, Ubuntu).** O **server string** define basicamente um comentário que aparecerá no nome do servidor.

A seção global ficará da seguinte maneira:

Texto

Descrição gerada automaticamente

Nota-se que esse arquivo tem várias seções, e um arquivo que você pode estabelecer varias configurações para diferentes tipos de serviços. Por exemplo: domínio

Essa seção de “homes”, quando você define e cria parâmetros aqui, você define acessos a diretórios particulares de usuários

Texto

Descrição gerada automaticamente

A seção “netlogon” e utilizada para ativar a execução de scripts quando um usuário efetua logon em um domínio específico. Por exemplo; Usuario fez login de uma máquina e um domínio específico e executado algum script como por exemplo “script de bloquear papel de parede” quando o usuário não pode alterar papel de parede

Texto

Descrição gerada automaticamente

A seção “profile” define o perfil do usuário quando o samba e configurado

Texto

Descrição gerada automaticamente

A seção “printers” opção que permite que os clientes possam se conectar e utilizar qualquer impressora especificada e conectada na máquina local.

Fundo preto com letras brancas

Descrição gerada automaticamente

Pode selecionar as configurações da impressora na máquina local

Texto

Descrição gerada automaticamente

4º Passo: Fazendo a configuração de um compartilhamento de pasta

Vai ser uma seção/”pasta publica”

Essa linha “comment” e basicamente a descrição do compartilhamento que estou criando

Exemplo:

comment = pasta publica que todos tem acesso a leitura e escrita

A linha “path” diz o caminho que está a pasta

Exemplo:

Path = /home/bruno/publica

A configuração “guest ok = yes” simplesmente habilita o acesso público ao compartilhamento sem senha,ou seja, na hora que a maquina cliente tentar acessar essa pasta pública não será pedido nenhuma senha de usuário porque o convidado pode acessar

O “browseable = yes” basicamente especifica que o compartilhamento será visualizado na lista de itens compartilhados, ou seja, se esse browseable estiver da seguinte maneira “browseable = no” e porque vai ser oculto o compartilhamento

Quando o “browseable estiver dessa forma “browseable = yes” e porque ele estará disponível, estará visível nos arquivos compartilhados

O “writable = yes” significa que o compartilhamento ficará disponível para escrita e leitura na pasta, poderia colocar da seguinte maneira “writable = no” e seria somente leitura

Texto

Descrição gerada automaticamente

Salva as configurações

5° Passo: Criar a pasta chamada “publica”

Como essa pasta que criei no arquivo samba “/home/bruno/publica” não existe

Eu terei que criar uma pasta chamada “publica” no caminho “/home/bruno”

bruno@servidor:~$ mkdir publica



6° Alterar as permissões dessa pasta chamada “publica”

bruno@servidor:~$ chmod 777 publica

Texto

Descrição gerada automaticamente

Coloquei todas as permissões da pasta chamada “publica”

Observação:

Quando eu criei a pasta ela tinha permissão “drwxrwxr-x”, ao alterar as permissões da pasta publica ela ficou com a permissão “drwxrwxrwx”, ou seja, ficou com todas as permissões

Uma imagem contendo Tabela

Descrição gerada automaticamente

7°Passo: Estou acessando a pasta chamada “publica” e vendo o caminho aonde ela foi criada

Texto

Descrição gerada automaticamente

8° Passo: Vou criar alguns arquivos de exemplo. No caso estou criando 3 arquivos de exemplo:

bruno@servidor:~/publica$ touch arq{1..3}.txt

Texto

Descrição gerada automaticamente

9° Passo: Alterando as permissões dos 3 arquivos que criei na etapa anterior

bruno@servidor:~/publica$ chmod 777 arq{1..3}.txt

Texto

Descrição gerada automaticamente

Nas 3 primeiras linhas em amarelo os arquivos tinham algumas permissões,

Agora na última 3 linhas, depois de executar o comado chmod as permissões foram alteradas.

10° Passo: Criando um usuário no servidor Samba, porque algumas máquinas Windows não utilizam o mesmo tipo de criptografia do que no Linux para armazenamento de senha, principalmente se for uma pasta que precise de autenticação de algum usuário no servidor.

Existe a diferença de algoritmo de criptografia entre Linux e Windows, devido a isso e necessário que a gente crie um usuário dentro do servidor Samba

bruno@servidor:~/publica$ sudo su

bruno@servidor:/home/bruno/publica# smbpasswd -a bruno

Texto

Descrição gerada automaticamente

Podemos utilizar algumas opções no smbpwd

Por exemplo:

X para excluir usuário do Samba

D para bloquear usuário do Samba, mas não remove usuário do Samba

E para desbloquear usuário do Samba, mas não remove usuário do Samba

Fazendo um teste para verificar se está tudo de acordo

bruno@servidor:/home/bruno/publica# testparm

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

11° Passo: Reiniciando o Samba

bruno@servidor:~/publica$ sudo su

bruno@servidor:/home/bruno/publica# service smbd restart

12° Acessando a pasta chamada “Publica” no computador do cliente.

A máquina do cliente tem que está na mesma rede do servidor

Podemos colocar no explorador de arquivo smb//192.168.1.10/ que e o IP do nosso servidor, vai aparecer a pasta pública

Principais parâmetros de compartilhamento de pastas e arquivos no SAMBA

No nosso caso:

Servidor Ubuntu Server tem dois adaptadores de redes:

Adaptador de rede 1: NAT enp0s3 : 10.0.2.15

Adaptador de rede 2: Rede Interna enp0s8 : 192.168.1.10

Nome: INFO4

Máquina cliente: Windows 7

Adaptador de rede 1: Rede Interna

Nome: INFO4

1° Passo: Ir no Servidor Ubuntu, e criar as pastas que vamos compartilhar. Nesse caso estou criando duas pastas

bruno@servidor:~$ mkdir aula-samba textos

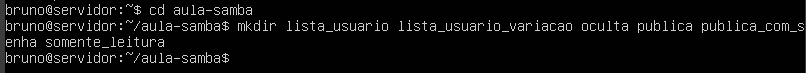
Texto branco sobre fundo preto

Descrição gerada automaticamente

2° Passo: Entrei na Pasta chamada “aula-samba” e criei 6 pastas dentro da pasta “aula-samba”

bruno@servidor:~$ cd aula-samba

bruno@servidor:~/aula-samba$ mkdir lista\_usuario lista\_usuario\_variacao oculta publica publica\_com\_senha somente\_leitura



Confirmando a criação das pastas

Texto

Descrição gerada automaticamente

3°Passo: Criando arquivos dentro das pastas que criei anteriormente

bruno@servidor:~/aula-samba$ cd lista\_usuario

bruno@servidor:~/aula-samba/lista\_usuario$ touch arq\_usuario.txt

bruno@servidor:~/aula-samba/lista\_usuario$ cd ..

bruno@servidor:~/aula-samba$ cd lista\_usuario\_variacao

bruno@servidor:~/aula-samba/lista\_usuario\_variacao $ touch arq\_usuario\_variacao.txt

bruno@servidor:~/aula-samba/lista\_usuario\_variacao $ cd ..

bruno@servidor:~/aula-samba$ cd oculta

bruno@servidor:~/aula-samba/ oculta$ touch arq\_oculto.txt

bruno@servidor:~/aula-samba/oculta$ cd ..

bruno@servidor:~/aula-samba$ cd publica

bruno@servidor:~/aula-samba/publica$ touch arq\_publico.txt

bruno@servidor:~/aula-samba/ publica$ cd ..

bruno@servidor:~/aula-samba$ cd publica\_com\_senha

bruno@servidor:~/aula-samba/ publica\_com\_senha$ touch arq\_publico\_com\_senha.txt

bruno@servidor:~/aula-samba/ publica\_com\_senha$ cd ..

bruno@servidor:~/aula-samba$ cd somente\_leitura

bruno@servidor:~/aula-samba/ somente\_leitura$ touch arq\_leitura\_somente.txt

bruno@servidor:~/aula-samba/ somente\_leitura$ cd ..



Texto

Descrição gerada automaticamente

Verificando a criação dos arquivos nas determinada pastas

Texto

Descrição gerada automaticamente

Criei as pastas com seguintes nomes, para facilitar o entendimento

Por exemplo: Na pasta chamada “publica” será possível compartilhar essa pasta, aonde todos os usuários podem ler e escrever dentro dela, qualquer pessoa pode acessar

Por exemplo: Na pasta chamada “publica\_com\_senha” todos podem ter o acesso contanto que você tenha a senha do usuário criada no Samba

Por exemplo: Na pasta chamada “somente\_leitura” todos tem acesso, mas não podem modificar os arquivos que estão na pasta, o acesso será apenas para leitura, não será possível realizar a modificação de arquivos

Por exemplo: Na pasta chamada “lista\_usuario”, somente alguns usuários podem acessar essa pasta

Por exemplo: Na pasta chamada “lista\_usuario\_variacao”, nesse caso vai ser um compartilhamento onde uma lista de usuários podem modificar os arquivos que estão nessa pasta e outros usuários podem apenas ler

4° Passo: Instalar a ferramenta “Samba”

bruno@servidor:~$ sudo apt install samba

5° Passo: Fazer a configuração no arquivo “smb.conf”. Nesse arquivo e aonde eu vou fazer as minhas configurações de compartilhamento de pastas e arquivos utilizando a ferramenta Samba

bruno@servidor:~$ sudo nano /etc/samba/smb.conf

workgroup = lab Grupo de trabalho

security =user Segurança a nível de usuário

Texto

Descrição gerada automaticamente

Principais parâmetros de compartilhamento de pastas

* 1º Compartilhamento, será a pasta de nome “publica”. Essa pasta de nome “publica”, todos tem acesso a leitura e a escrita

O parâmetro “guest ok” significa que “convidados poderão acessar a essa pasta” no caso coloquei “guest ok = yes” , ou seja, qualquer usuário pode acessar essa pasta, não precisa de senha

O parâmetro “browseable” significa que se “está pasta estará oculta ou não”. No caso “browseable = yes” significa se ela vai está na lista disponível, ou seja, a pasta não vai está oculta, vai aparecer na parte de compartilhamento.

Si o “browseable” estiver “no” dessa maneira “browseable = no” a pasta vai ficaria oculta, você poderia acessar a pasta através do caminho completo da pasta

O “writable = yes” que dizer basicamente que essa pasta todos podem ler e escrever nessa pasta

Texto

Descrição gerada automaticamente

* Fazendo o compartilhamento da pasta oculta

A diferença da pasta de nome “publica” para a pasta de nome “oculta”, foi somente o parâmetro

“browseable = no”, ou seja, a pasta não vai está listada nas opções para você visualizar a pasta, mas si digitar o caminho do servidor e colocar “/oculta” vai consegui acessar essa pasta chamada de “oculta”, essa pasta só não vai está listada no compartilhamento, mas ela vai está oculta

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

* Fazendo o compartilhamento de uma pasta somente leitura

Nesse caso utilizamos o parâmetro “read only = yes” si o parâmetro estiver exatamente dessa maneira indica que a pasta será de somente leitura.Todos os arquivos que estiverem dessa pasta, no nosso caso a pasta “somente\_leitura” o usuário só pode ler o conteúdo desses arquivos, não vão conseguir modificar

Texto

Descrição gerada automaticamente

* Fazendo o compartilhamento de uma pasta com senha

Nesse caso o parâmetro “guest ok = no”, ou seja, todos os usuários que eu estiver cadastrado no Samba podem acessar a pasta de nome “publica\_com\_senha” desde que coloque o usuário e a senha dele para acessar a pasta

Texto

Descrição gerada automaticamente

* Fazendo o compartilhamento da pasta que apenas alguns usuários podem acessar

Nesse caso estou utilizando dois parâmetros novos

O parâmetro “valid users = jose,maria” basicamente eu digo que quem pode acessar essa pasta chamada “lista\_usuario” e apenas o usuário “jose” e o usuário “maria” que estão no Samba, nenhum outro usuário pode acessar. Se não ter os usuários “jose” e “maria” e necessário criar esses usuários.

O parâmetro “create mask = 0777” basicamente determina as permissões que os novos arquivos devem ter quando eles forem criados nessa pasta. Todos os arquivos que forem criados nessa pasta, vai ter a mascara 0777

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

* Fazendo o compartilhamento da pasta que apenas o usuário João tem permissão para escrever e os demais usuários podem somente ler os arquivos

O parâmetro “write list = joao” indica que apenas o usuário João pode fazer a leitura e a escrita, ou seja, ele pode modificar os arquivos que estão dentro da pasta chamada “lista\_usuario\_variacao”

O parâmetro “writable = no” e o parâmetro “guest ok = yes” indica que os restante dos usuários poderam somente ler os arquivos que encontram-se dentro dessa pasta Texto

Descrição gerada automaticamente

Observação: Parametro username = usuario\_ze: lista quais usuarios tem permissão para utilização do recurso. Neste caso do exemplo. Usuario\_ze é um usuario e ele terá permissão para utilização do recurso de acordo com a seção em que for inserido;

Parametro read list: lista os usuarios que tem acesso apenas de leitura ao compartilhamento

Parametro hosts allow = 192.168.0.2, 192.168.0.5 lista quais os IPs que podem ter acesso e se conectar no servidor. É possível combinar esse parâmetro com o “write list” para especificar que somente aqueles usuarios utilizando aquela máquina (n° IP) podem ter acesso ao servidor;

Parametro create mask = 0775 determina as permissões que novos arquivos devem ter quando criados

Parametro mask = 0775 similar ao create mask, mas se aplica para diretórios

Nosso arquivo ficara assim:

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Em seguida iremos salvar o arquivo

Ctrl + O

Enter

Ctrl + x

6° Passo: Verificar se está tudo correto com o arquivo

bruno@servidor:~/aula-samba$ testeparm

Texto

Descrição gerada automaticamente

7° Passo: Vou dar um restart no serviço Samba

bruno@servidor:~/aula-samba$ sudo service smbd restart

Texto

Descrição gerada automaticamente

8° Passo: Verificando o status do serviço Samba

bruno@servidor:~/aula-samba$ sudo service smbd status

Texto

Descrição gerada automaticamente

9° Criando os usuários João, Maria, Jose, pois só eles podem acessar algumas pastas compartilhadas que configuramos no arquivos smb.conf

Adicionando usuário João no Sistema/Servidor

bruno@servidor:~/aula-samba$ sudo adduser joao

Texto

Descrição gerada automaticamente

Em seguida:

Adicionando usuário João no Samba

bruno@servidor:~/aula-samba$ sudo smbpasswd -a joao

Texto

Descrição gerada automaticamente

Fazendo os mesmos passos anteriores para criar usuario para jose

Adicionando usuário Jose no Sistema/Servidor

bruno@servidor:~/aula-samba$ sudo adduser jose

Texto

Descrição gerada automaticamente

Em seguida:

Adicionando usuário João no Samba

bruno@servidor:~/aula-samba$ sudo smbpasswd -a jose

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Fazendo os mesmos passos anteriores para criar usuario para maria

Adicionando usuário Jose no Sistema/Servidor

bruno@servidor:~/aula-samba$ sudo adduser maria

Texto

Descrição gerada automaticamente

Em seguida:

Adicionando usuário João no Samba

bruno@servidor:~/aula-samba$ sudo smbpasswd -a jose

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

**Compartilhamento de Internet no Ubuntu Server**

Interface gráfica do usuário

Descrição gerada automaticamente com confiança média

No nosso caso:

Servidor Ubuntu Server tem dois adaptadores de redes:

Adaptador de rede 1: NAT enp0s3 : 10.0.2.15

Adaptador de rede 2: Rede Interna enp0s8 : 192.168.1.10

Nome: INFO4

SIMULANDO UM CLIENTE ->Máquina cliente: Windows 7

Adaptador de rede 1: Rede Interna

Nome: INFO4

1° Passo: Verificar se o Servidor tem acesso a internet

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Tenho acesso a internet, por meio da minha placa de rede “enp0s3” que está em modo NAT

Observação: Para a gente conseguir fazer o compartilhamento de internet, precisamos

1° Passo: transformar o servidor em um roteador;

2° Passo: Ativar o modo NAT e inscrever uma regra de IP TABLES

2°Passo: Adentrar como usuário “root”

bruno@servidor:~$ sudo su

3°Passo: Esse comando, basicamente escrevemos “1” dentro do arquivo “ip\_forward”, por padrão o arquivo “ip\_forward” vem com o valor 0, mas quando você coloca o valor 1 dentro desse arquivo e ativado o roteamento de pacotes, permitindo que o servidor consiga rotear pacotes de uma rede para outra. Basicamente ele passa a rotear os pacotes de uma interface para outra e vice-versa.

A partir do momento que colocamos o valor “1” ele começou a rotear os pacotes IPs da interface de “Rede Interna INFO4 enp0s8 : 192.168.1.10” para a interface “NAT enp0s3 : 10.0.2.15” e vice-versa

root@servidor:/home/bruno# echo 1 > /proc/sys/net/ipv4/ip\_forward

Texto

Descrição gerada automaticamente

4°Passo: Ativar o NAT no modo IP TABLES, em seguida escrever a regra de IP TABLES para fazer o roteamento de pacotes de uma interface interna para uma interface externa e vice-versa

Explicação, por que precisamos ativar o NAT no modo IP TABLES

Basicamente o modo NAT, e uma técnica que consiste em reescrever utilizando uma tabela Hash. Os endereços IPs (clientes) de origem de um pacote que passam por um roteador(servidor ubuntu) de uma rede interna para uma rede externa(NAT).

No NAT vai ter uma tabela HASH para controlar os endereços de origem e destino, já que os pacotes vão passar da rede interna para a rede externa e vice versa, precisando assim de uma tabela. Quem controla essa tabela e o “NAT”

Exemplo:

Diagrama

Descrição gerada automaticamente

Vamos supor que a máquina cliente “192.168.1.80” vai querer fazer um acesso ao IP de Destino 172.217.30.99, ou seja, fazer acesso a algum site que está na rede externa.Estou tentando acessar a internet utilizando a máquina cliente de IP 192.168.1.80

Quando chega no servidor (Ubuntu) que está configurado como NAT, ele traduz o “endereço de origem da rede interna 192.168.1.80” no “endereço de origem da interface de rede externa 10.0.2.15”, ou seja, que e o IP da rede externa do meu servidor. Essa tradução e aonde temos nossa tabela NAT/HASH.

Depois de solicitar ao site, ele devolveu ao “endereço de Destino 10.0.2.15”, quando chegar no Servidor Ubuntu ele vai traduzir novamente o “endereço de destino 10.0.2.15”

5° Passo:Estou ativando o modo NAT no modo IP TABLE



6°Passo: Escrever a regra de IP TABLES

Esse comando cria uma regra de IP TABLES para o NAT, estou dizendo que todos os pacotes que passarem “POSTROUTING” da origem “-s” da subnet “192.168.1.0/24” vai ter saída “-o” pela interface “enp0s3”

A interface “enp0s3” e a interface para a rede externa

“-s” source

“-o” output

O “-j MASQUERADE” e basicamente para dizer que os pacotes que estão na rede de origem 192.168.1.0 serão mascarados como si fosse os pacotes que estão saindo da interface “enp0s3”

Essa regra e como si fosse esse processo de ida do pacote para a internet

Diagrama

Descrição gerada automaticamente



root@servidor:/home/bruno# iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -o enp0s3 -j MASQUERADE

Observação: Por acaso se reiniciar o servidor, será necessário digitar novamente esses comandos

Texto

Descrição gerada automaticamente





Pode ser configurado no arquivo “etc/rc.local” inserir esses comandos para que todas as vezes que inicie o servidor ele consiga executar os comandos que está dentro desses arquivos



Dentro desse arquivo “rc.local” eu posso digitar os 3 comandos

Texto

Descrição gerada automaticamente





Que serão executados toda vez que você reiniciar o servidor