

Faculdade de Tecnologia – FATEC Centro Paula Souza – CPS

Alessandra de Souza Lopes
Éverson Adriano Bernardes Lemes

Protocolos e Roteamento de Redes
Detonado de IPv4

Ourinhos- São Paulo
2024

Alessandra de Souza Lopes
Éverson Adriano Bernardes Lemes

Protocolos e Roteamento de Redes
Detonado de IPv4

Trabalho desenvolvido na
disciplina de protocolos e
roteamento de redes,
proposto pelo Dr. Thiago
Lucas.

Ourinhos- São Paulo
2024

Sumário

| | |
|---------------------------|----|
| Resumo..... | 6 |
| Conceitos..... | 7 |
| Comutadores..... | 8 |
| Hub..... | 8 |
| Switch..... | 9 |
| Switch e Hub..... | 9 |
| Camada de rede..... | 10 |
| Modelo OSI..... | 10 |
| | 11 |
| Modelo TCP/IP..... | 11 |
| | 12 |
| IPv4..... | 12 |
| Detonado de IPv4..... | 12 |
| 1ºDiretriz..... | 13 |
| 2ºDiretriz..... | 13 |
| | 21 |
| 3ºDiretriz..... | 22 |
| | 25 |
| 4ºDiretriz..... | 25 |
| | 30 |
| Teste..... | 30 |
| | 32 |
| 5ºDiretriz..... | 32 |
| | 41 |
| 6ºDiretriz..... | 41 |
| | 48 |
| Considerações finais..... | 49 |
| Lista de códigos..... | 50 |
| Bibliografia..... | 51 |

Lista de Figuras

| | |
|-----------------------------------|----|
| Figura 01 – Hub..... | 8 |
| Figura 02 – Switch..... | 9 |
| Figura 03 – Hub e Switch..... | 9 |
| Figura 04 – Modelo OSI..... | 10 |
| Figura 05 – TCP/IP..... | 11 |
| Figura 06 – Logo Virtual Box..... | 12 |
| Figura 07 – Organograma..... | 13 |
| Figura 08 – tela1..... | 13 |
| Figura 09 – tela2..... | 14 |
| Figura 10 – tela3..... | 14 |
| Figura 11 – tela4..... | 15 |
| Figura 12 – tela5..... | 15 |
| Figura 13 – tela6..... | 16 |
| Figura 14 – tela7..... | 16 |
| Figura 15 – tela8..... | 17 |
| Figura 16 – tela9..... | 17 |
| Figura 17 – tela10..... | 18 |
| Figura 18 – tela11..... | 18 |
| Figura 19 – tela12..... | 19 |
| Figura 20 – tela13..... | 19 |
| Figura 21 – tela14..... | 20 |
| Figura 22 – tela15..... | 20 |
| Figura 23 – tela16..... | 21 |
| Figura 24 – tela17..... | 21 |
| Figura 25 – tela18..... | 22 |
| Figura 26 – tela19..... | 22 |
| Figura 27 – tela20..... | 22 |
| Figura 28 – tela21..... | 23 |
| Figura 29 – tela22..... | 23 |
| Figura 30 – tela23..... | 23 |
| Figura 31 – tela24..... | 24 |
| Figura 32 – tela25..... | 24 |
| Figura 33 – tela26..... | 24 |
| Figura 34 – tela27..... | 25 |
| Figura 35 – tela28..... | 25 |
| Figura 36 – tela29..... | 25 |
| Figura 37 – tela30..... | 25 |
| Figura 38 – tela31..... | 26 |
| Figura 39 – tela32..... | 26 |
| Figura 40 – tela33..... | 26 |
| Figura 41 – tela34..... | 27 |
| Figura 42 – tela35..... | 27 |
| Figura 43 – tela36..... | 27 |
| Figura 44 – tela37..... | 28 |
| Figura 45 – tela38..... | 28 |
| Figura 46 – tela39..... | 28 |
| Figura 47 – tela40..... | 28 |
| Figura 48 – tela41..... | 29 |
| Figura 49 – tela42..... | 29 |
| Figura 50 – tela43..... | 29 |
| Figura 51 – tela44..... | 30 |
| Figura 52 – tela45..... | 30 |
| Figura 53 – tela46..... | 31 |

| | |
|--------------------------|----|
| Figura 54 – tela47..... | 31 |
| Figura 55 – tela48..... | 31 |
| Figura 56 – tela49..... | 32 |
| Figura 57 – tela50..... | 32 |
| Figura 58 – tela51..... | 33 |
| Figura 59 – tela52..... | 33 |
| Figura 60 – tela53..... | 34 |
| Figura 61 – tela54..... | 34 |
| Figura 62 – tela55..... | 35 |
| Figura 63 – tela56..... | 35 |
| Figura 64 – tela57..... | 36 |
| Figura 65 – tela58..... | 36 |
| Figura 66 – tela59..... | 37 |
| Figura 67 – tela60..... | 37 |
| Figura 68 – tela61..... | 38 |
| Figura 69 – tela62..... | 38 |
| Figura 70 – tela63 | 39 |
| Figura 71 – tela64..... | 39 |
| Figura 72 – tela65..... | 40 |
| Figura 73 – tela66..... | 40 |
| Figura 74 – tela67..... | 41 |
| Figura 75 – tela68..... | 41 |
| Figura 76 – tela69..... | 42 |
| Figura 77 – tela70..... | 42 |
| Figura 78 – tela71..... | 43 |
| Figura 79 – tela72..... | 43 |
| Figura 80 – tela73..... | 43 |
| Figura 81 – tela74..... | 44 |
| Figura 82 – tela75..... | 44 |
| Figura 83 – tela76..... | 44 |
| Figura 84 – tela77..... | 45 |
| Figura 85 – tela78..... | 45 |
| Figura 86 – tela79..... | 46 |
| Figura 87 – tela80..... | 46 |
| Figura 88 – tela81..... | 47 |
| Figura 89 – tela82..... | 47 |
| Figura 90 – tela83..... | 47 |
| Figura 91 – tela84 | 48 |

Resumo

Este trabalho, elaborado como parte da disciplina de protocolos e roteamento de redes, explora o processo de comunicação entre servidores e clientes nos sistemas operacionais Linux e Windows. Além disso, discute-se o papel dos comutadores de rede, destacando suas características distintas e qualidades. São apresentadas as diferenças entre os sistemas operacionais Linux e Windows no contexto de servidor e cliente, bem como uma análise comparativa das funcionalidades dos comutadores. O objetivo é fornecer uma compreensão abrangente desses elementos essenciais no contexto das redes de computadores, contribuindo para um melhor entendimento de sua operação e configuração.

Conceitos

1. LAN (Local Area Network): Uma rede de área local é uma rede de computadores que abrange uma área limitada, geralmente dentro de um único prédio ou campus. É usada para conectar dispositivos próximos, como computadores, impressoras e dispositivos de armazenamento em rede.
2. MAC (Media Access Control): Endereço MAC é um identificador único atribuído a cada placa de interface de rede (NIC) para comunicação em uma rede. É uma série de números hexadecimal que identifica exclusivamente um dispositivo em uma rede.
3. IP (Internet Protocol): Endereço IP é um número de identificação atribuído a cada dispositivo em uma rede que usa o protocolo IP para comunicação. Existem dois principais padrões de IP: IPv4 e IPv6.
4. NAT (Network Address Translation): Tradução de Endereço de Rede é um processo usado pelos roteadores para modificar os endereços IP de origem ou destino em pacotes de dados enquanto eles estão sendo transmitidos através de uma rede. Isso é comumente usado para permitir que vários dispositivos em uma rede privada compartilhem um único endereço IP público.
5. NETMASK (Máscara de Sub-rede): Uma máscara de sub-rede é um conjunto de números binários que especifica quais bits de um endereço IP correspondem à parte da rede e à parte do host. Ela é usada em conjunto com o endereço IP para determinar a rede e os hosts disponíveis em uma LAN.
6. DHCP (Dynamic Host Configuration Protocol): Protocolo de Configuração Dinâmica de Host é um protocolo de rede que permite que dispositivos em uma rede obtenham automaticamente um endereço IP e outras informações de configuração de um servidor DHCP, eliminando a necessidade de configuração manual.
7. ADDRESS (Endereço): Em termos de rede, pode se referir a um endereço IP atribuído a um dispositivo na rede.
8. GATEWAY (Porta de Entrada): É o ponto de entrada para outra rede a partir da rede local. Geralmente é o endereço IP do roteador que conecta a LAN à internet ou a outra rede externa.
9. LOOPBACK: O endereço de loopback é um endereço especial usado para testar a conectividade da interface de rede de um dispositivo com ele mesmo.

Comutadores

Comutadores, também conhecidos como switches, são dispositivos de rede utilizados para interligar diversos dispositivos em uma rede local (LAN - Local Area Network). Eles operam na camada de enlace de dados do modelo OSI (Open Systems Interconnection) e são responsáveis por encaminhar pacotes de dados entre dispositivos dentro da mesma rede.

Os comutadores funcionam analisando os endereços MAC (Media Access Control) dos dispositivos conectados às suas portas e encaminhando os dados apenas para a porta específica onde o destinatário está localizado. Isso melhora a eficiência da rede, pois reduz o tráfego desnecessário, tornando a comunicação mais rápida e segura.

Existem diferentes tipos de comutadores, incluindo os comutadores de camada 2, que operam principalmente com endereços MAC, e os comutadores de camada 3, que também podem rotear dados com base em endereços IP, tornando-os mais sofisticados em redes maiores ou mais complexas.

No geral, os comutadores desempenham um papel fundamental na infraestrutura de rede moderna, facilitando a comunicação entre dispositivos dentro de uma LAN e contribuindo para o desempenho eficiente e confiável das redes de computadores.

HUB



Figura 01 "HUB"
FONTE:

Um "hub" é um dispositivo de rede que atua como um ponto central para conectar dispositivos em uma rede local. Ele é comumente utilizado para conectar vários dispositivos, como computadores, impressoras, câmeras etc. Em uma rede local para permitir a comunicação entre eles. Existem diferentes tipos de hubs, incluindo hubs Ethernet, hubs USB e hubs de energia. Eles trabalham para receber dados de um dispositivo e transmiti-los para todos os outros dispositivos conectados à rede. No entanto, é importante notar que, com o avanço da tecnologia, os hubs estão sendo amplamente substituídos por switches de rede, que oferecem melhor desempenho e eficiência na comunicação de dados em uma rede.

Switch



Figura 02 “SWITCH”

Um switch é um dispositivo de rede usado para conectar vários dispositivos em uma rede local (LAN) e direcionar o tráfego de dados entre eles de forma eficiente. Ao contrário de um hub, que simplesmente repete os dados recebidos para todos os dispositivos na rede, um switch é capaz de identificar o endereço de destino de cada pacote de dados e encaminhá-lo apenas para o dispositivo correto, minimizando o tráfego desnecessário na rede e melhorando a eficiência do uso da largura de banda.

Switch e Hub

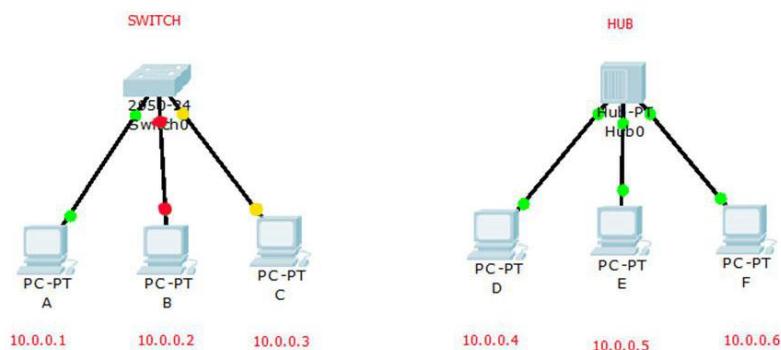


Figura 03 “Hub e switch”

Hub repete todos os dados em todas as portas, causando congestionamento e reduzindo a eficiência da rede. Ele compartilha toda a largura de banda entre os dispositivos, pode ser plug-and-play e oferece menos segurança, já a Switch encaminha dados apenas para o dispositivo correto, minimizando o tráfego desnecessário e melhorando a eficiência. Fornece largura de banda dedicada para cada porta, pode ser gerenciado ou não, também oferece mais segurança ao reduzir a exposição do tráfego de rede a dispositivos não autorizados.

Camada de Rede

Uma "camada de rede" refere-se a um dos níveis de abstração em um modelo de arquitetura de rede, como o modelo OSI (Open Systems Interconnection) ou o modelo TCP/IP (Transmission Control Protocol/Internet Protocol). Eles são usados para entender e padronizar a comunicação entre dispositivos em uma rede de computadores.

Na maioria dos modelos de arquitetura de rede, a camada de rede é responsável pelo roteamento dos dados de origem para o destino, garantindo que eles cheguem de forma efetiva e confiável. Ela trata de questões como endereçamento lógico, encaminhamento de pacotes, controle de congestionamento e fragmentação de dados.

Em resumo, a camada de rede fornece os meios para estabelecer, manter e finalizar conexões entre dispositivos em uma rede, independentemente de sua localização física na rede. Isso permite a comunicação entre dispositivos em redes locais, redes de longa distância e a Internet como um todo.

Modelo OSI

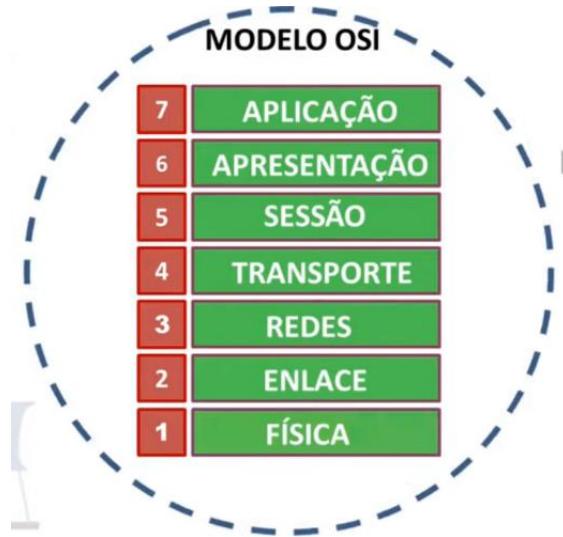


Figura 04 “Modelo OSI”

1. Camada Física.

- Lida com a transmissão física dos dados através de meios como cabos, fibra óptica ou sem fio.

2. Camada de Enlace de Dados.

- Responsável pela comunicação entre dispositivos diretamente conectados.
- Gerencia erros de transmissão e controla o fluxo de dados.

3. Camada de Rede.

- Foca na entrega de dados de origem para destino, independentemente do meio físico.
- Roteamento e encaminhamento de pacotes são realizados nesta camada.

4. Camada de Transporte.

- Fornece comunicação de fim a fim entre os aplicativos.
- Controla a transmissão de dados, garantindo que eles sejam entregues na ordem correta e sem erros.

5. Camada de Sessão.

- Gerencia e estabelece sessões de comunicação entre sistemas.
- Controla o diálogo entre aplicativos em diferentes computadores.

6. Camada de Apresentação.

- Lida com a tradução, compressão e criptografia de dados.
- Garante que os dados sejam apresentados de forma compreensível pelo aplicativo.

7. Camada de Aplicação.

- Fornece interfaces para aplicativos de usuário e serviços de rede.
- Inclui protocolos de aplicativos como HTTP, SMTP e FTP.

Modelo TCP/IP:



Figura 05“Tcp ip”

1. Camada de Interface de Rede.

- Equivalente às camadas Física e de Enlace de Dados do modelo OSI.
- Lida com a transmissão de dados entre dispositivos na mesma rede local.

2. Camada de Internet.

- Equivalente à camada de Rede do modelo OSI.
- Responsável pela transmissão de dados entre redes diferentes, incluindo roteamento e endereçamento IP.

3. Camada de Transporte.

- Similar à camada de Transporte do modelo OSI.
- Fornece comunicação fim a fim, com controle de fluxo e correção de erros.

4. Camada de Aplicação.

- Agrega funcionalidades das camadas de Apresentação e de Aplicação do modelo OSI.
- Inclui protocolos de aplicativos como HTTP, FTP, DNS e SMTP.

IPV4

IPv4 (Internet Protocol version 4) é uma das versões do protocolo de Internet mais amplamente utilizadas para atribuir endereços únicos a dispositivos em uma rede.

1. Endereços IPv4: Os endereços IPv4 consistem em quatro conjuntos de números decimais separados por pontos, por exemplo, 192.168.1.1. Cada conjunto pode variar de 0 a 255.

2. 32 bits: O endereço IPv4 é um número de 32 bits. Isso significa que há um total de 2^{32} (ou aproximadamente 4,3 bilhões) de endereços IPv4 possíveis.

3. Classes de endereço: Os endereços IPv4 são divididos em classes, que foram originalmente usadas para identificar o tamanho da rede. As classes incluem A, B, C, D e E.

4. CIDR (Classless Inter-Domain Routing): É um sistema de notação que permite uma maneira mais flexível de atribuir e rotear endereços IP. Ele usa uma combinação de endereço IP e máscara de sub-rede para representar uma rede.

5. Protocolos associados: IPv4 é frequentemente usado em conjunto com outros protocolos de rede, como TCP (Transmission Control Protocol) e UDP (User Datagram Protocol), formando o conjunto de protocolos TCP/IP, que é a base da Internet.

Detonado de roteamento IPv4

Para iniciar, é necessário dispor de um aplicativo ou dispositivo capaz de criar máquinas virtuais, possibilitando a visualização do encaminhamento de roteamento. Para começar, faremos uso do Oracle VM VirtualBox¹.



Figura 06“logovirtualbox”

Oracle VM VirtualBox¹. VirtualBox é um software de virtualização desenvolvido pela empresa Innotek, visa criar ambientes para instalação de sistemas distintos.

1º Diretriz

Primeiramente, vamos construir um organograma de com nossa rede será. Nesse projeto teremos o servidor de configuração Linux Debian que será o provedor de internet para as demais máquinas. A máquina 1 tem a configuração Debian juntamente com a máquina 2, já a restante será uma Windows.

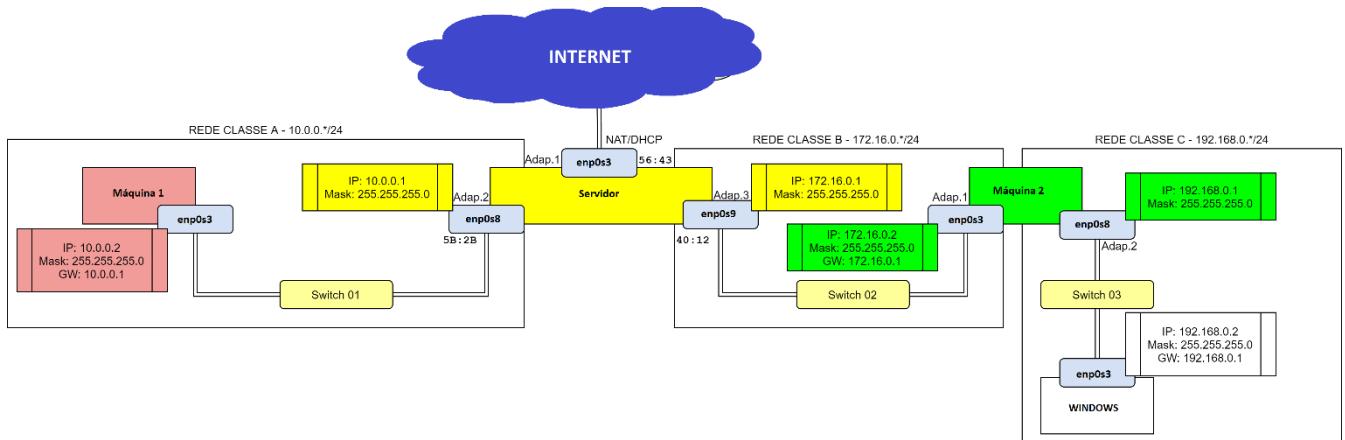


Figura 07 “Organograma”

O tutorial a seguir tem como objetivo traçar a conexão das máquinas com o promotor de internet e a conversação entre elas.

2º Diretriz

No VirtualBox, é necessário criar quatro máquinas virtuais, das quais três serão sistemas Linux Debian - uma atuando como servidor e provedor de internet, enquanto as outras duas serão clientes. A quarta máquina será um sistema Windows, também designada como cliente.

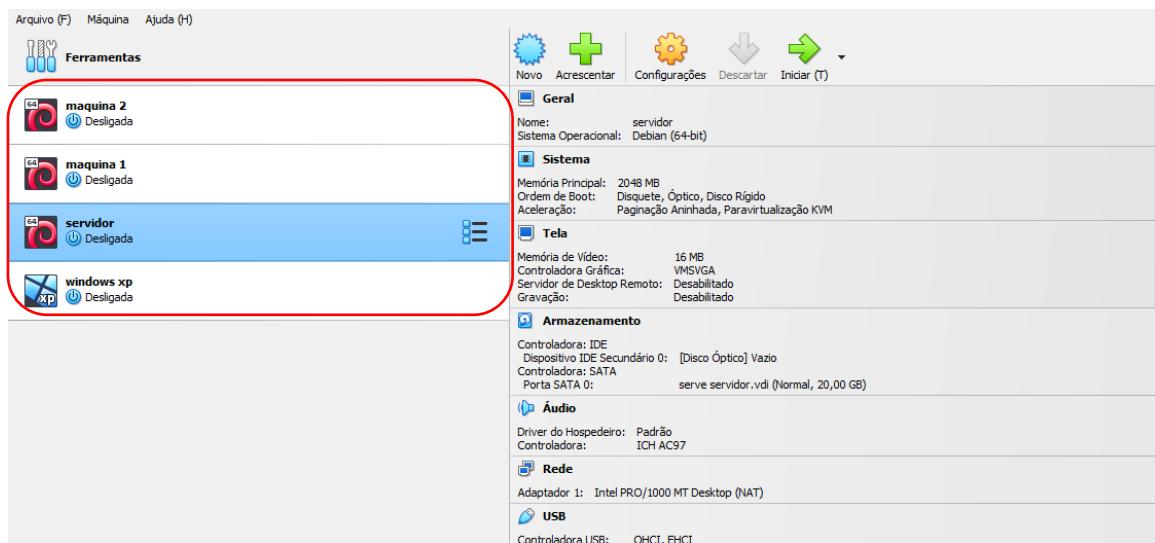


Figura 08 “Tela1”

Na máquina designada como servidor, é necessário ajustar as configurações de rede de ambos os adaptadores, tanto o Adaptador 1 quanto o Adaptador 2 e 3. No entanto, devemos entrar nas configurações de rede conforme a imagem abaixo.

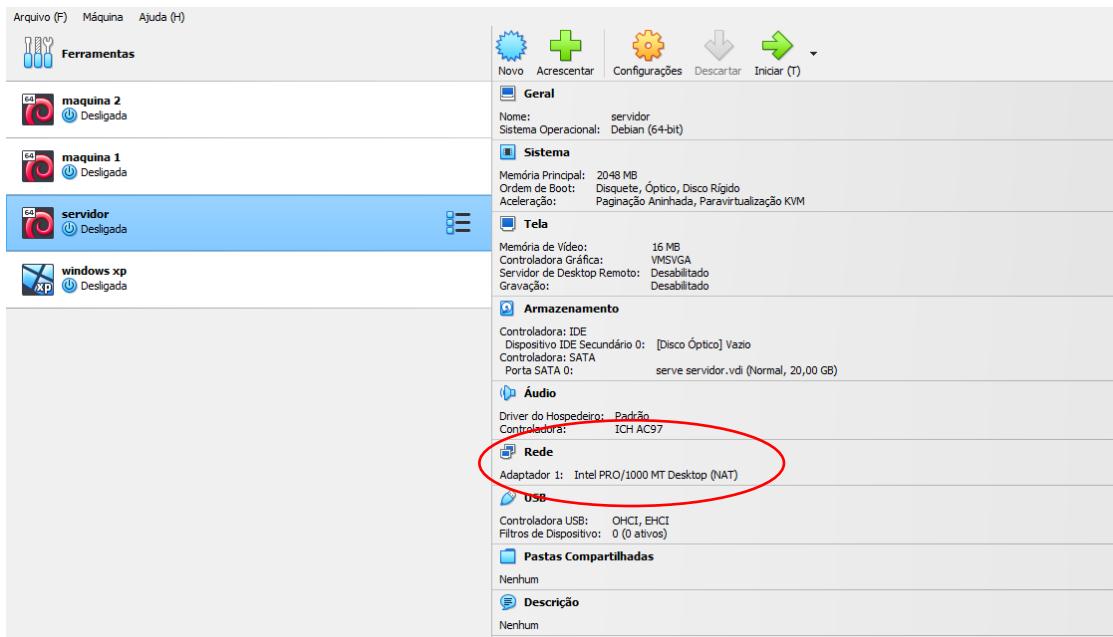


Figura 09 “Tela2”

Depois de acessarmos a tela mencionada, precisamos verificar se o Adaptador 1 está configurado como NAT¹. Caso não esteja, é necessário alterá-lo para a configuração mencionada a seguir.

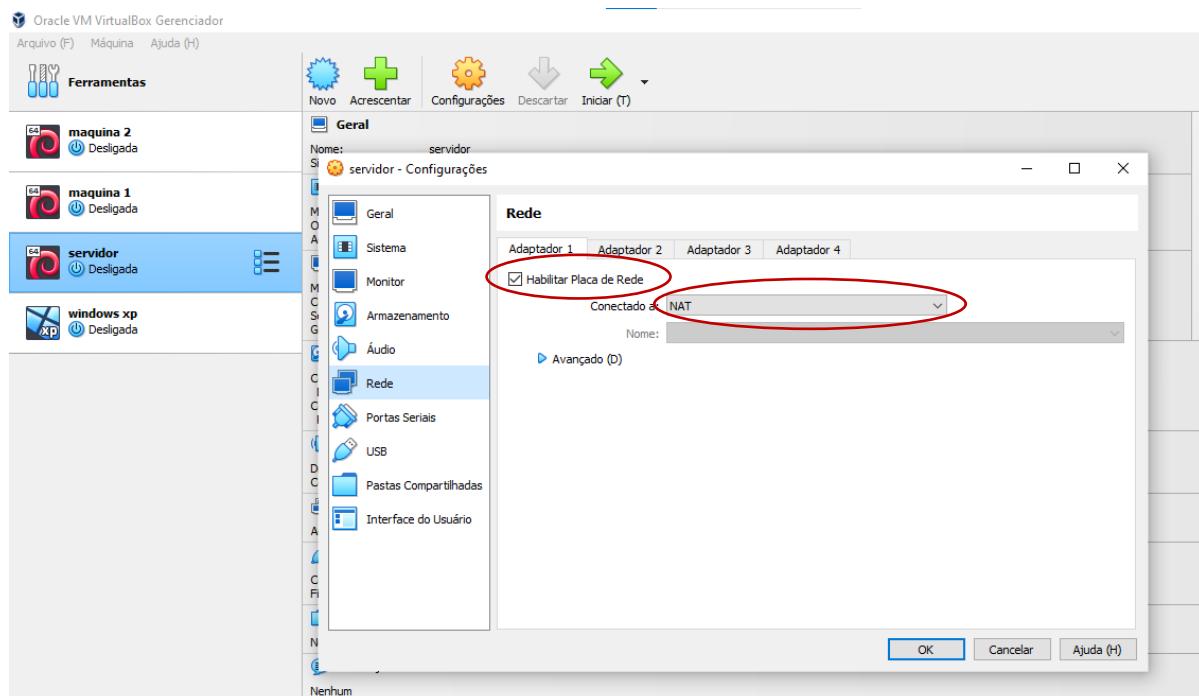


Figura 10 “Tela3”

NAT¹- (Network Address Translation) é usado em redes de computadores para traduzir endereços IP de dispositivos internos para um endereço IP público compartilhado, permitindo que vários dispositivos em uma rede privada compartilhem uma única conexão de Internet.

Depois de configurar o adaptador 1, é necessário escolher o adaptador 2, que geralmente é reconhecido sem exigir qualquer configuração adicional, conforme mostrado na imagem abaixo.

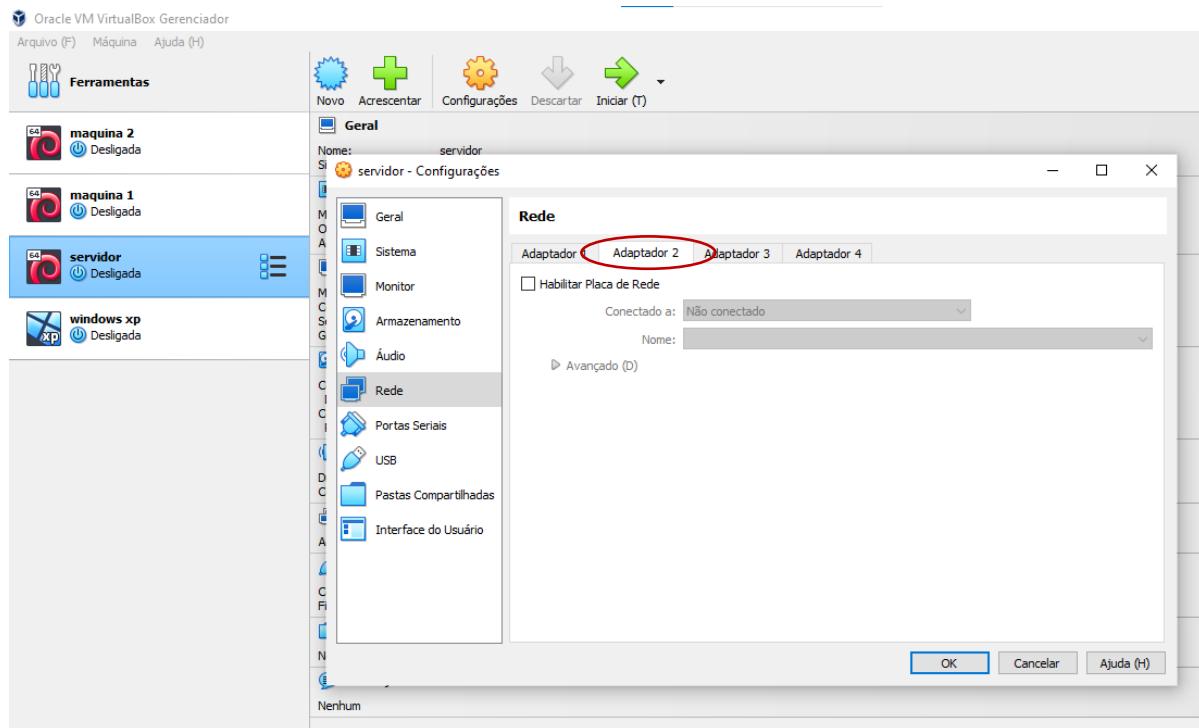


Figura 11 “Tela4”

Depois de verificar o adaptador 2, é preciso ativar sua placa de rede e designá-la como uma rede interna. Esta rede interna deve ser denominada como SWITCH 1, já que será o primeiro switch conectado ao servidor.

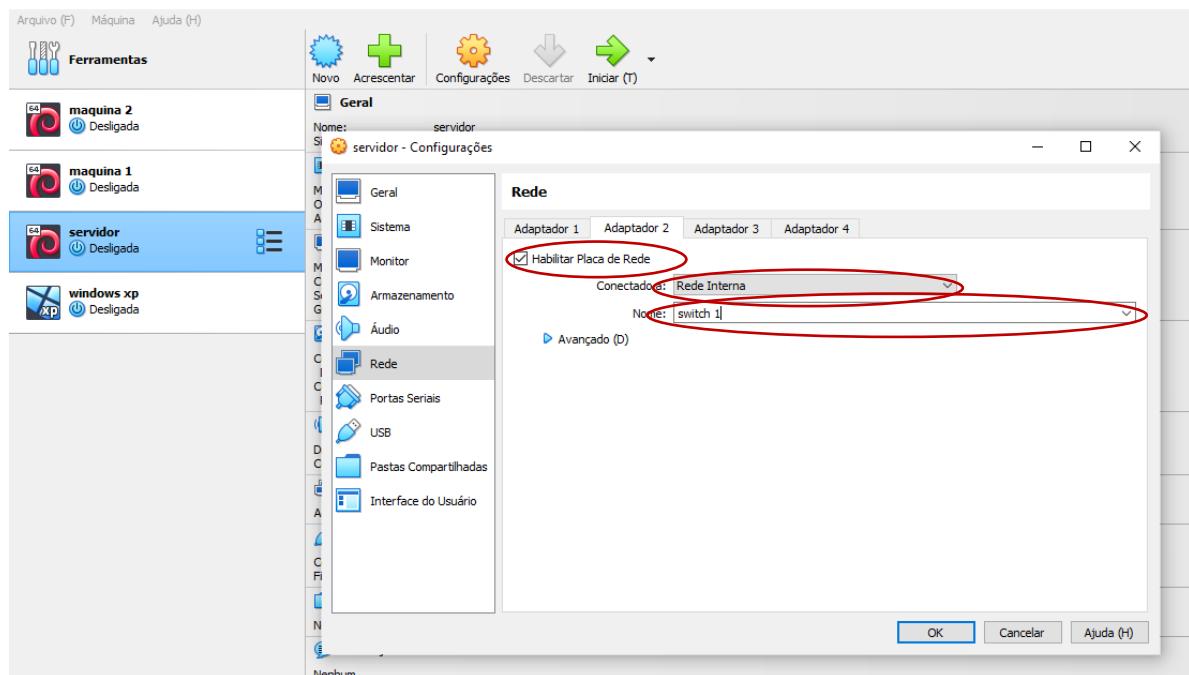


Figura 12 “Tela5”

Após a configuração do adaptador 2, é necessário seguir os mesmos procedimentos com o adaptador 3. No entanto, devemos atribuir um nome de switch diferente para cada, garantindo assim que o sistema seja capaz de distinguir entre eles.

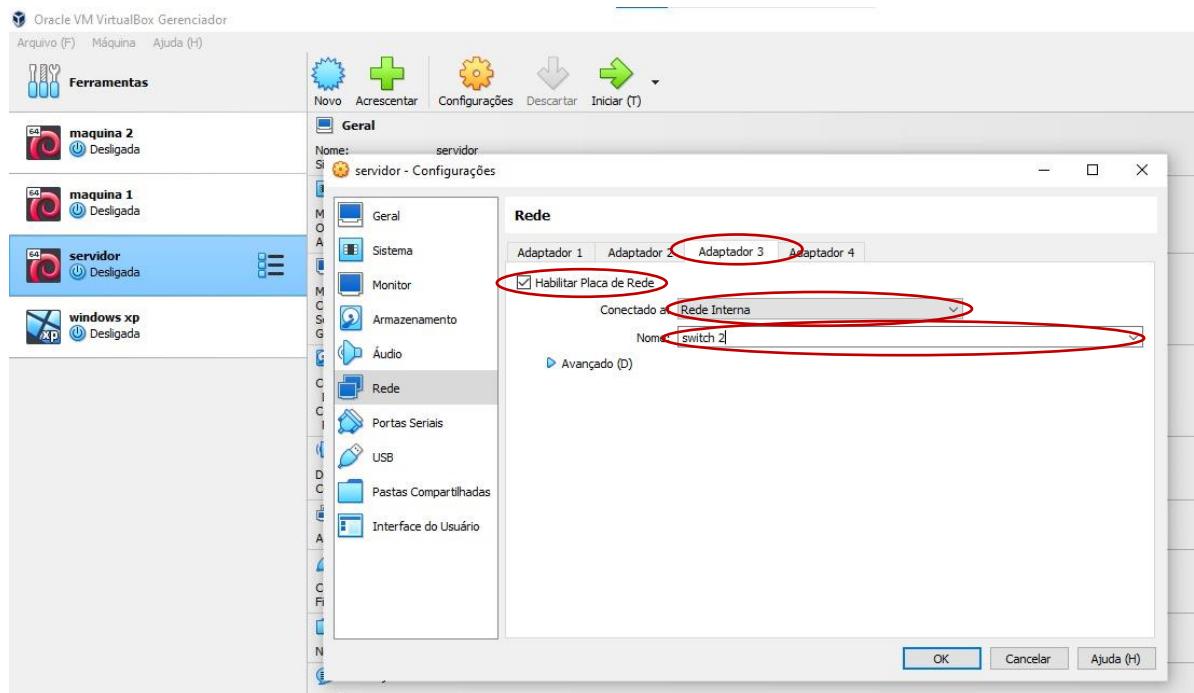


Figura 13 “Tela6”

Para salvar as configurações realizadas na máquina cliente, basta clicar em "OK".

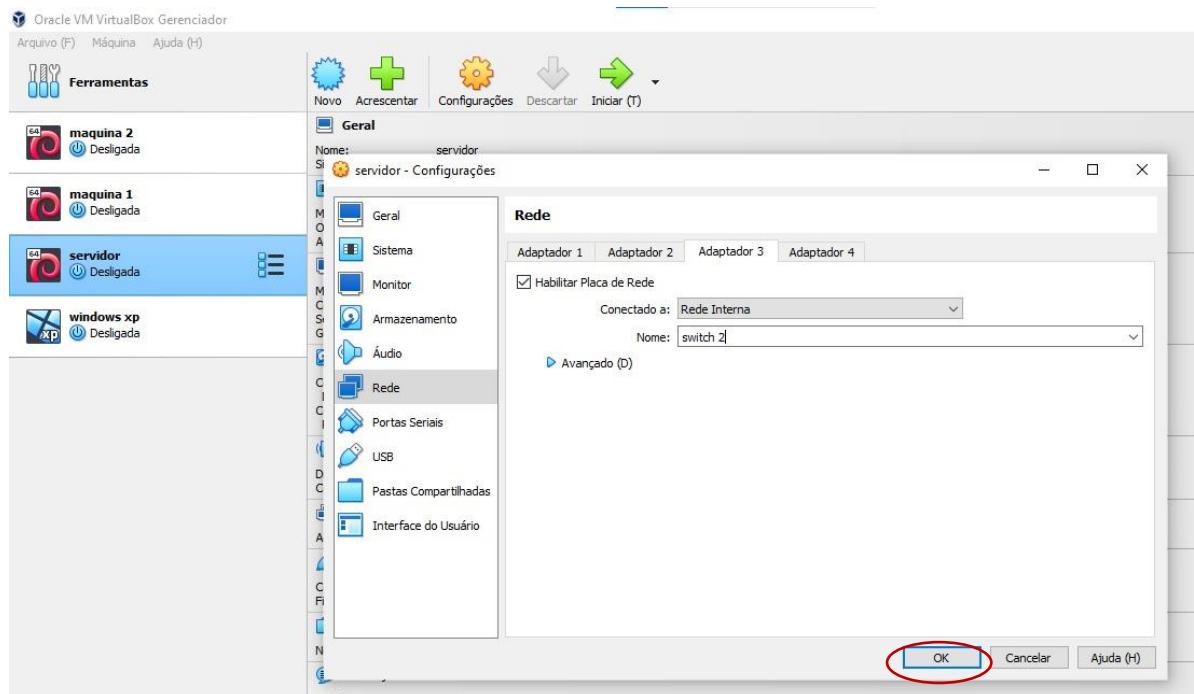


Figura 14 “Tela7”

Após concluir as configurações mencionadas, é necessário acessar a rede de configuração na máquina 1.

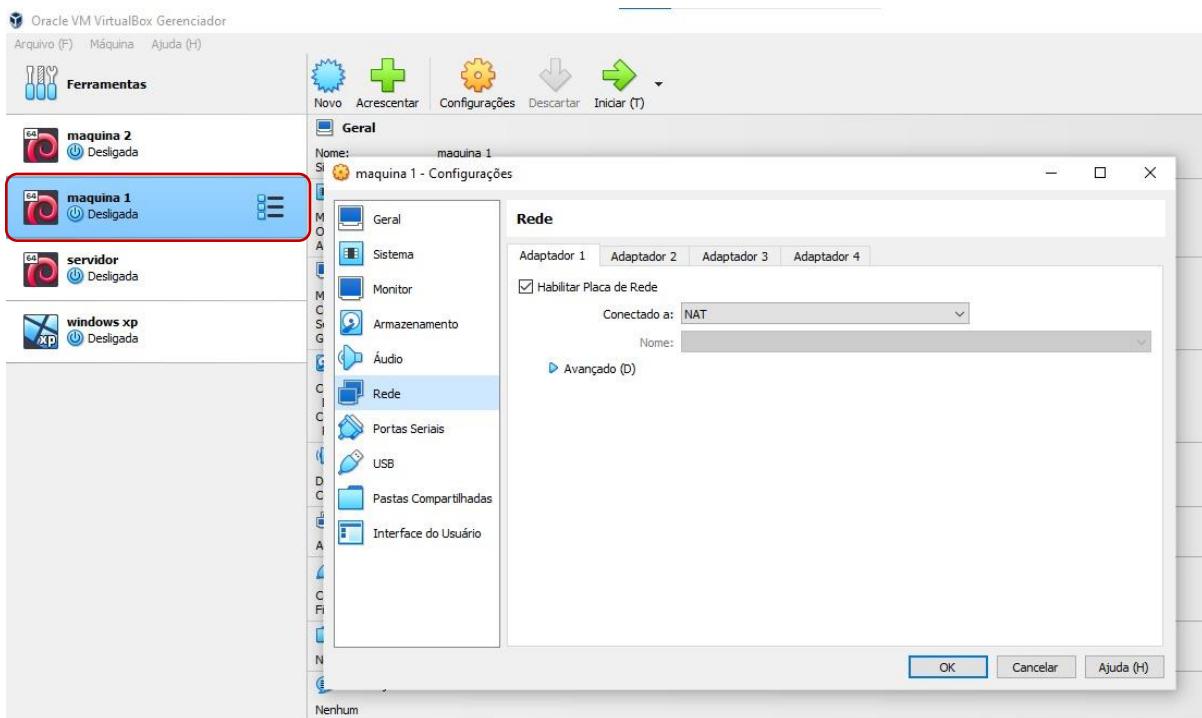


Figura 15 “Tela8”

É possível observar que o estado de configuração inicial é idêntico ao da máquina servidor. No adaptador 1, iremos modificar a configuração de NAT para rede interna e renomear como "switch 1" novamente, pois esta será a chave para as conexões entre a máquina cliente e a máquina servidor.

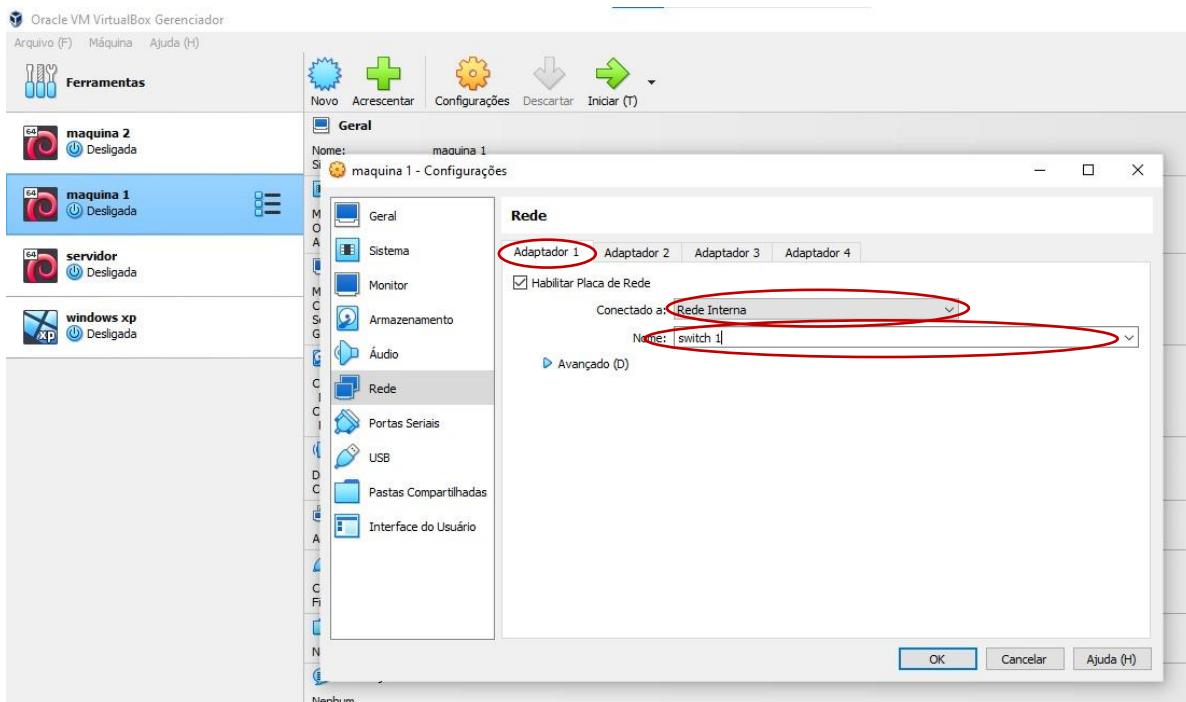


Figura 16 “Tela9”

Após concluir as configurações, basta clicar em "OK" para salvar as modificações realizadas. Além disso, após essas alterações, também é necessário modificar a rede da máquina dois, então deve-se acessar as configurações de rede dela.

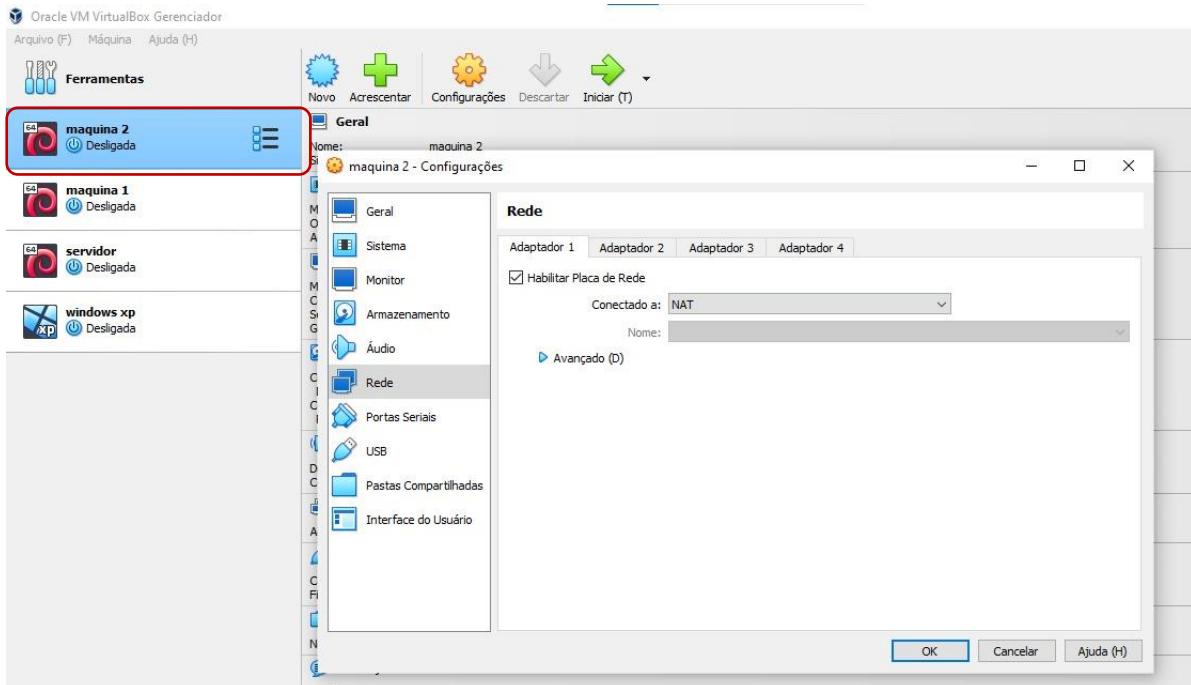


Figura 17 “Tela10”

Observamos que mais uma vez seu estado inicial é semelhante ao da máquina servidor e da máquina 1. É necessário alterar novamente a conexão de NAT para rede interna e renomear desta vez como Switch 2, pois esta é a chave de conexão entre o servidor e o cliente "máquina 2".

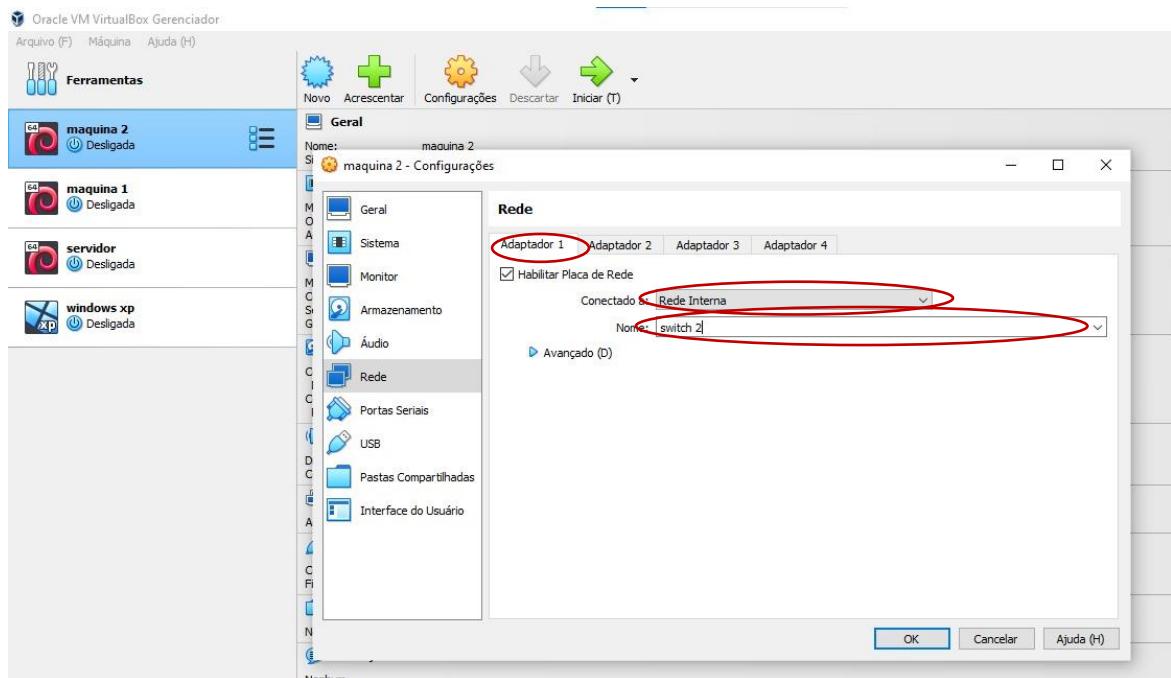


Figura 18 “Tela11”

Na máquina 2, serão conectados duas switches. Portanto, após finalizar a modificação do adaptador 1, é necessário configurar o adaptador 2 da mesma forma. Transformando-o em um servidor para a máquina Windows, é preciso habilitar sua placa de rede e conectá-la a uma rede interna, denominando-o como Switch 3, já que essa será a conexão entre ambos.

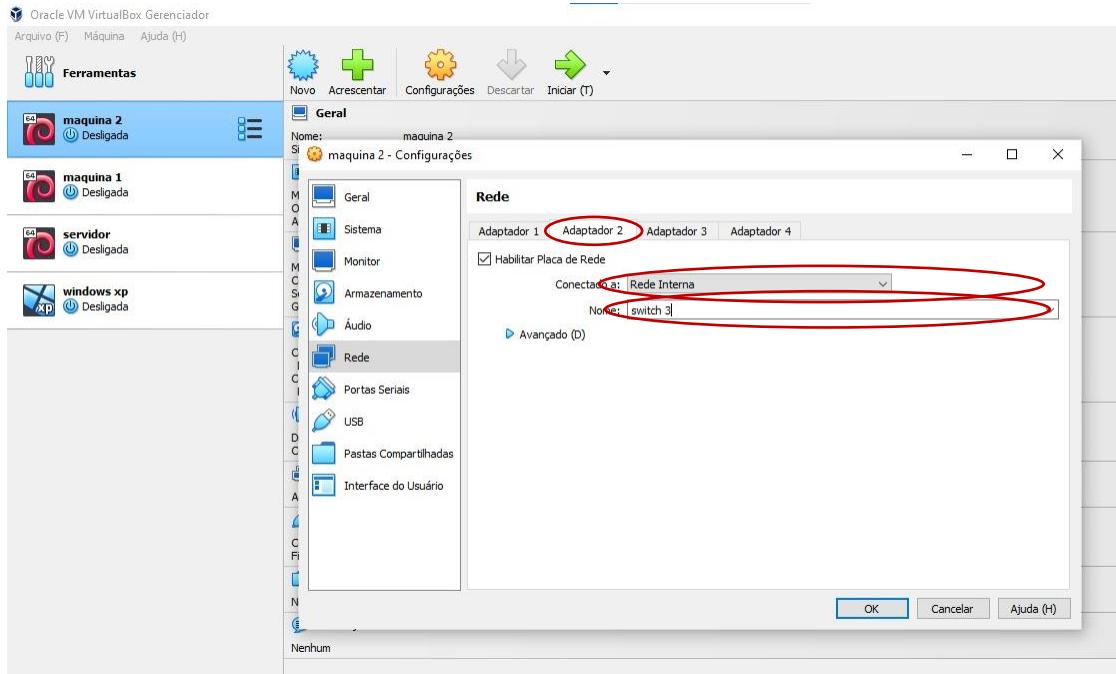


Figura 19 “Tela12”

Após concluir as alterações, para salvar, basta clicar em "OK". Na máquina Windows, devemos seguir os mesmos procedimentos, acessando suas configurações de rede. No adaptador 1, habilitar sua placa, conectá-la a uma rede interna e nomeá-la como switch 3, que é a conexão dela com o servidor.

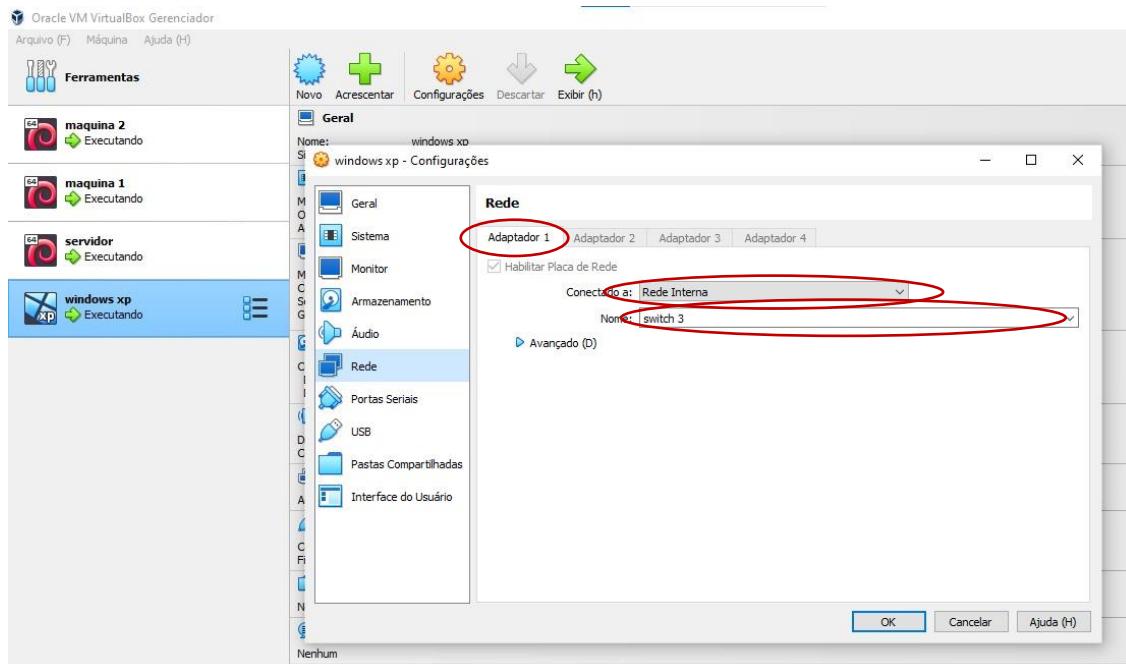


Figura 20 “Tela13”

Depois de terminar as alterações, é só clicar em "OK" para guardar. Seria útil registrar o endereço MAC de cada máquina para simplificar nossas configurações. Para isso, ainda nas configurações de rede nos adaptadores, basta clicar em "Avançado", e ele nos fornecerá o endereço MAC.

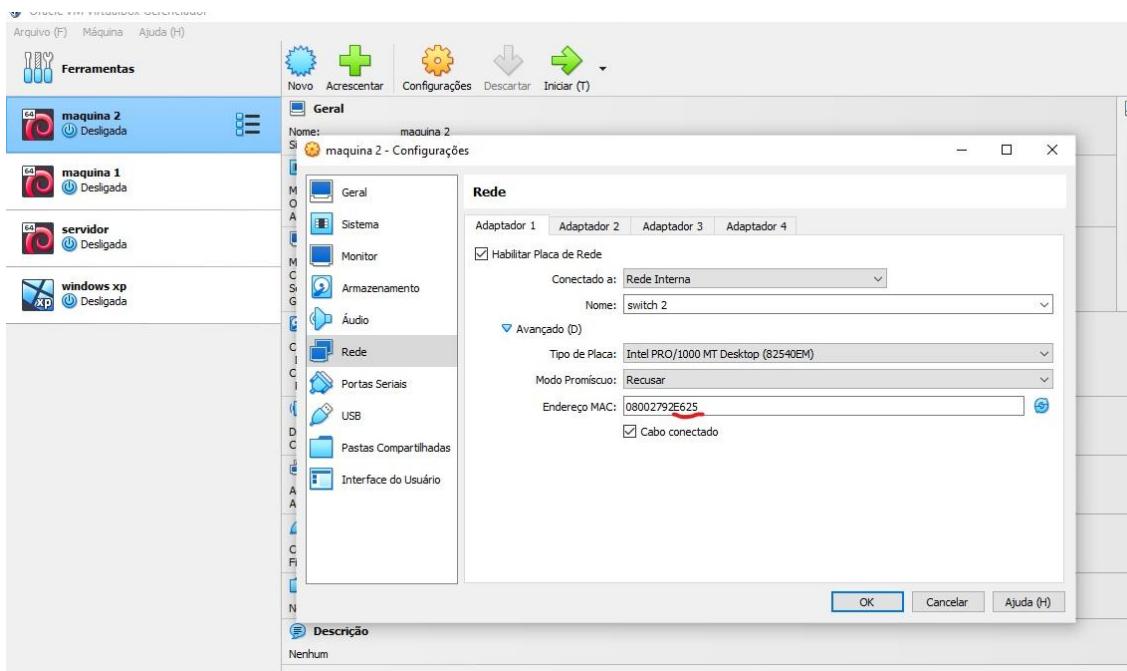


Figura 21 “Tela14”

Deve-se fazer isso em cada máquina presente para facilitar as configurações e identificações.

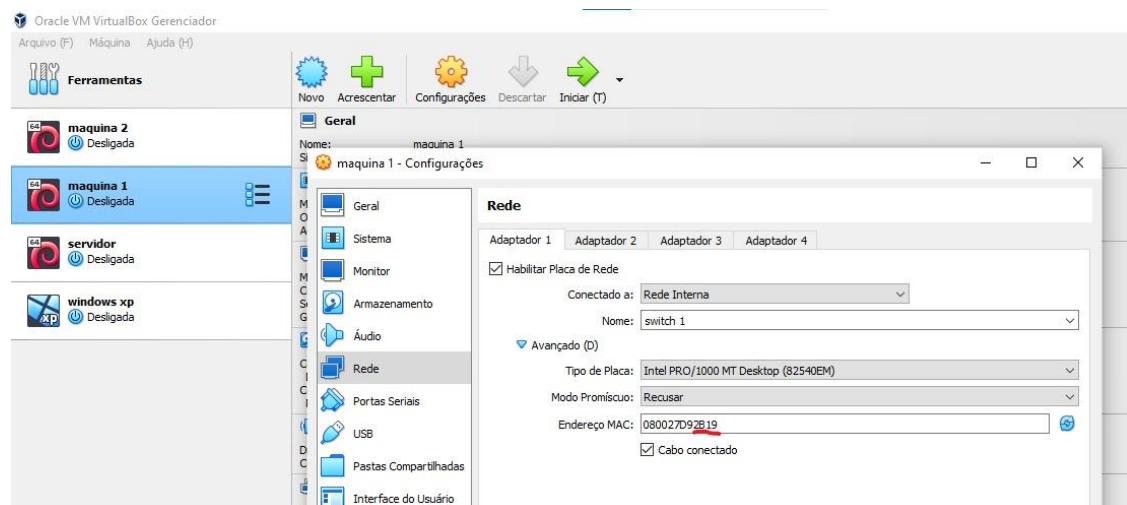


Figura 22 “Tela15”

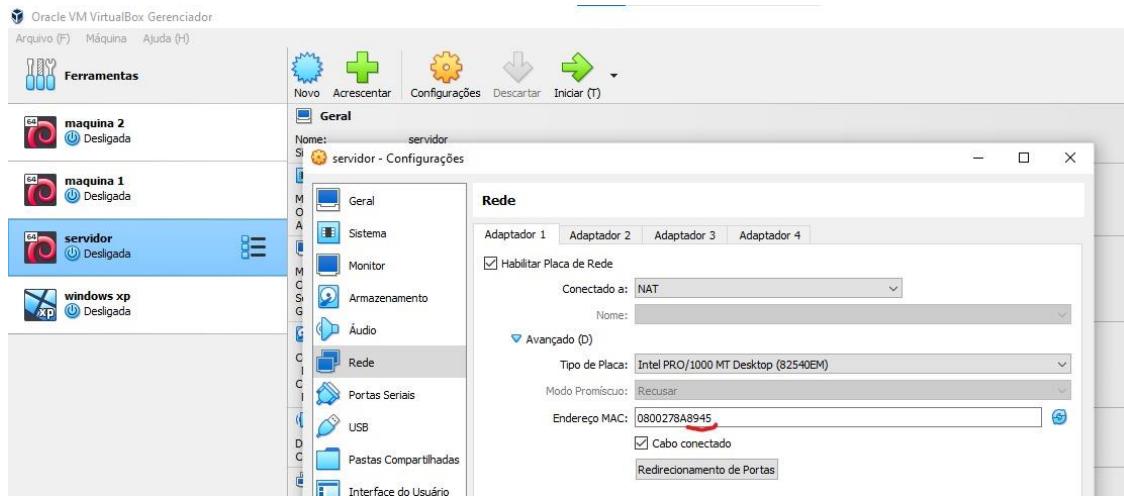


Figura 23 “Tela16”

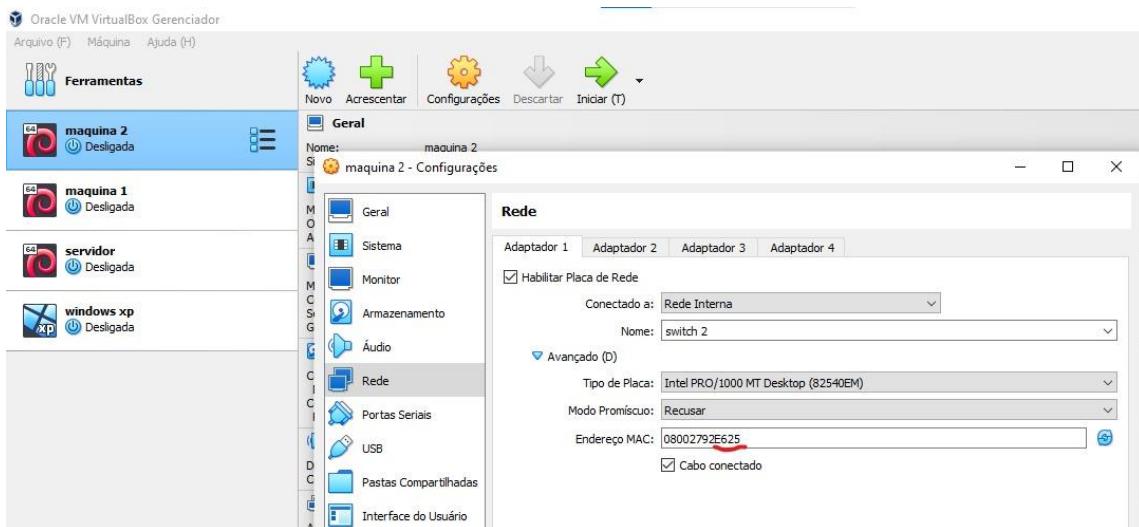


Figura 24 “Tela17”

Após a configuração desses procedimentos, deve-se seguir para a terceira diretriz.

3º Diretriz

Após a finalização da segunda diretriz deve-se logar na máquina que designamos como servidor.

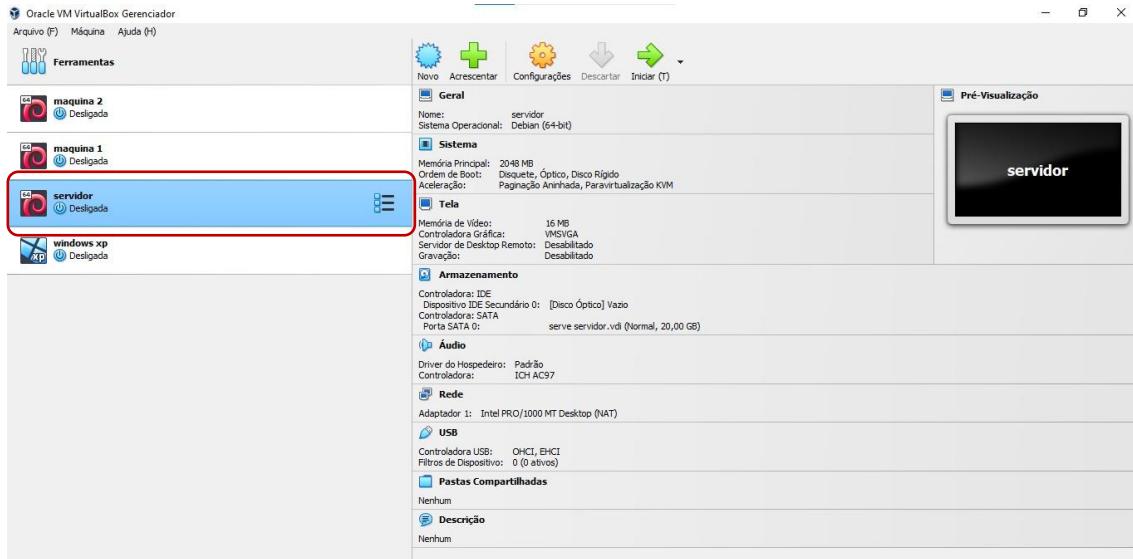


Figura 25 “Tela18”

Caso tenha colocado login, deve-se acessar com o usuário e senha que designou. O primeiro comando que vamos utilizar será o ‘ip a s’ ou ‘ip address show’. Ele é usado no sistema operacional Linux para exibir informações sobre interfaces de rede e seus endereços IP atribuídos.

```
root@debian:~# ip a s
```

Figura 26 “Tela19”

```
servidor [Executando] - Oracle VM VirtualBox
root@debian:~# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:8a:89:45 brd ff:ff:ff:ff:ff:ff
        inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
            valid_lft 86383sec preferred_lft 86383sec
            inet6 fe80::a00:27ff:fe8a:8945/64 scope link
                valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:1b:d1 brd ff:ff:ff:ff:ff:ff
4: enp0s9: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:12:9e:5d brd ff:ff:ff:ff:ff:ff
root@debian:~#
```

Figura 27 “Tela20”

Na imagem podemos verificar que todas as placas de rede estão em estado de DOWN, é um termo comumente usado em redes de computadores para indicar que uma interface de rede, como uma placa de rede ou uma conexão de rede, está desativada ou desconectada. Quando uma interface de rede está "down", ela não está operacional e não pode enviar ou receber dados pela rede. Para modificarmos e reativar o fluxo devemos digitar o seguinte comando ainda no servidor ‘ vi /etc/network/interfaces’.

```
root@debian:~# vi /etc/network/interfaces_
```

Figura 28 “Tela21”

O ‘vi’ é uma ferramenta de edição que permite que façamos alteração em arquivos ou sistemas. Após abrirmos o arquivo receberemos a seguinte imagem.

```
senhor [Executando] - Oracle VM VirtualBox
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*
# The loopback network interface
auto lo
iface lo inet loopback
# The primary network interface
auto enp0s3
iface enp0s3 inet dhcp
~
```

Figura 29 “Tela22”

Analisamos que não possui nenhuma configuração de rede nesse sistema, para alterarmos devemos selecionar a tecla ‘i’, que entra em modo de edição de texto e escrever os seguintes dados.

```
# The secundary network interface
auto enp0s8
iface enp0s8 inet static
    address 10.0.0.1
    netmask 255.255.255.0

# The tertiary network interface
auto enp0s9
iface enp0s9 inet static
    address 172.16.0.1
    netmask 255.255.255.0
~
```

Figura 30 “Tela23”

Os dados de address, netmask foram tirados do sistema em que criamos, presente na página 13. Finalizando a digitação teremos os seguintes dados na tela.

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s3
iface enp0s3 inet dhcp

# The secondary network interface
auto enp0s8
iface enp0s8 inet static
    address 10.0.0.1
    netmask 255.255.255.0

# The tertiary network interface
auto enp0s9
iface enp0s9 inet static
    address 172.16.0.1
    netmask 255.255.255.0
```

Figura 31 “Tela24”

Para salvarmos as seguintes alterações basta clicar na tecla “Esc” digitar :wq. Logo será concluída a alteração de rede do servidor, porém se novamente digitarmos o comando “ip a s” verificaremos que nossa máquina ainda estará em estado de “DOWN”

```
root@debian:~# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:8a:89:45 brd ff:ff:ff:ff:ff:ff
        inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
            valid_lft 85626sec preferred_lft 85626sec
            inet6 fe80::a00:27ff:fe8a:8945/64 scope link
                valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:1b:7a:d1 brd ff:ff:ff:ff:ff:ff
4: enp0s9: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:12:9e:5d brd ff:ff:ff:ff:ff:ff
```

Figura 32 “Tela25”

```
root@debian:~# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:8a:89:45 brd ff:ff:ff:ff:ff:ff
        inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
            valid_lft 85626sec preferred_lft 85626sec
            inet6 fe80::a00:27ff:fe8a:8945/64 scope link
                valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:1b:7a:d1 brd ff:ff:ff:ff:ff:ff
4: enp0s9: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:12:9e:5d brd ff:ff:ff:ff:ff:ff
```

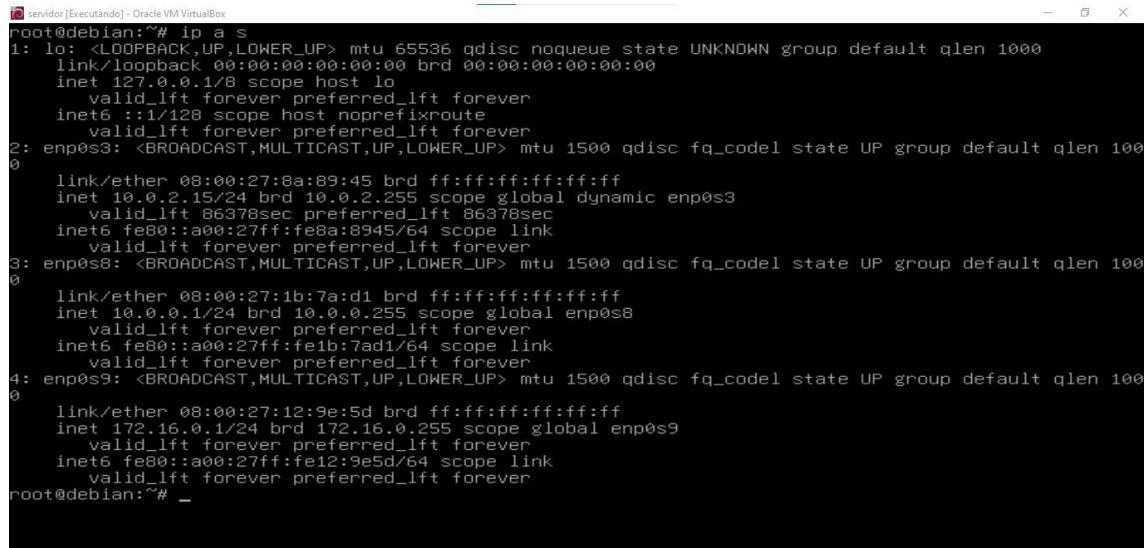
Figura 33 “Tela26”

Isso acontece pois o sistema precisa ser atualizado/reiniciado para fazer as alterações. Então para isso acontecer devemos digitar o seguinte comando “service networking restart”, ele vai reiniciar os serviços de internet do servidor.

```
root@debian:~# service networking restart
```

Figura 34“Tela27”

Logo o sistema será atualizado e se novamente digitarmos o comando “ip a s” verificaremos que as placas de redes foram ativadas.



```
senhor [Executando] - Oracle VM VirtualBox
root@debian:~# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:8a:89:45 brd ff:ff:ff:ff:ff:ff
        inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
            valid_lft 86378sec preferred_lft 86378sec
            inet6 fe80::a00:27ff:fe8a:8945/64 scope link
                valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:1b:d1:01 brd ff:ff:ff:ff:ff:ff
        inet 10.0.0.1/24 brd 10.0.0.255 scope global enp0s8
            valid_lft forever preferred_lft forever
            inet6 fe80::a00:27ff:fe1b:d1/64 scope link
                valid_lft forever preferred_lft forever
4: enp0s9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:12:9e:5d brd ff:ff:ff:ff:ff:ff
        inet 172.16.0.1/24 brd 172.16.0.255 scope global enp0s9
            valid_lft forever preferred_lft forever
            inet6 fe80::a00:27ff:fe12:9e5d/64 scope link
                valid_lft forever preferred_lft forever
root@debian:~# _
```

Figura 35 “Tela28”

Conseguimos analisar que as placas de rede saíram do estado de “DOWN”, caso alguma sua tenha permanecido, deve verificar se os comandos e alterações de texto foram digitadas corretamente.

4ºDiretriz

Após a conclusão da 3º Diretriz, temos que fazer a conexão do servidor com a máquina 1, 2 e 3(Windows). O primeiro passo é acessar a máquina 1, como root, e digitarmos novamente o primeiro comando “ip a s”.

```
root@debian:~# ip a s
```

Figura 36“Tela29”



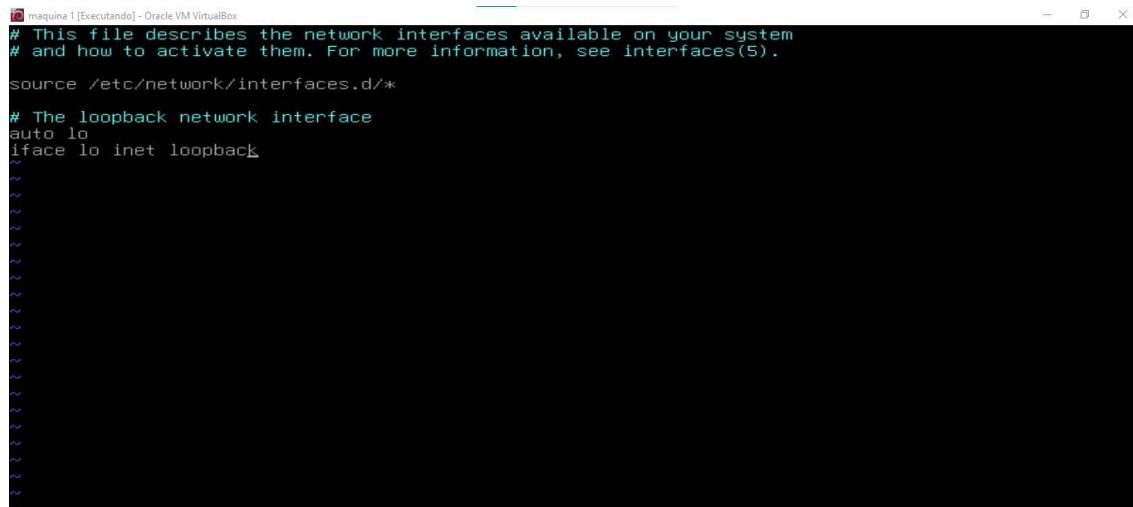
```
maquina 1 [Executando] - Oracle VM VirtualBox
root@debian:~# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:d9:2b:19 brd ff:ff:ff:ff:ff:ff
        inet6 fe80::a00:27ff:fed9:2b19/64 scope link
            valid_lft forever preferred_lft forever
root@debian:~# _
```

Figura 37“Tela30”

Assim como no servidor o estado da nossa rede está em “DOWN”. Para configurar esse sistema devemos novamente utilizar o comando “vi /etc/network/interfaces”

```
root@debian:~# vi /etc/network/interfaces_
```

Figura 38“Tela31”



```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s3
iface enp0s3 inet static
    address 10.0.0.2
    netmask 255.255.255.0
    gateway 10.0.0.1
```

Figura 39“Tela32”

Entramos no arquivo de edição de rede da máquina 1, para configurarmos o seu sistema devemos ir com o percursor até o final “loopback”, clicar na letra “i”, dar enter e digitar os seguintes comandos.

```
root@maquina1:~# vi /etc/network/interfaces
```

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
```

```
source /etc/network/interfaces.d/*
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s3
iface enp0s3 inet static
    address 10.0.0.2
    netmask 255.255.255.0
    gateway 10.0.0.1
```

Figura 40“Tela33”

Na máquina 1 usamos o gateway e netmask para facilitar a comunicação entre os dispositivos. O gateway é o ponto de entrada e saída de uma rede que conecta diferentes redes ou sub-redes, já a netmask é usada para determinar qual parte de um endereço IP pertence à identificação da rede e qual parte pertence à identificação do host. Para salvarmos as seguintes alterações basta clicar na tecla “Esc” digitar :wq. Logo será concluída a alteração de rede do servidor, porém se novamente digitarmos o comando “ip a s” verificaremos que nossa máquina ainda estará em estado de “DOWN”.

```
root@debian:~# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
            valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:d9:2b:19 brd ff:ff:ff:ff:ff:ff
root@debian:~# _
```

Figura 41“Tela34”

Isso acontece pois o sistema precisa ser atualizado/reiniciado para fazer as alterações. Então para isso acontecer devemos digitar o seguinte comando “service networking restart”, ele vai reiniciar os serviços de internet da máquina.

```
root@debian:~# service networking restart
```

Figura 42“Tela35”

Logo o sistema será atualizado e se novamente digitarmos o comando “ip a s” verificaremos que as placas de redes foram ativadas.

```
root@debian:~# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
            valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:d9:2b:19 brd ff:ff:ff:ff:ff:ff
        inet 10.0.0.2/24 brd 10.0.0.255 scope global enp0s3
            valid_lft forever preferred_lft forever
        inet6 fe80::a00:27ff:fed9:2b19/64 scope link
            valid_lft forever preferred_lft forever
root@debian:~# _
```

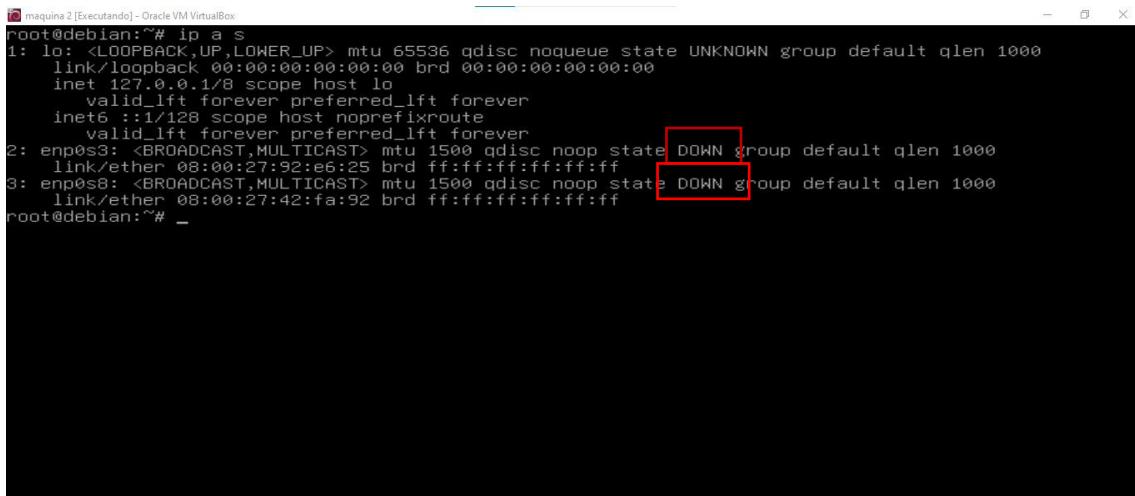
Figura 43“Tela36”

Conseguimos analisar que as placas de rede saíram do estado de “DOWN”, caso alguma sua tenha permanecido, deve verificar se os comandos e alterações de texto foram digitadas corretamente.

Na máquina 2, novamente verificaremos o status que a placa de rede de encontra, então novamente utilizaremos o comando “ip a s”

```
root@debian:~# ip a s
```

Figura 44“Tela37”



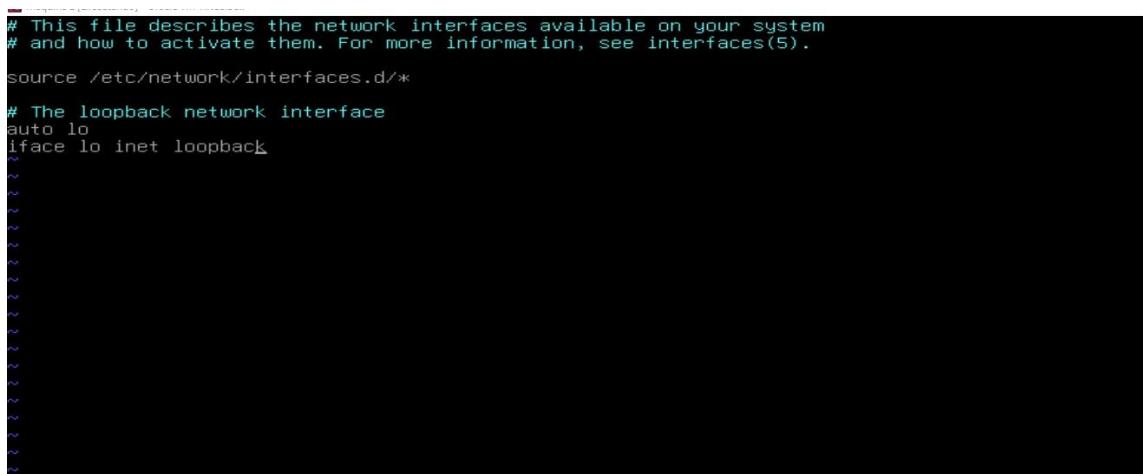
```
root@debian:~# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:92:e6:25 brd ff:ff:ff:ff:ff:ff
3: enp0s8: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:42:fa:92 brd ff:ff:ff:ff:ff:ff
root@debian:~# _
```

Figura 45“Tela38”

Conseguimos visualizar que novamente a máquina no seu início está em estado de “DOWN”, na máquina 2 temos duas conexões devido ela estar ligado a um servidor e a uma máquina Windows, para alterar este estado devemos digitar o comando “vi /etc/network/interfaces”.

```
root@debian:~# vi /etc/network/interfaces_
```

Figura 46“Tela39”

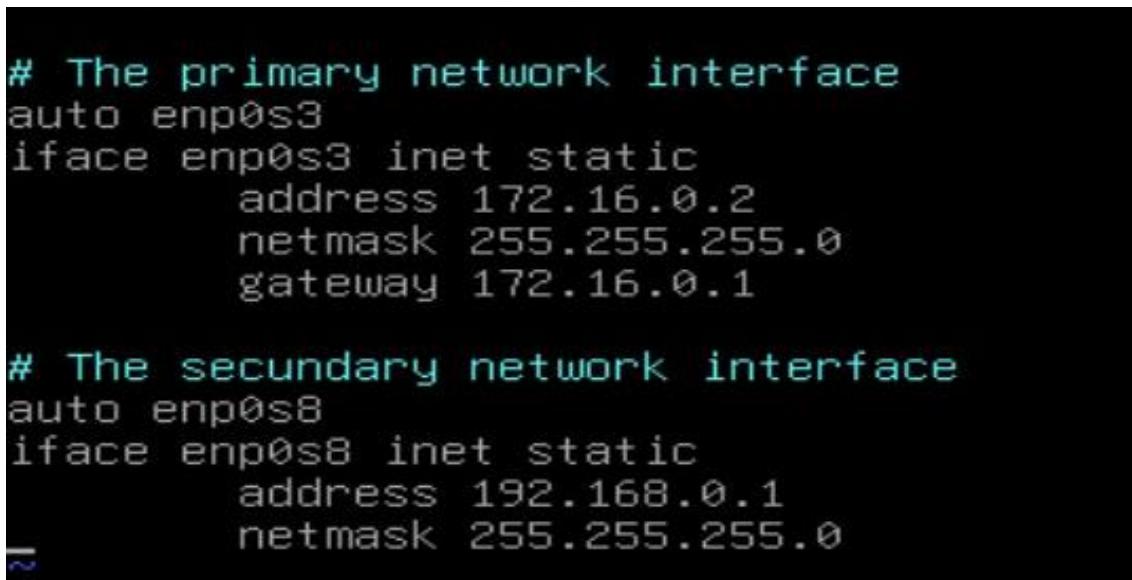


```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*
# The loopback network interface
auto lo
iface lo inet loopback
```

Figura 47“Tela40”

Ele nos levará novamente em uma tela de edição de sistema e para configurarmos a rede de ambas as conexões devemos configurarmos o seu sistema, devemos ir com o precursor até o final “loopback”, clicar na letra “i”, dar enter e digitar seguintes configurações.

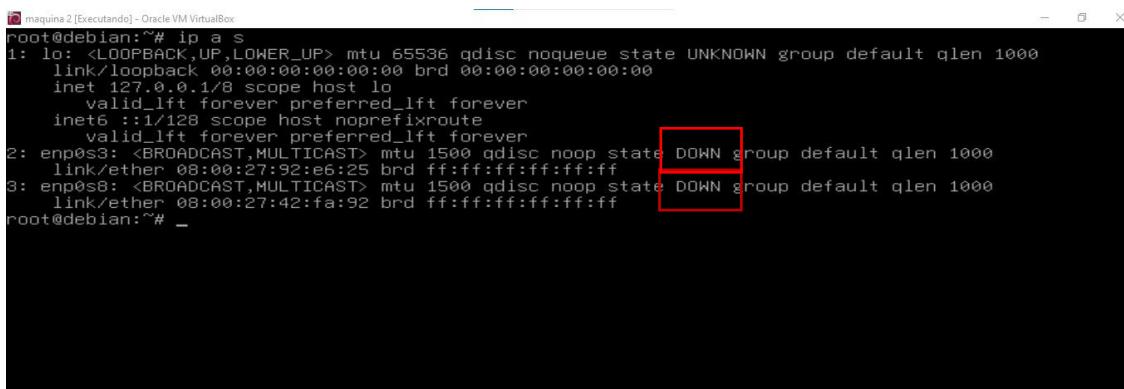


```
# The primary network interface
auto enp0s3
iface enp0s3 inet static
    address 172.16.0.2
    netmask 255.255.255.0
    gateway 172.16.0.1

# The secundary network interface
auto enp0s8
iface enp0s8 inet static
    address 192.168.0.1
    netmask 255.255.255.0
```

Figura 48“Tela41”

Para salvarmos as seguintes alterações basta clicar na tecla “Esc.” digitar :wq. Logo será concluída a alteração de rede do servidor, porém se novamente digitarmos o comando “ip a s” verificaremos que nossa máquina ainda estará em estado de “DOWN”.



```
root@debian:~# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
            valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:92:e6:25 brd ff:ff:ff:ff:ff:ff
3: enp0s8: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:42:fa:92 brd ff:ff:ff:ff:ff:ff
root@debian:~# _
```

Figura 49“Tela42”

Isso acontece pois o sistema precisa ser atualizado/reiniciado para fazer as alterações. Então para isso acontecer devemos digitar o seguinte comando “service networking restart”, ele vai reiniciar os serviços de internet da máquina.

```
root@debian:~# service networking restart
```

Figura 50“Tela43”

Logo o sistema será atualizado e se novamente digitarmos o comando “ip a s” verificaremos que as placas de redes foram ativadas.

```

root@debian:~# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
            valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:92:e6:25 brd ff:ff:ff:ff:ff:ff
        inet 172.16.0.2/24 brd 172.16.0.255 scope global enp0s3
            valid_lft forever preferred_lft forever
        inet6 fe80::a00:27ff:fe92:e625/64 scope link
            valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:42:fa:92 brd ff:ff:ff:ff:ff:ff
        inet 192.168.0.1/24 brd 192.168.0.255 scope global enp0s8
            valid_lft forever preferred_lft forever
        inet6 fe80::a00:27ff:fe42:fa92/64 scope link
            valid_lft forever preferred_lft forever
root@debian:~# 

```

Figura 51“Tela44”

Conseguimos analisar que as placas de rede saíram do estado de “DOWN”, caso alguma sua tenha permanecido, deve verificar se os comandos e alterações de texto foram digitadas corretamente.

Teste

Para testar a conectividade deles com o servidor vamos utilizar o comando Ping. O “ping” é uma ferramenta de rede utilizada para testar a conectividade entre dois dispositivos através de uma rede IP (Internet Protocol). Quando você envia um comando "ping" para um determinado endereço IP ou nome de domínio, seu computador envia pacotes de dados para o destino especificado e aguarda por uma resposta. Se o destino estiver acessível e configurado para responder ao ping, você receberá uma resposta, indicando que a conexão está ativa e funcionando.

Se digitarmos ping na máquina 2 com o ip do servidor podemos ver que existe a conectividade entre elas.

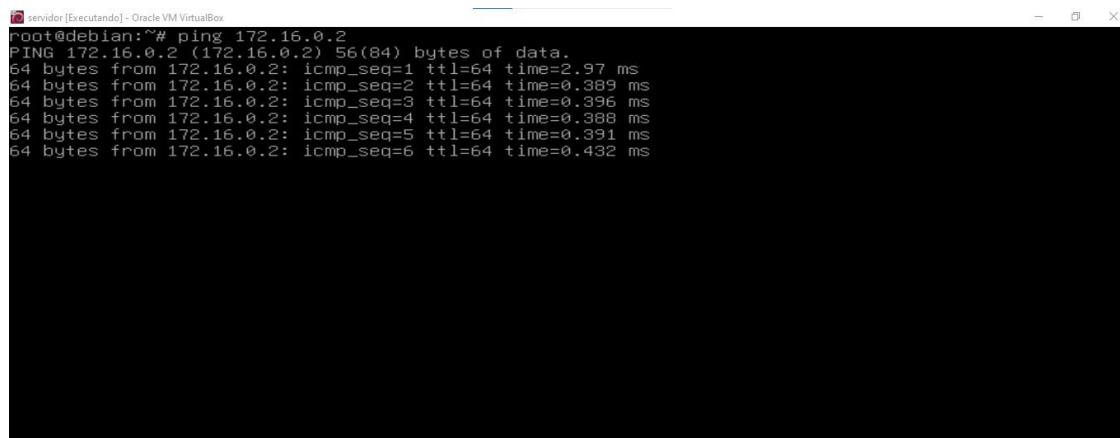
```

root@debian:~# service networking restart
root@debian:~# ping 172.16.0.1
PING 172.16.0.1 (172.16.0.1) 56(84) bytes of data.
64 bytes from 172.16.0.1: icmp_seq=1 ttl=64 time=0.431 ms
64 bytes from 172.16.0.1: icmp_seq=2 ttl=64 time=0.401 ms
64 bytes from 172.16.0.1: icmp_seq=3 ttl=64 time=0.392 ms
64 bytes from 172.16.0.1: icmp_seq=4 ttl=64 time=0.383 ms
64 bytes from 172.16.0.1: icmp_seq=5 ttl=64 time=0.390 ms
64 bytes from 172.16.0.1: icmp_seq=6 ttl=64 time=0.398 ms
64 bytes from 172.16.0.1: icmp_seq=7 ttl=64 time=0.395 ms
64 bytes from 172.16.0.1: icmp_seq=8 ttl=64 time=0.390 ms
64 bytes from 172.16.0.1: icmp_seq=9 ttl=64 time=0.394 ms
64 bytes from 172.16.0.1: icmp_seq=10 ttl=64 time=0.399 ms
64 bytes from 172.16.0.1: icmp_seq=11 ttl=64 time=0.471 ms

```

Figura 52“Tela45”

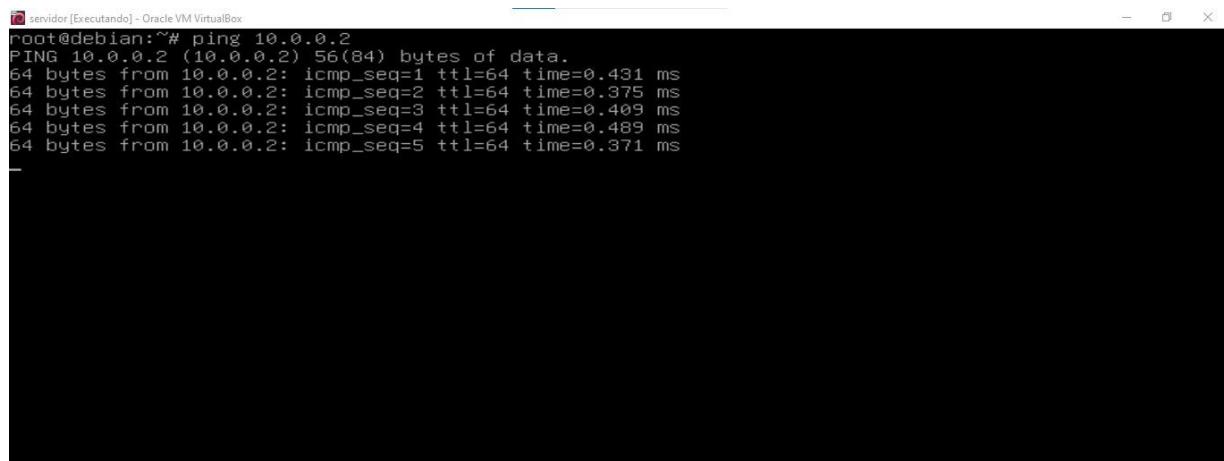
Caso não tenha dado ping, verifique as configurações se estão de acordo. Agora verificaremos o ping da máquina 2 dentro do servidor para ver se ela responde ao servidor.



```
root@debian:~# ping 172.16.0.2
PING 172.16.0.2 (172.16.0.2) 56(84) bytes of data.
64 bytes from 172.16.0.2: icmp_seq=1 ttl=64 time=2.97 ms
64 bytes from 172.16.0.2: icmp_seq=2 ttl=64 time=0.389 ms
64 bytes from 172.16.0.2: icmp_seq=3 ttl=64 time=0.396 ms
64 bytes from 172.16.0.2: icmp_seq=4 ttl=64 time=0.388 ms
64 bytes from 172.16.0.2: icmp_seq=5 ttl=64 time=0.391 ms
64 bytes from 172.16.0.2: icmp_seq=6 ttl=64 time=0.432 ms
```

Figura 53“Tela46”

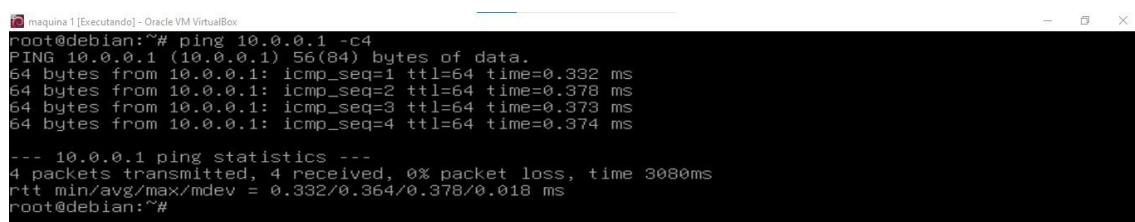
Vemos que a conectividade tanto de envio como recebimento de pacote entre elas está funcionando. Agora dentro do servidor vamos dar o ping da máquina 1 para ele.



```
root@debian:~# ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.431 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.375 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.409 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.489 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.371 ms
```

Figura 54“Tela47”

Depois de vermos que o servidor está recebendo pacotes da máquina 1, verificaremos se ela recebe do servidor, novamente utilizaremos o comando ping com o ip do servidor na máquina 1.



```
root@debian:~# ping 10.0.0.1 -c4
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.332 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.378 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.373 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=0.374 ms

--- 10.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3080ms
rtt min/avg/max/mdev = 0.332/0.364/0.378/0.018 ms
root@debian:~#
```

Figura 55“Tela48”

Após verificar que a conectividade entre elas existe vamos começar a configuração da máquina Windows, caso algum ping esteja dando errado, não demonstrando sinal de conectividade deve-se repetir as operações e verificar possíveis erros.

5º Diretriz

Entrando na máquina Windows podemos ver que a máquina não está conectada. Para configurá-la devemos acessar o painel de controle conforme o passo-a-passo abaixo.



Figura 56“Tela49”

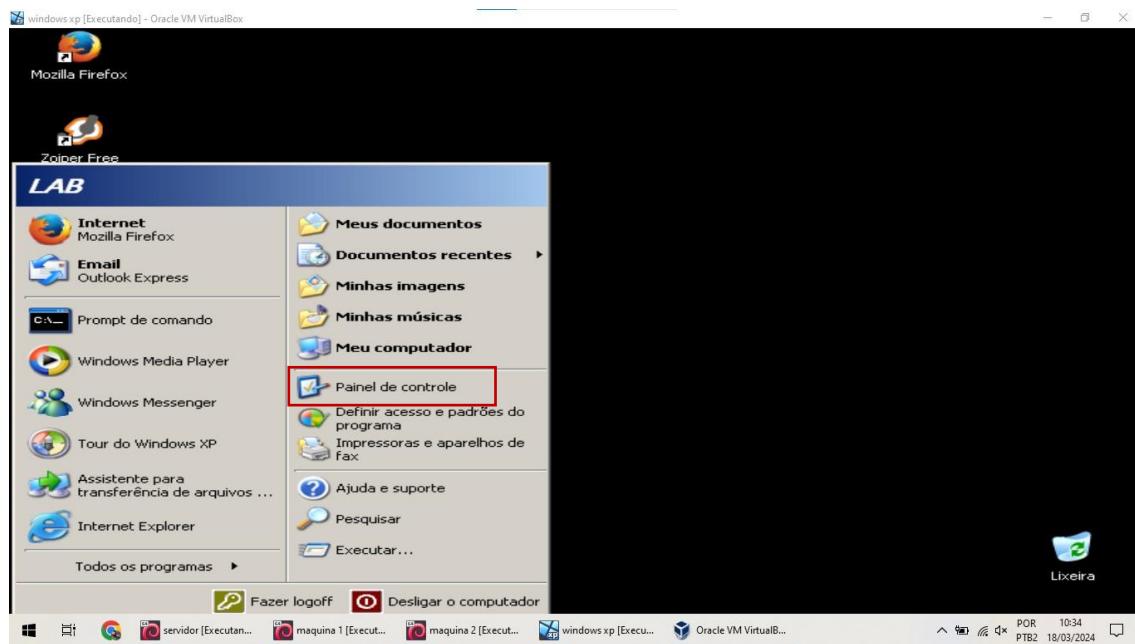


Figura 57“Tela50”

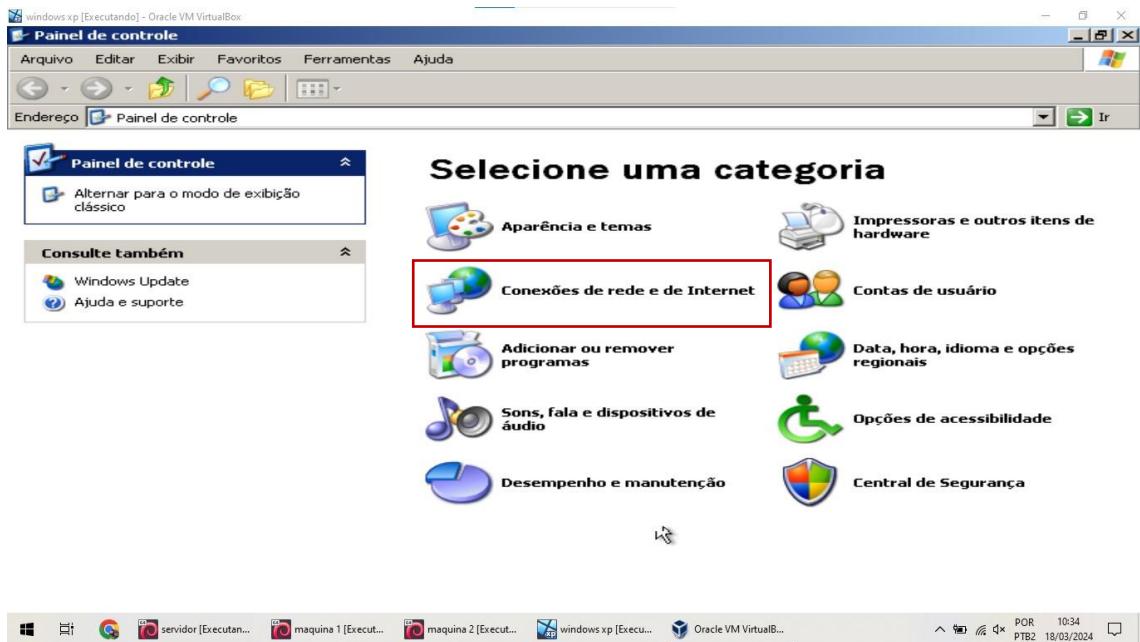


Figura 58“Tela51”

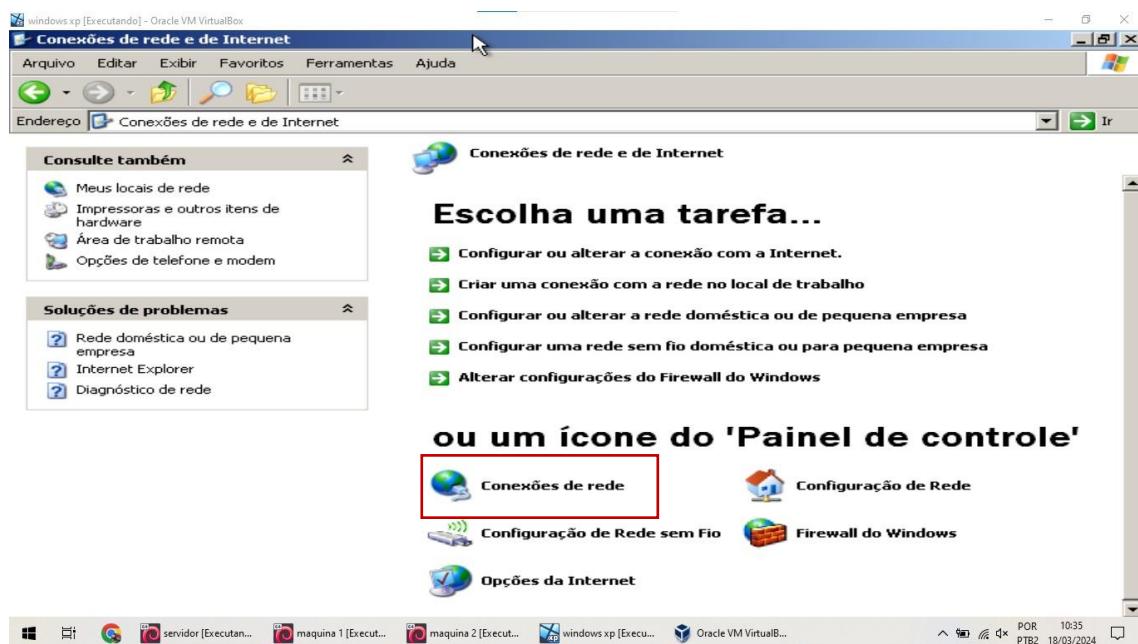


Figura 59“Tela52”

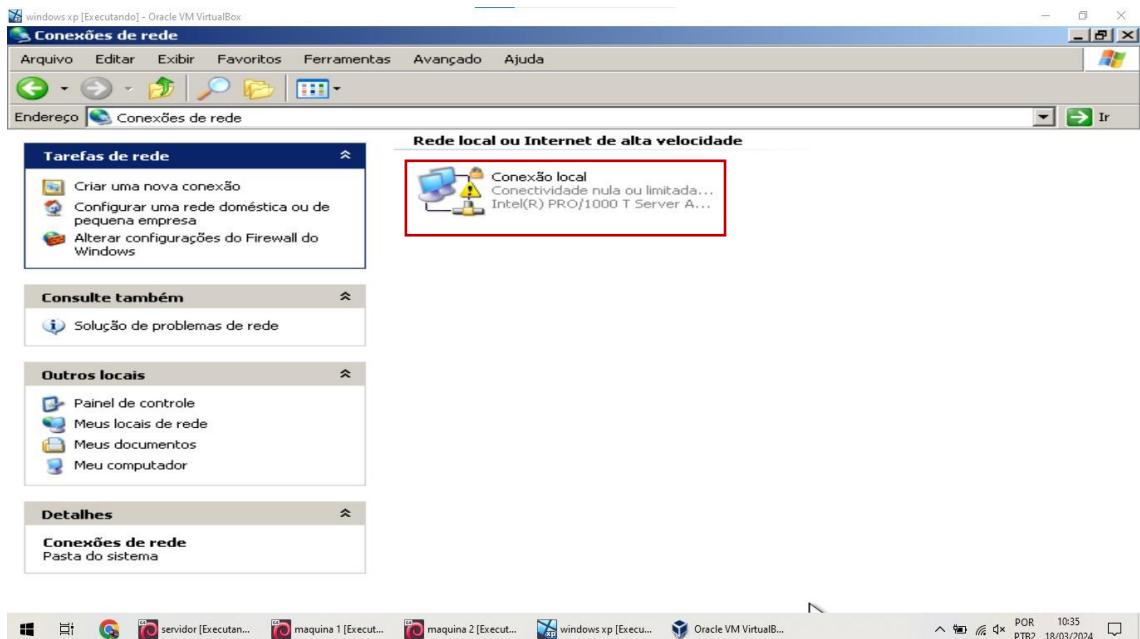


Figura 60“Tela53”

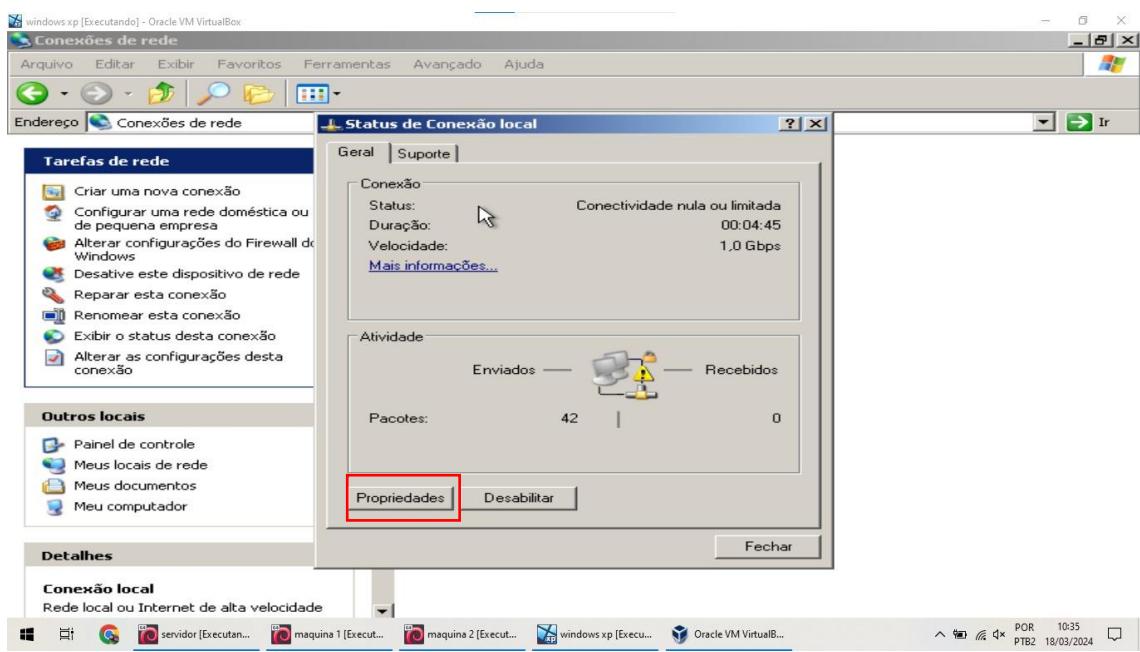


Figura 61“Tela54”

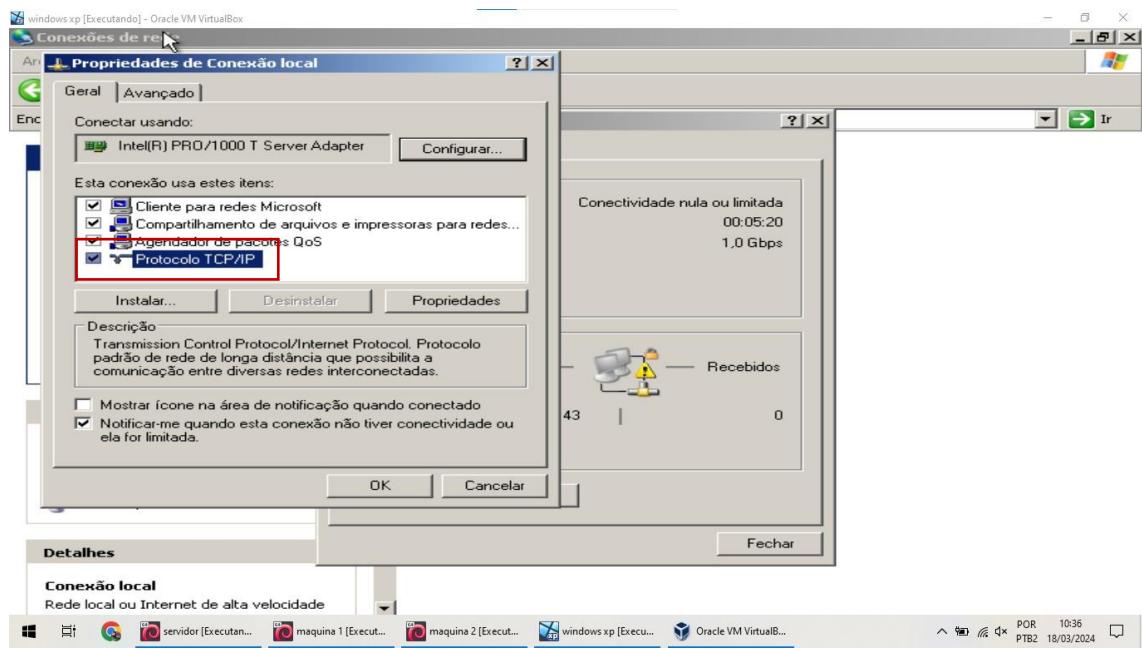


Figura 62“Tela55”

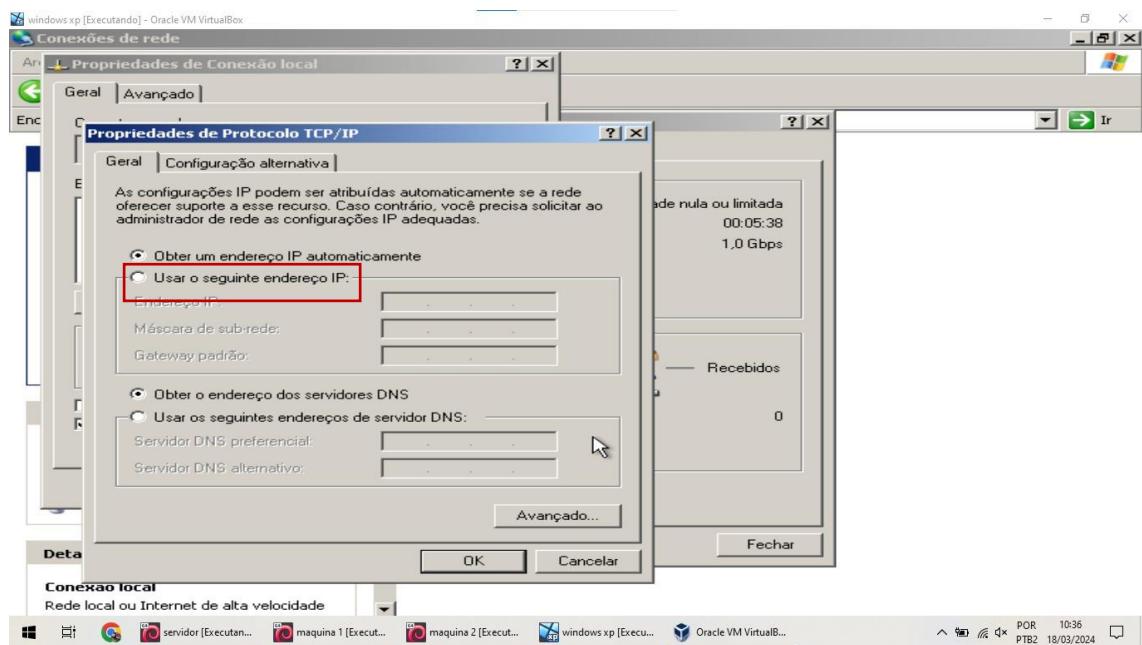


Figura 63“Tela56”

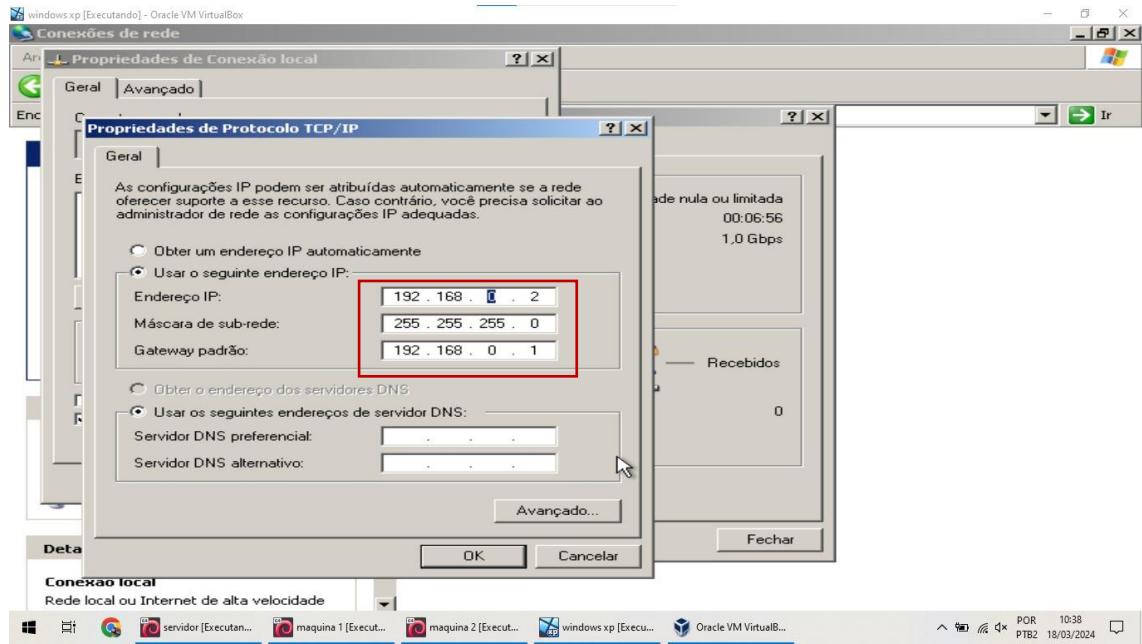


Figura 64“Tela57”

Aqui colocaremos os ips fornecidos conforme o organograma fornecido na página 13. Para finalizar e guardar as informações basta clicar em Ok.

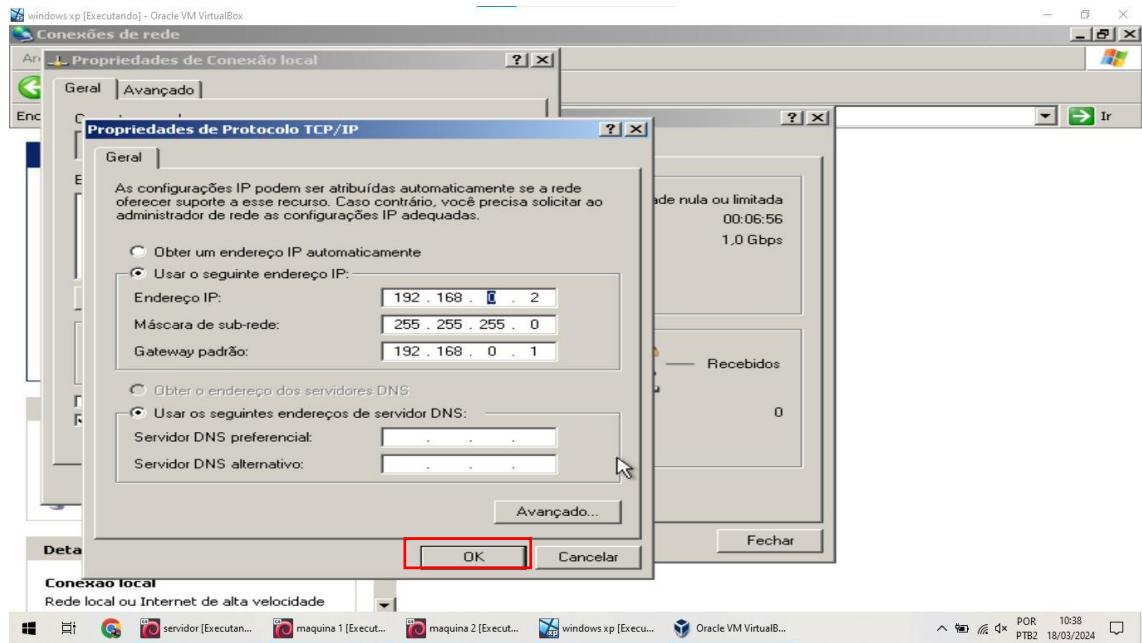


Figura 65“Tela58”

Agora vamos acessar o CMD do Windows para realizar o teste se o Windows está recebendo pacotes do servidor. No Prompt de Comando, digite o comando PING juntamente o IP da máquina 3.

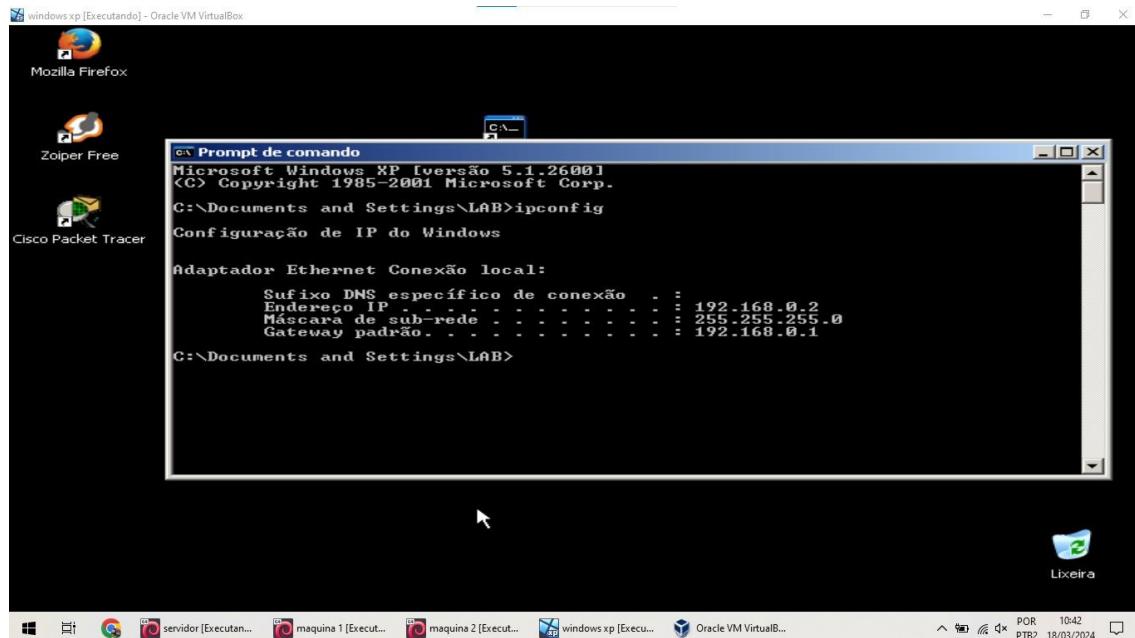


Figura 66“Tela59”

Observamos que o Windows não consegue receber pacotes da máquina 3 devido à restrição imposta pelo firewall do Windows, que impede a recepção de pacotes de certas fontes enquanto ativado. Para desativá-lo, siga as instruções abaixo.

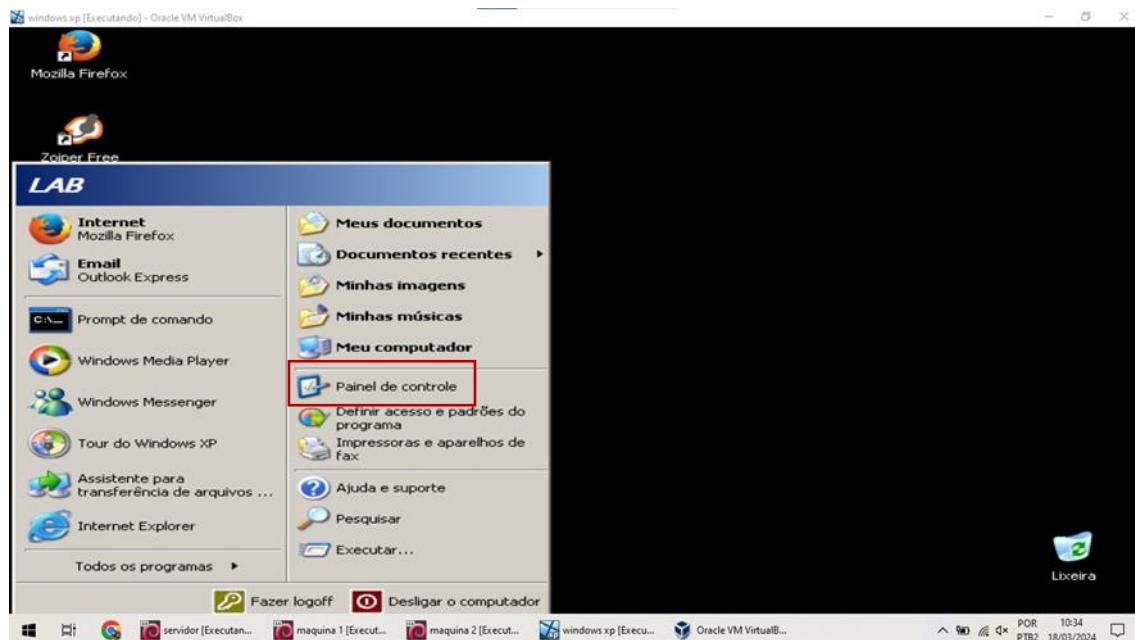


Figura 67“Tela60”

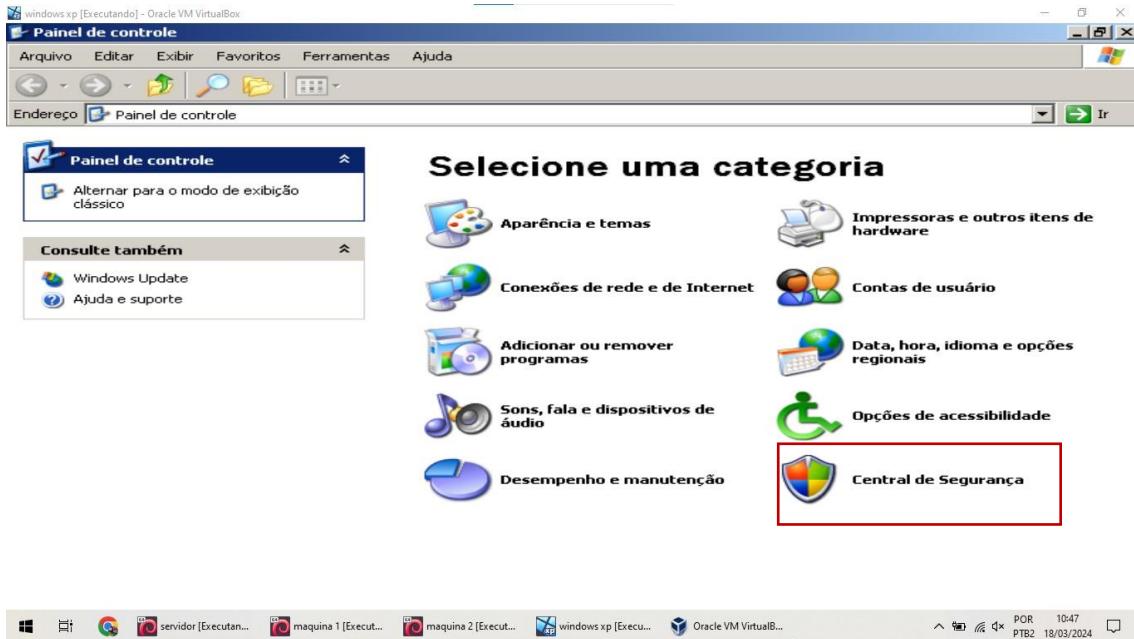


Figura 68“Tela61”



Figura 69“Tela62”

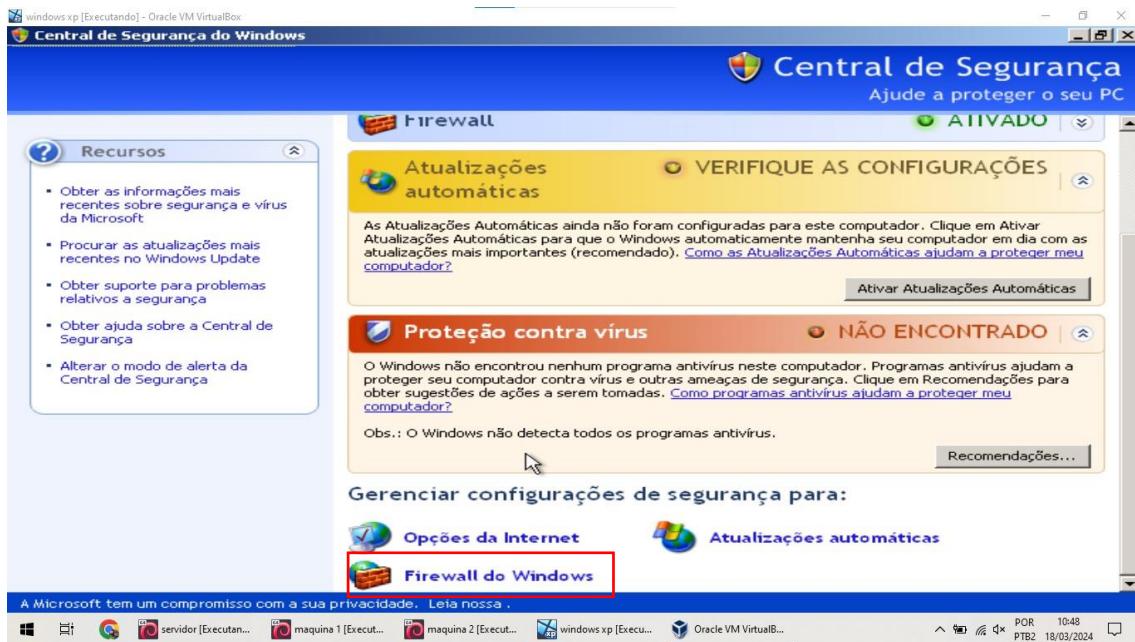


Figura 70“Tela63”

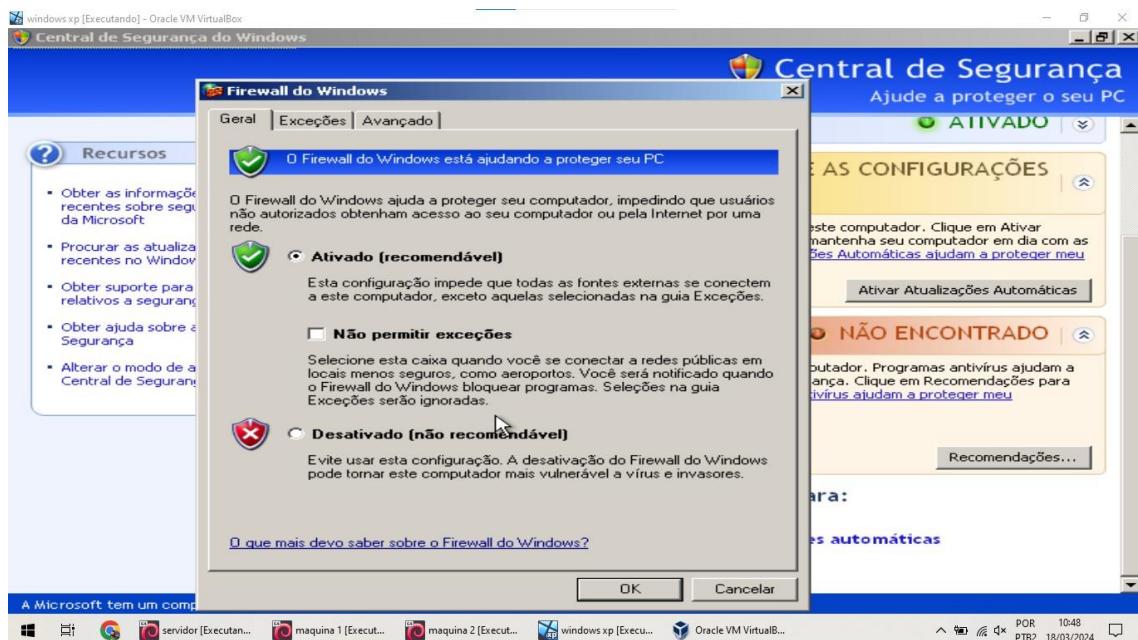


Figura 71“Tela64”

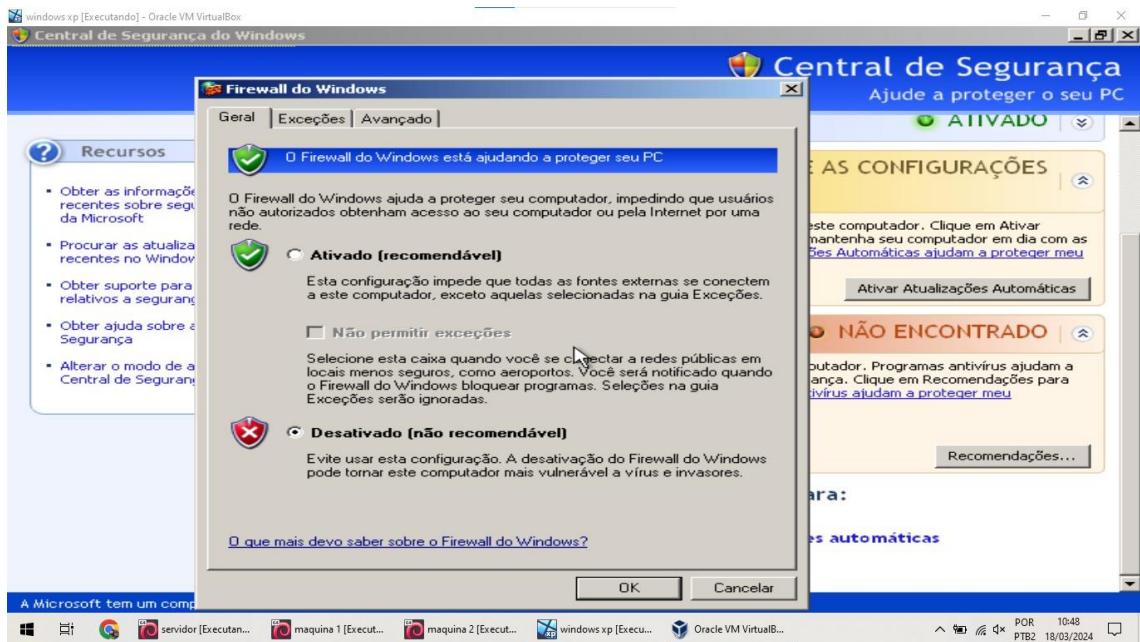


Figura 72“Tela65”

Se revisarmos o prompt de comando e inserirmos o comando PING seguido do IP da máquina 3, receberemos respostas.

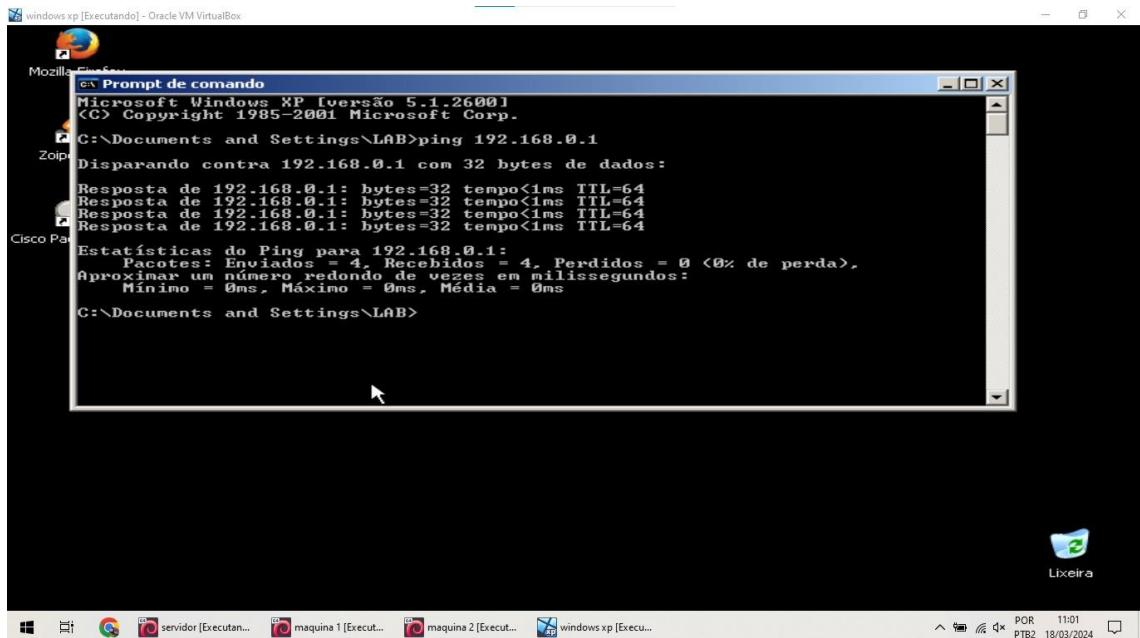
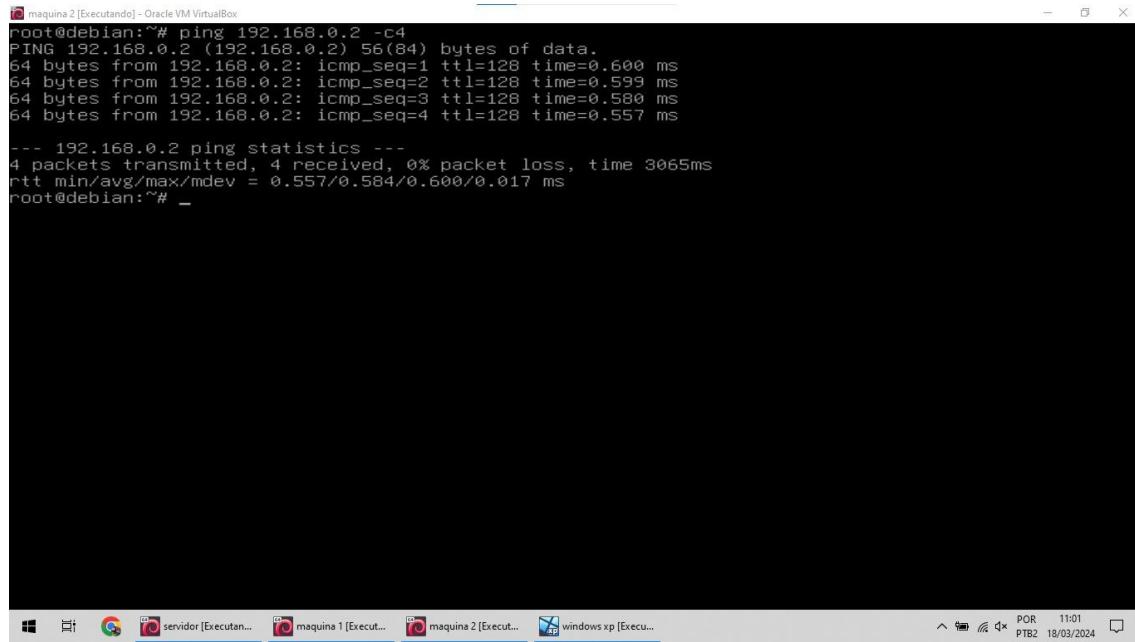


Figura 73”Tela66”

Se entrarmos na máquina 3 e tentar dar o ‘Ping’ da máquina Windows, também vamos ver que elas conversem entre elas



```
maquina 2 [Executando] - Oracle VM VirtualBox
root@debian:~# ping 192.168.0.2 -c4
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
64 bytes from 192.168.0.2: icmp_seq=1 ttl=128 time=0.600 ms
64 bytes from 192.168.0.2: icmp_seq=2 ttl=128 time=0.599 ms
64 bytes from 192.168.0.2: icmp_seq=3 ttl=128 time=0.580 ms
64 bytes from 192.168.0.2: icmp_seq=4 ttl=128 time=0.557 ms

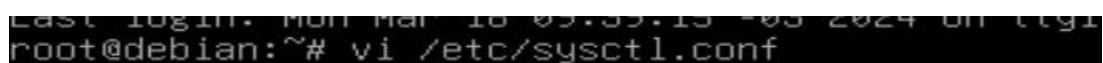
--- 192.168.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3065ms
rtt min/avg/max/mdev = 0.557/0.584/0.600/0.017 ms
root@debian:~# _
```

Figura 74” Tela67”

Caso não tenhas obtido a resposta, verifique possíveis erros refazendo os comandos.

6º Diretriz

Se tentarmos fazer o comando Ping da máquina 1 para a Windows, pode-se observar que a máquina não irá receber pacotes. Para traçar a conexão entre elas devemos fazer os seguintes passos. Primeiro vamos digitar o comando vi /etc/sysctl.conf, na máquina servidor.



```
Last login: Mon Mar 18 09:39:13 -03 2024 on ttys0
root@debian:~# vi /etc/sysctl.conf
```

Figura 75” Tela68”

Após, receberemos a seguinte tela.

```

# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.

#
#kernel.domainname = example.com

# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3

#####
# Functions previously found in netbase
# 

# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

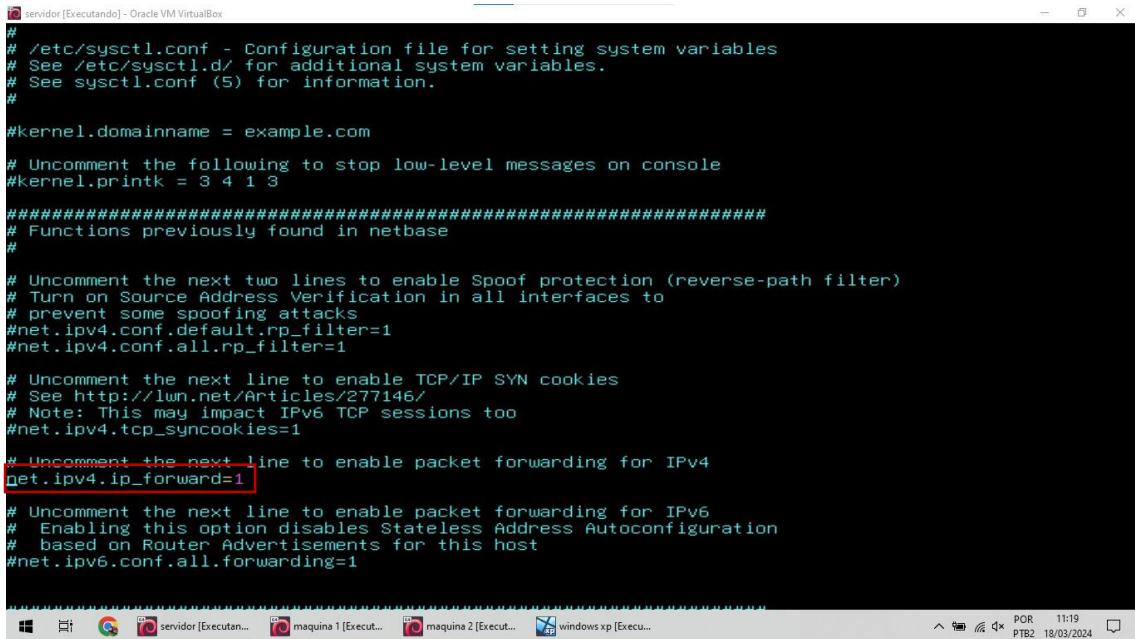
# Uncomment the next line to enable packet forwarding for IPv4
#net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1

```

Figura 76” Tela69”

Aqui entramos nas configurações padrões de rede, podemos ver que o IPv4 o qual estamos utilizando está comentado (#), pra ativar esse recurso temos que apertar a tecla i que entra em modo de edição de texto, ir até a linha #net.ipv4.ip_forward=1, e tirar a # do texto.



```

# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.

#
#kernel.domainname = example.com

# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3

#####
# Functions previously found in netbase
# 

# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
#net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1

```

Figura 77” Tela70”

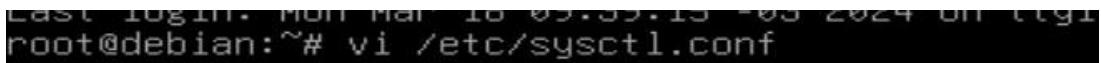
Para guardar informações bata apertar a tecla Esc digitar :wq. Se tentarmos dar novamente o comando ping não irá funcionar, pois precisamos que o sistema reinicie então para reiniciar esse sistema, devemos digitar o seguinte comando ainda na máquina servidor.



```
root@debian:~# sysctl -p
```

Figura 78” Tela71”

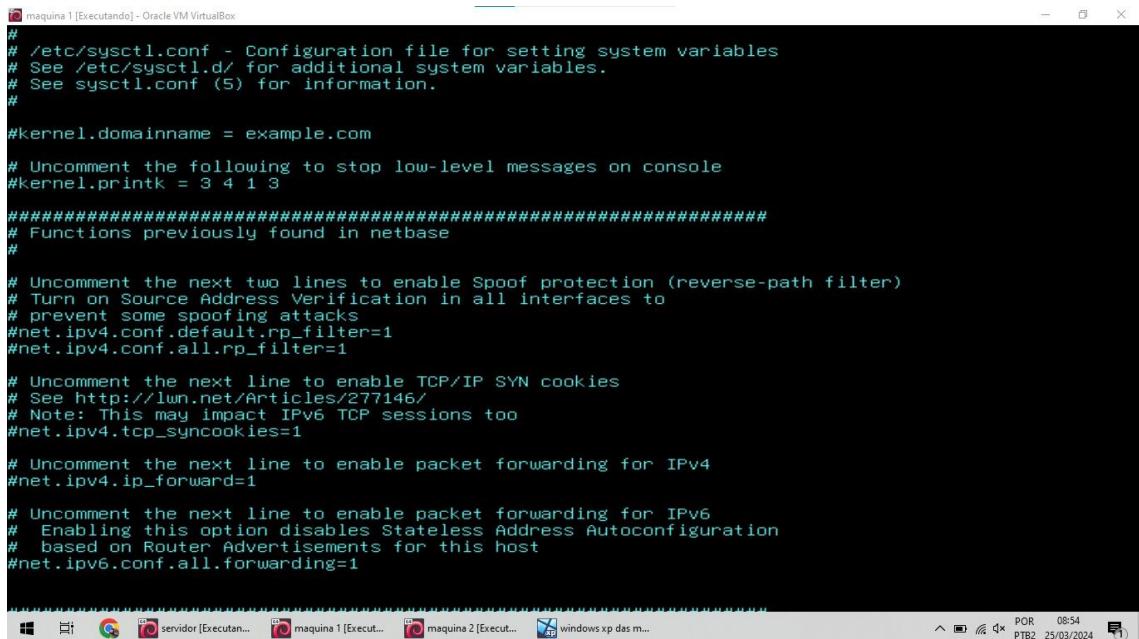
Assim o sistema será reiniciado. Na máquina 1 temos que fazer os mesmos procedimentos, para conseguirmos a chave de conexão dela. Acessando a máquina 1, vamos usar novamente o comando vi /etc/sysctl.conf.



```
Last login: Mon Mar 18 09:35:15 -03 2024 on ttys0
root@debian:~# vi /etc/sysctl.conf
```

Figura 79” Tela72”

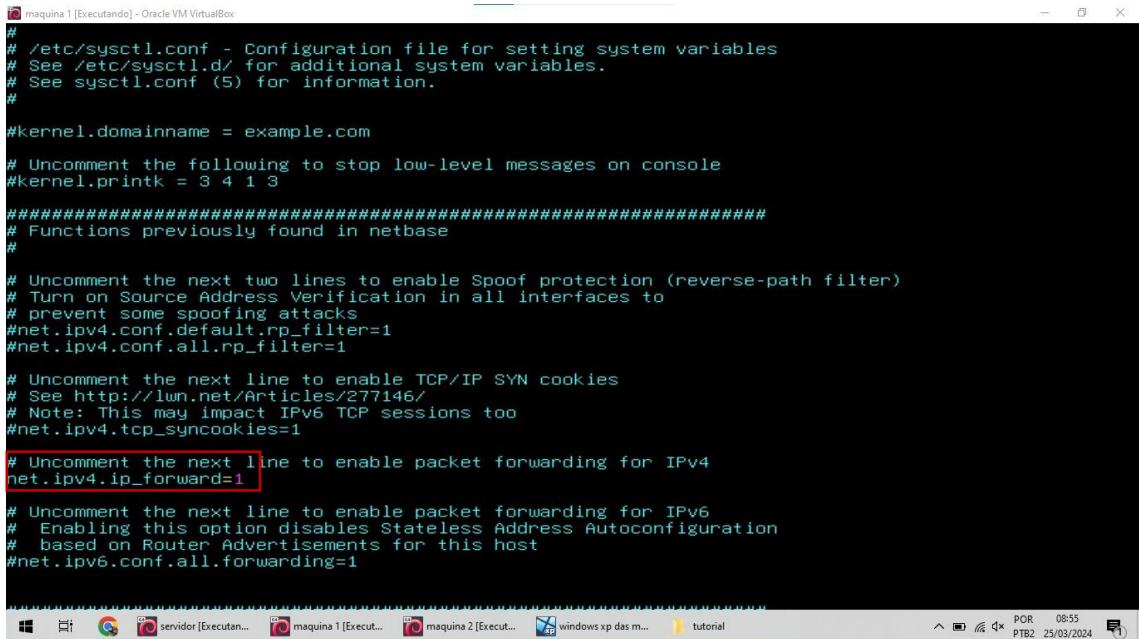
Após receberemos a seguinte imagem.



```
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.
#
#kernel.domainname = example.com
# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3
#####
# Functions previously found in netbase
#
# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1
#
# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1
#
# Uncomment the next line to enable packet forwarding for IPv4
#net.ipv4.ip_forward=1
#
# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1
```

Figura 80” Tela73”

Podemos analisar que a mesma configuração do servidor está comentada na máquina 1. E novamente devemos fazer sua alteração.



```
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.
#
#kernel.domainname = example.com
# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 8
#####
# Functions previously found in netbase
#
# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1
#
# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1
#
# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1
#
# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1
```

Figura 81” Tela74”

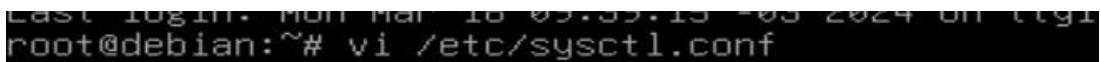
Para guardar temo que apertar a tecla Esc, digitar :wq. Novamente o sistema precisará ser reiniciado, para isso digitar o comando `sysctl -p`.



```
root@debian:~# sysctl -p
```

Figura 82” Tela75”

Na máquina dois temos que traçar os mesmos passos que seguimos na máquina 1 e servidor. Acessando vamos utilizar o comando `vi /etc/sysctl.conf`.



```
Last login: Mon Mar 18 03:33:15 -03 2024 on ttys0
root@debian:~# vi /etc/sysctl.conf
```

Figura 83” Tela76”

Após receberemos a seguinte imagem.

```

# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.

#
#kernel.domainname = example.com

# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3

#####
# Functions previously found in netbase
#

# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
#net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1

```

Figura 84” Tela77”

Podemos analisar que a mesma configuração do servidor e da máquina 1 está presente na máquina 2. E novamente devemos fazer sua alteração.

```

# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.

#
#kernel.domainname = example.com

# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3

#####
# Functions previously found in netbase
#

# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
#net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1

```

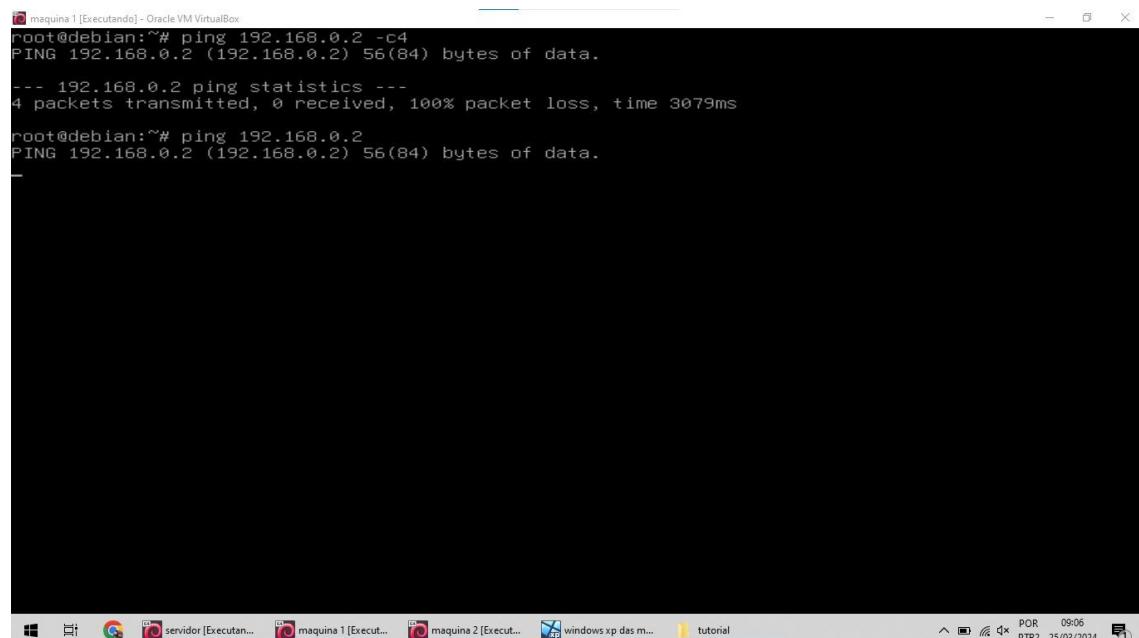
Figura 85” Tela78”

Para guardar temo que apertar a tecla Esc, digitar :wq. Novamente o sistema precisará ser reiniciado, para isso digitar o comando sysctl -p.

```
root@debian:~# sysctl -p
```

Figura 86” Tela79”

Se novamente tentarmos utilizar o comando ping com ip da Windows dentro da máquina 1, vamos perceber que a máquina ainda não está recebendo pacotes.



```
maquina 1 [Executando] - Oracle VM VirtualBox
root@debian:~# ping 192.168.0.2 -c4
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
--- 192.168.0.2 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3079ms
root@debian:~# ping 192.168.0.2
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
```

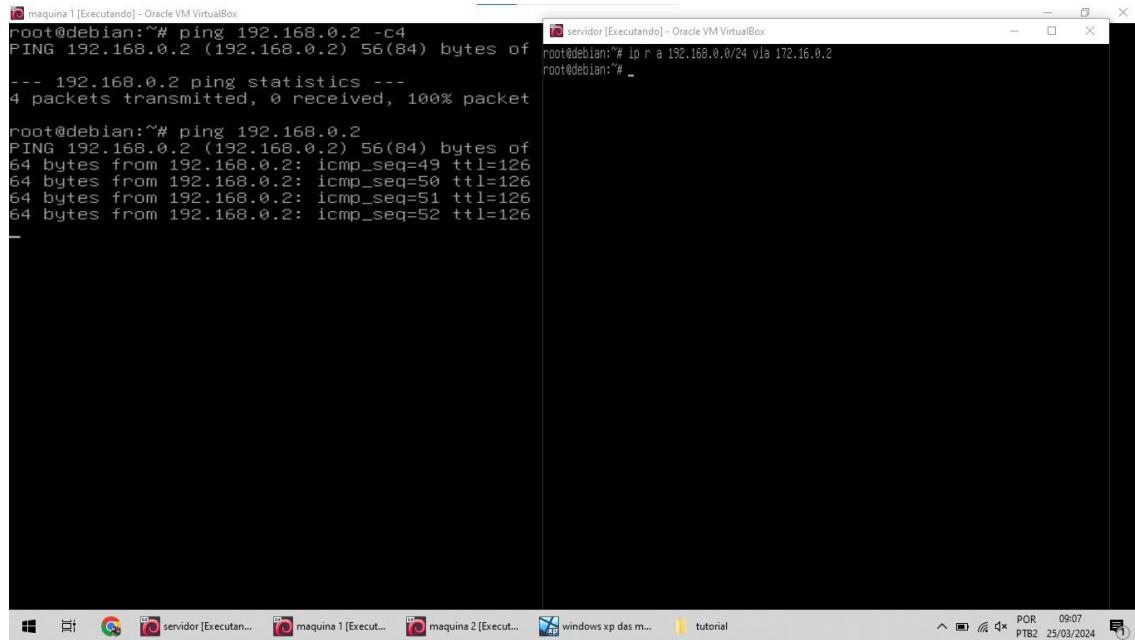
Figura 87” Tela80”

Isso acontece pois ainda precisamos traçar a via entre eles, o meio de conexão. Para isso no servidor temos que simplesmente digitar o comando “ip a r (ip da máquina 2 e todas que estão conectadas nela) via (ip da máquina servidor).

```
servidor [Executando] - Oracle VM VirtualBox
root@debian:~# ip r a 192.168.0.0/24 via 172.16.0.2
```

Figura 88” Tela81”

Na hora que traçarmos esse caminho, a máquina 1 conseguirá receber pacotes da Windows.

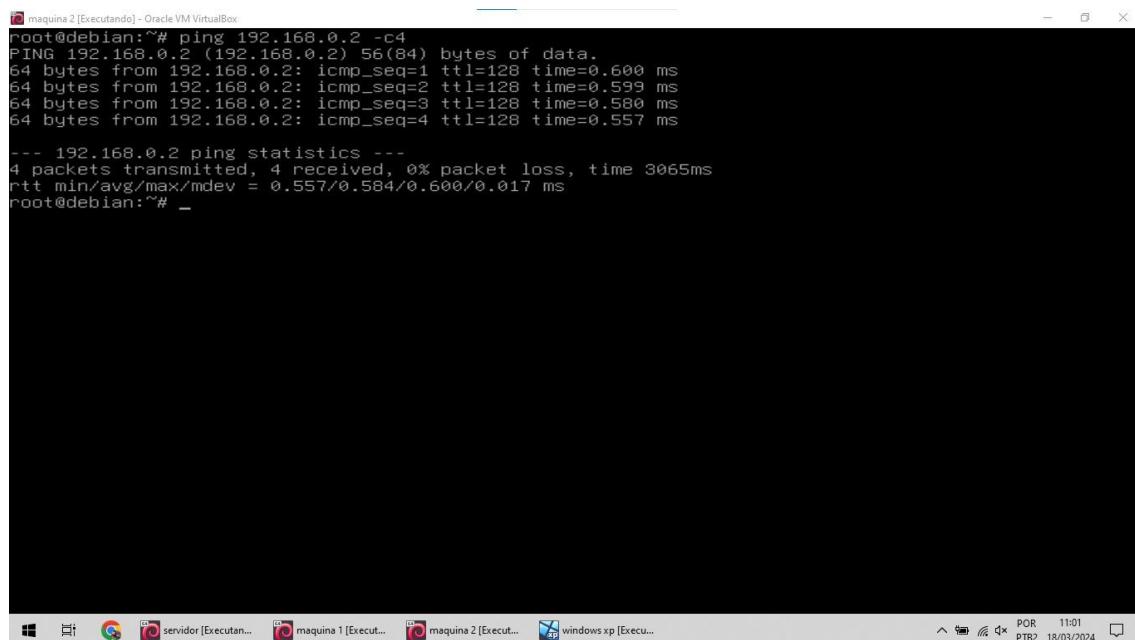


```
maquina 1 [Executando] - Oracle VM VirtualBox
root@debian:~# ping 192.168.0.2 -c4
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of
--- 192.168.0.2 ping statistics ---
4 packets transmitted, 0 received, 100% packet
root@debian:~# ping 192.168.0.2
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of
64 bytes from 192.168.0.2: icmp_seq=49 ttl=126
64 bytes from 192.168.0.2: icmp_seq=50 ttl=126
64 bytes from 192.168.0.2: icmp_seq=51 ttl=126
64 bytes from 192.168.0.2: icmp_seq=52 ttl=126

servidor [Executando] - Oracle VM VirtualBox
root@debian:~# ip r a 192.168.0.0/24 via 172.16.0.2
root@debian:~#
```

Figura 89” Tela82”

Na máquina 2 também podemos verificar que está conseguindo receber pacotes dela, assim como o servidor também está conseguindo.



```
maquina 2 [Executando] - Oracle VM VirtualBox
root@debian:~# ping 192.168.0.2 -c4
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
64 bytes from 192.168.0.2: icmp_seq=1 ttl=128 time=0.600 ms
64 bytes from 192.168.0.2: icmp_seq=2 ttl=128 time=0.599 ms
64 bytes from 192.168.0.2: icmp_seq=3 ttl=128 time=0.580 ms
64 bytes from 192.168.0.2: icmp_seq=4 ttl=128 time=0.557 ms
--- 192.168.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3065ms
rtt min/avg/max/mdev = 0.557/0.584/0.600/0.017 ms
root@debian:~#
```

Figura 90” Tela83”

Na máquina Windows, tentaremos ver a conexão dela com a máquina 1, para isso é digitarmos o comando Ping no cmd juntamente ao ip da máquina 1.

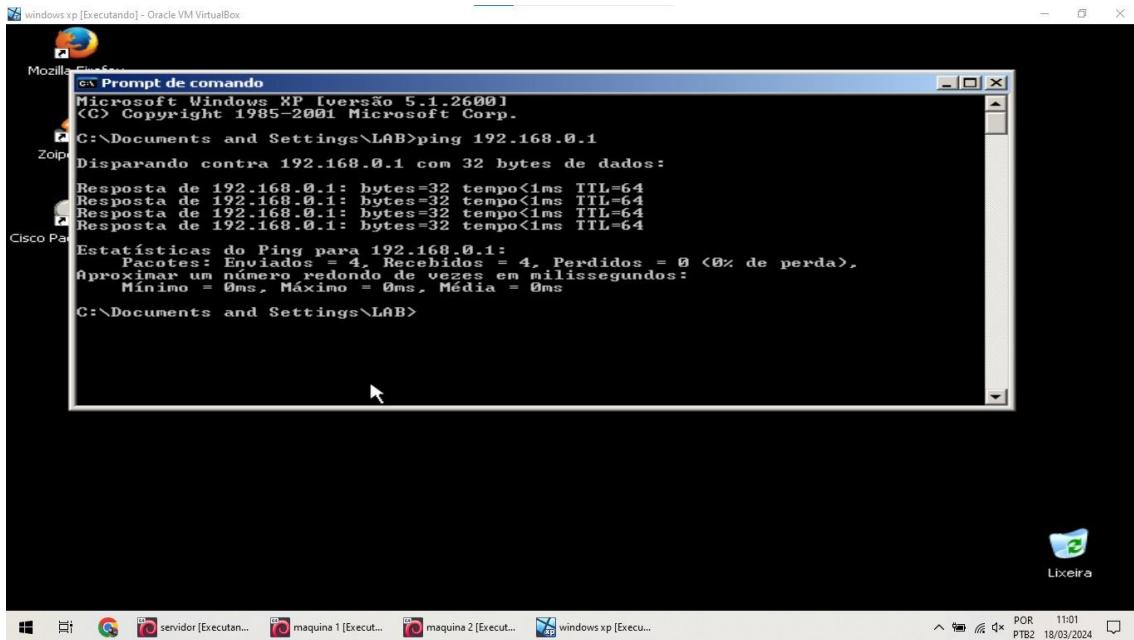


Figura 91” Tela84”

Terminamos o detonado com todas as máquinas conversando entre elas e tendo a acesso a um provedor de internet em estado funcional. Caso aconteça algum erro, repasse o passo a passo e verifique erros.

Considerações finais

Esse trabalho foi desenvolvido conforme as aulas de protocolos e roteamento de redes, 3º semestre de Segurança da Informação-matutino.

As imagens estão disponíveis no Google Drive pelo Link:

https://drive.google.com/drive/folders/1xWr-wrn-18zpxxEoqv9Kzhfx1RU6NUJI?usp=drive_link

Link para obter o aplicativo Virtual Box:

<https://www.virtualbox.org/>

Lista de Códigos

```
Ip a s  
vi /etc/network/interfaces  
service networking restart  
sysctl -p  
ip a r (IP) via (IP)  
vi /etc/sysctl.conf  
ping
```

Bibliografia

Vídeo youtube: Tutorial - IP forward, interceptação, troubleshooting, depuração de pacotes e NAT - Redes Linux - <https://www.youtube.com/watch?v=7w8jLCPKy7I> – acesso 26 de março de 2024.

FERNANDES, Lucas B.; RIBEIRO, Pedro H.; PASQUINI, Leonardo Martins Rafael; FAINA, Luis F.; CAMARGOS, Lasaro. Comutador P4 com Suporte a Roteamento Multicaminhos. In: WORKSHOP DE PESQUISA EXPERIMENTAL DA INTERNET DO FUTURO (WPEIF), 9. , 2018, Campos do Jordão. *Anais* [...]. Porto Alegre: Sociedade Brasileira de Computação, 2018. ISSN 2595-2692.