

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
DO CEARÁ  
CAMPUS FORTALEZA  
CURSO: TELEMÁTICA**



***Francisco Thyago de Lima Fernandes / Matrícula: 20231013020023***

**ATIVIDADE 4**

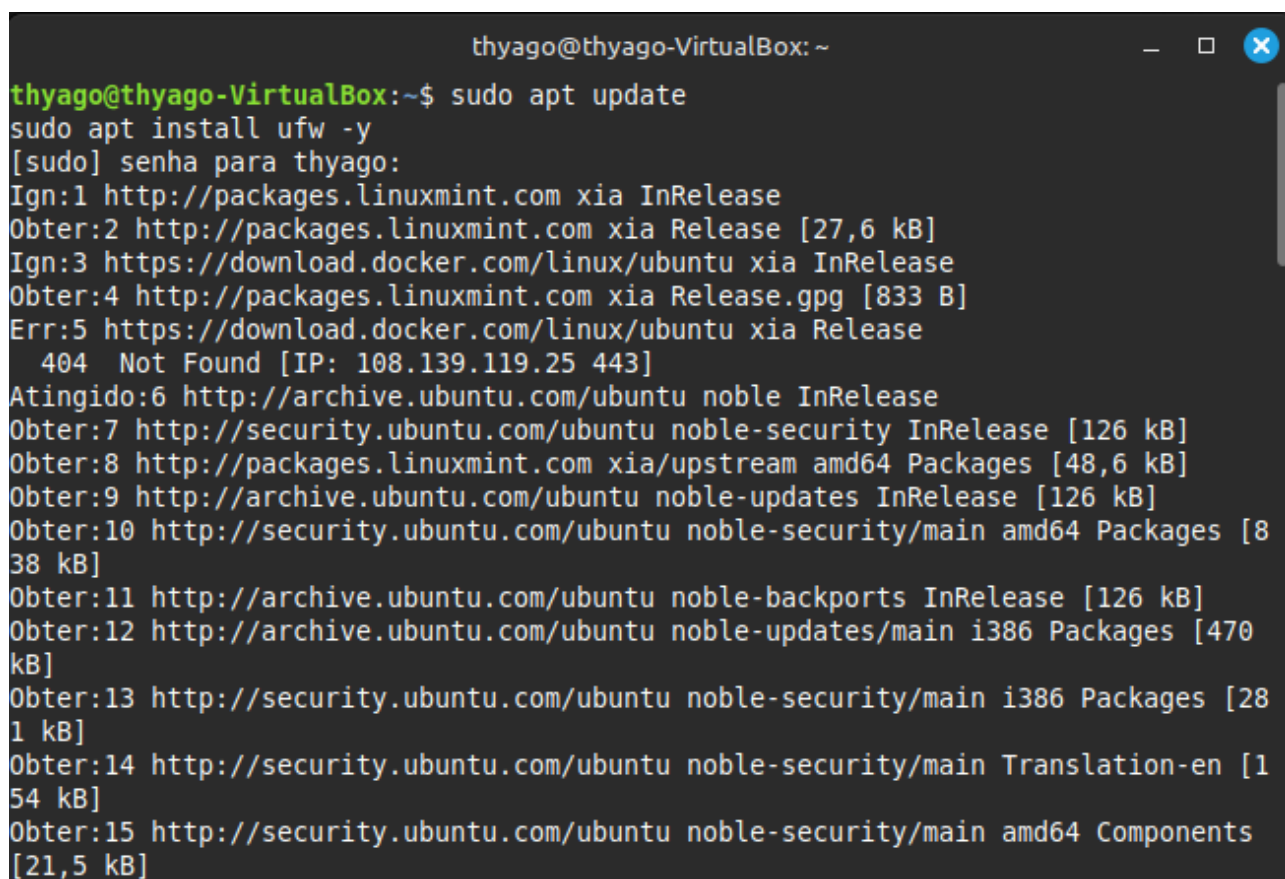
**FORTALEZA - CE  
15 / 05 / 2025**

Este trabalho tem como objetivo demonstrar, de forma prática e didática, o processo de instalação, configuração e testes do firewall **UFW (Uncomplicated Firewall)** no sistema operacional **Linux Mint**, utilizando uma máquina virtual no **VirtualBox**, com a rede configurada em **modo Bridge**, acessada a partir de uma máquina com **Windows 11**.

Inicialmente, foi necessário configurar corretamente a rede da máquina virtual. No VirtualBox, a placa de rede foi ajustada para operar em **modo Bridge**, garantindo que o Linux Mint pudesse receber e responder a conexões provenientes da rede local — incluindo o host com Windows.

Com a máquina Linux Mint inicializada, identificou-se seu endereço IP através do comando *ip a* no terminal (**ip da máquina virtual linux Mint: 192.168.100.39**). Esse IP seria utilizado posteriormente nos testes de conectividade.

Em seguida, procedeu-se com a instalação do firewall UFW por meio dos seguintes comandos (*sudo apt update*, seguido do comando *sudo apt install ufw -y*, conforme a figura 1:

A terminal window titled 'thyago@thyago-VirtualBox: ~' with standard window controls. The terminal shows the execution of 'sudo apt update' followed by 'sudo apt install ufw -y'. It prompts for the password 'thyago'. The output lists various package sources and their sizes, including a 404 error for a Docker package. The window has a scrollbar on the right.

```
thyago@thyago-VirtualBox: ~  
thyago@thyago-VirtualBox:~$ sudo apt update  
sudo apt install ufw -y  
[sudo] senha para thyago:  
Ign:1 http://packages.linuxmint.com xia InRelease  
Obter:2 http://packages.linuxmint.com xia Release [27,6 kB]  
Ign:3 https://download.docker.com/linux/ubuntu xia InRelease  
Obter:4 http://packages.linuxmint.com xia Release.gpg [833 B]  
Err:5 https://download.docker.com/linux/ubuntu xia Release  
404 Not Found [IP: 108.139.119.25 443]  
Atingido:6 http://archive.ubuntu.com/ubuntu noble InRelease  
Obter:7 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]  
Obter:8 http://packages.linuxmint.com xia/upstream amd64 Packages [48,6 kB]  
Obter:9 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]  
Obter:10 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [838 kB]  
Obter:11 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]  
Obter:12 http://archive.ubuntu.com/ubuntu noble-updates/main i386 Packages [470 kB]  
Obter:13 http://security.ubuntu.com/ubuntu noble-security/main i386 Packages [281 kB]  
Obter:14 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [154 kB]  
Obter:15 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [21,5 kB]
```

Figura 1 – Instalação do firewall.

Após a instalação, foram definidas as políticas padrão de segurança, bloqueando todas as conexões de entrada e permitindo todas as saídas utilizando os comandos *sudo ufw default deny incoming* e *sudo ufw default allow outgoing*. Veja na figura 2, a seguir:

```
thyago@thyago-VirtualBox: ~  
thyago@thyago-VirtualBox:~$ sudo ufw default deny incoming  
sudo ufw default allow outgoing  
[sudo] senha para thyago:  
A política padrão incoming mudou para 'deny'  
(assegure-se de ter atualizado suas regras apropriadamente)  
A política padrão outgoing mudou para 'allow'  
(assegure-se de ter atualizado suas regras apropriadamente)  
thyago@thyago-VirtualBox:~$
```

Figura 2 – Definindo as políticas padrão de segurança.

Posteriormente, o UFW foi ativado com o comando *sudo ufw enable*, conforme a figura 3:

```
thyago@thyago-VirtualBox: ~  
thyago@thyago-VirtualBox:~$ sudo ufw enable  
Firewall está ativo e habilitado na inicialização do sistema  
thyago@thyago-VirtualBox:~$
```

Figura 3 – Ativação do UFW.

Para permitir a resposta a requisições de **ping** — o que não é possível diretamente via comandos do UFW — foi necessário editar manualmente o arquivo de regras do firewall, utilizando o comando *sudo nano /etc/ufw/before.rules*. A seguinte regra foi inserida após a linha *\*filter*: *-A ufw-before-input -p icmp --icmp-type echo-request -j ACCEPT*

Verifique a figura 4:

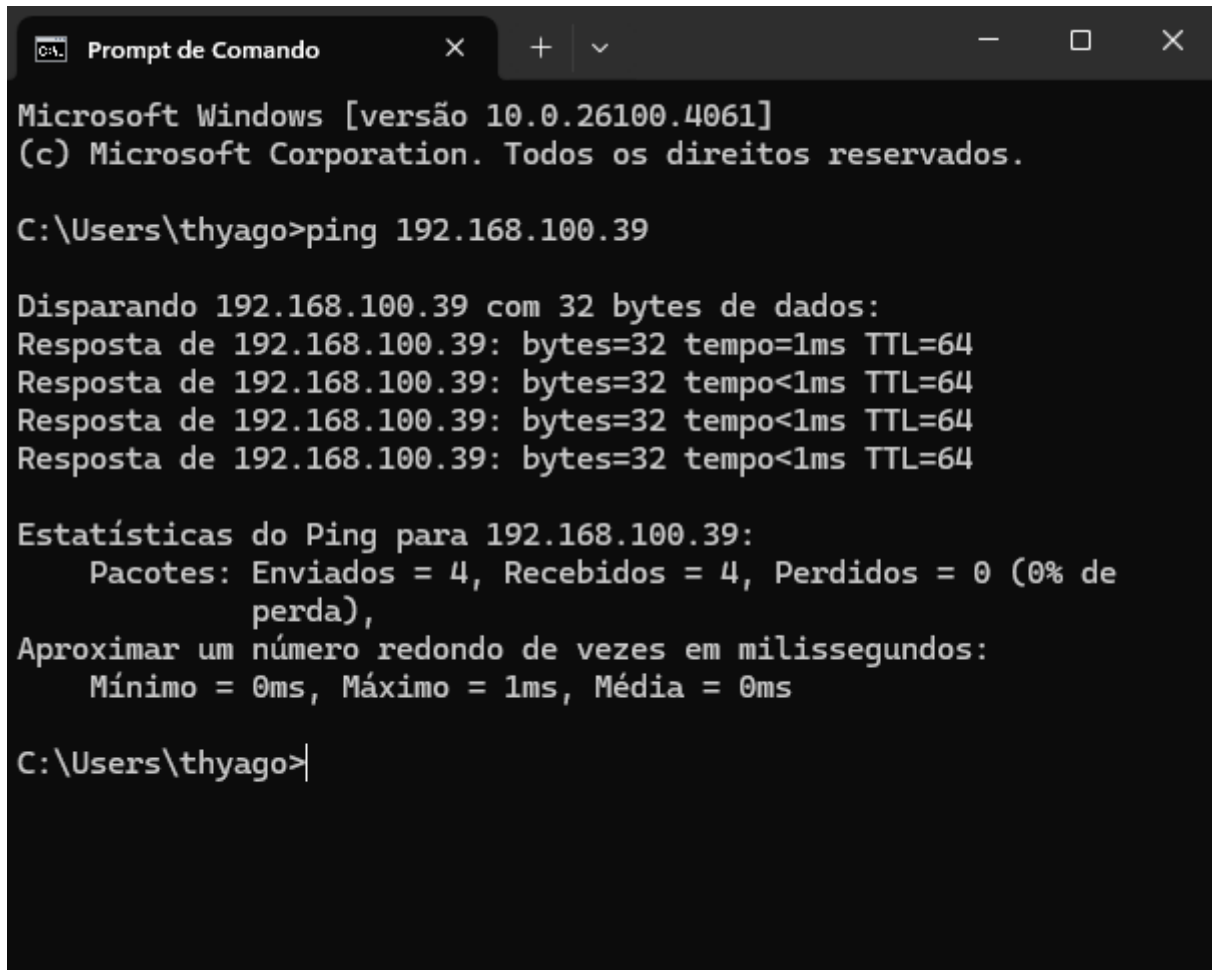
```
thyago@thyago-VirtualBox: ~  
GNU nano 7.2 /etc/ufw/before.rules  
#  
# rules.before  
#  
# Rules that should be run before the ufw command line added rules. Custom  
# rules should be added to one of these chains:  
#   ufw-before-input  
#   ufw-before-output  
#   ufw-before-forward  
#  
# Don't delete these required lines, otherwise there will be errors  
*filter  
-A ufw-before-input -p icmp --icmp-type echo-request -j ACCEPT  
:ufw-before-input - [0:0]  
:ufw-before-output - [0:0]  
:ufw-before-forward - [0:0]  
:ufw-not-local - [0:0]  
# End required lines  
  
^G Ajuda      ^O Gravar    ^W Onde está? ^K Recortar   ^T Executar   ^C Local  
^X Sair      ^R Ler o arq ^_ Substituir ^U Colar     ^J Justificar ^/ Ir p/ linha
```

Figura 4 – Adição de regra no firewall para efetuar o teste de ping.

Após a alteração, o UFW foi reiniciado com os comandos:

```
sudo ufw disable  
sudo ufw enable
```

Com as configurações aplicadas, foram realizados dois tipos de testes no **Windows 11**, sem a instalação de ferramentas adicionais. O primeiro teste utilizou o comando ping 192.168.100.39, inserido no prompt de comando da máquina física (host) Windows 11, para verificar a resposta ICMP da máquina virtual Linux Mint, conforme a figura 5:



```
Microsoft Windows [versão 10.0.26100.4061]  
(c) Microsoft Corporation. Todos os direitos reservados.  
  
C:\Users\thyago>ping 192.168.100.39  
  
Disparando 192.168.100.39 com 32 bytes de dados:  
Resposta de 192.168.100.39: bytes=32 tempo=1ms TTL=64  
Resposta de 192.168.100.39: bytes=32 tempo<1ms TTL=64  
Resposta de 192.168.100.39: bytes=32 tempo<1ms TTL=64  
Resposta de 192.168.100.39: bytes=32 tempo<1ms TTL=64  
  
Estatísticas do Ping para 192.168.100.39:  
    Pacotes: Enviados = 4, Recebidos = 4, Perdidos = 0 (0% de  
              perda),  
Aproximar um número redondo de vezes em milissegundos:  
    Mínimo = 0ms, Máximo = 1ms, Média = 0ms  
  
C:\Users\thyago>
```

Figura 5 – Teste de conexão ping a partir de do host Windows 11.

O segundo teste utilizou o comando *Test-NetConnection -ComputerName 192.168.100.39 -Port 9999*, no PowerShell do Windows 11, para verificar se a porta TCP estava sendo permitida pelo firewall da máquina virtual linux Mint, mesmo sem um serviço escutando, conforme a figura 6:

```
Windows PowerShell
PS C:\Users\thyago> Test-NetConnection -ComputerName 192.168.100.39 -Port 9999

ComputerName      : 192.168.100.39
RemoteAddress     : 192.168.100.39
RemotePort        : 9999
InterfaceAlias    : Ethernet
SourceAddress     : 192.168.100.37
TcpTestSucceeded  : True

PS C:\Users\thyago> |
```

Figura 6 – Teste para verificar se a porta TCP estava sendo permitida pelo firewall da máquina virtual linux Mint.

Antes disso, a porta 9999 foi liberada no UFW com o comando *sudo ufw allow 9999/tcp*. Veja na figura 7:

```
thyago@thyago-VirtualBox: ~
thyago@thyago-VirtualBox:~$ sudo ufw allow 9999/tcp
[sudo] senha para thyago:
Regra adicionada
Regra adicionada (v6)
thyago@thyago-VirtualBox:~$
```

Figura 7 – Comando para liberar a porta 9999.

A resposta do comando no PowerShell indicou, por meio do campo *TcpTestSucceeded*, se a conexão TCP foi bem-sucedida (porta liberada) ou negada (porta bloqueada).

## CONCLUSÃO

Através da execução prática deste procedimento, foi possível compreender de forma clara a funcionalidade do firewall UFW no Linux Mint, especialmente em ambientes virtuais controlados como o VirtualBox. A configuração da rede em modo Bridge se mostrou

essencial para simular um cenário de rede real, possibilitando testes eficazes a partir do sistema host.

O tutorial abordou desde a instalação e ativação do UFW até a liberação específica de portas e protocolos, com destaque para o protocolo ICMP, cujo desbloqueio exige modificação manual nos arquivos de configuração do firewall.

Os testes realizados a partir do Windows 11 demonstraram, de forma objetiva, o comportamento do firewall perante regras aplicadas, permitindo verificar com precisão o funcionamento das permissões e bloqueios.

Assim, conclui-se que o UFW é uma ferramenta poderosa e relativamente simples para configuração de segurança de rede no Linux, sendo plenamente funcional mesmo em ambientes virtualizados e acessível para iniciantes quando acompanhado de documentação clara e testes práticos.